

# PYTHON



# TH



Guide

# ACCESS

# POUR 4<sup>É</sup>ME ECONOMIE ET GESTION

# PANDAS



## Table des matières

A-	LES BASES DE DONNEES :.....	4
<b>I-</b>	<b>Introduction :.....</b>	<b>4</b>
<b>II-</b>	<b>Les composants d'une Base Données (BD) :.....</b>	<b>4</b>
<b>1-</b>	<b>Schéma d'une table :.....</b>	<b>5</b>
2-	Schéma d'une base de données :.....	5
3-	Lancement du logiciel Access :.....	5
4-	Création une base de données vide.....	5
5-	Création une table :.....	6
a-	Définition :.....	6
b-	Créer une clé primaire.....	7
7-	Les relations :.....	7
a-	Définition :.....	7
b-	Les types de relations :.....	8
c-	Créer une relation :.....	9
d-	L'intégrité référentielle :.....	10
e-	Saisie des données dans une table :.....	11
<b>III-</b>	<b>Création des requêtes :.....</b>	<b>12</b>
1-	Définition :.....	12
2-	Les requêtes de sélection.....	12
a-	Créer une requête de sélection :.....	12
b-	Exécuter une requête de sélection :.....	12
3-	Les requêtes paramétrées : (pour une donnée ou une valeur saisie par l'utilisateur).....	17
a-	Définition :.....	17
4-	Les requêtes de mise à jour :.....	19
a-	Définition.....	19
b-	Création d'une requête de mise à jour :.....	19
5-	Les requêtes de suppression:.....	20
a-	Définition.....	20
b-	Création d'une requête de suppression.....	20
6-	Les requêtes de calcul:.....	21
a-	Définition.....	21
b-	Création d'une requête de calcul.....	21
7-	Les requêtes d'ajout:.....	21
a-	Définition.....	21
b-	Création d'une requête d'ajout.....	21
	Pour plus d'information regarder la vidéo de création des requêtes sur youtube :.....	23
	.....	23
<b>IV-</b>	<b>Importer des données :.....</b>	<b>23</b>

1- A partir d'une feuille Excel : .....	23
.....	25
2- A partir un fichier texte ou CSV : .....	25
V- Exportation des données d'une table ou une requête .....	27
1- Vers une feuille Excel .....	27
2- Vers un fichier csv ou txt.....	28
<b>PROTOTYPE 2021(PARTIE ACCESS)</b> .....	31
<b>CORRECTION PROTOTYPE 2021 :PARTIE ACCESS)</b> .....	34
B- ANALYSE DE DONNEES.....	37
I- introduction : .....	37
II- importation de la bibliothèque pandas :.....	37
III- Chargement des données d'un DataFrame : .....	38
<b>1- Création d'un DataFrame :</b> .....	38
<b>2- Importation d'un DataFrame :</b> .....	39
a- A partir d'un fichier Excel.....	39
b- A partir d'un fichier csv.....	40
c- A partir d'un fichier texte .....	41
.....	42
IV- Information d'un DataFrame : .....	43
1- Dimension d'un DataFrame :.....	43
2- Taille d'un DataFrame :.....	44
3- Informations relatives d'un DataFrame : .....	44
5- Les noms des colonnes d'un DataFrame : .....	46
V- Manipulation d'un DataFrame : .....	46
<b>1- Renommage d'une ou plusieurs colonnes d'un DataFrame :</b> .....	46
.....	48
<b>1- Affichage des données d'un DataFrame :</b> .....	53
<b>2- Affichage des données d'une ligne d'un DataFrame :</b> .....	55
<b>3- Affichage le contenu d'une cellule d'un DataFrame :</b> .....	56
<b>4- Affichage n premières lignes d'un DataFrame :</b> .....	56
<b>5- Affichage n dernières lignes d'un DataFrame :</b> .....	57
<b>6- Affichage d'une plage de données d'un DataFrame :</b> .....	57
.....	58
<b>V- Tri :</b> .....	58
<b>1- Tri d'un DataFrame selon les données :</b> .....	58
VII. Filtrage des données d'un DataFrame.....	60
<b>1- Filtrage selon une seule condition :</b> .....	60
<b>2- Filtrage selon deux conditions :</b> .....	61
VIII. Les fonctions statistiques .....	61

IX. Graphiques : .....	62
1- <b>Graphique à barres</b> .....	62
2- <b>Graphique en courbe</b> .....	64
X. Création d'une colonne calculable : .....	65
XI- Nettoyage d'un DataFrame : .....	65
.....	66
TP de révision .....	67
Correction du TP : .....	68
Devoir 1 .....	69
Devoir 2 .....	71
Correction Devoir 1 :(pandas) .....	73
Correction devoir 2 : (pandas) .....	75
<b>correction Bac théorique 2022</b> .....	77
<b>PROTOTYPE 2022</b> .....	81
Correction de prototype 2022 .....	82
<b>ANNEXE</b> .....	83



## A- LES BASES DE DONNEES :

### I- Introduction :

Dans notre vie quotidienne on manipule une quantité importante des données (Nom de personne, son Téléphone, dates, notes...). On a besoin donc d'une structure permettant d'assurer une organisation simple de ces données en précisant leurs types et leurs tailles.

**Cette nouvelle notion est appelé base de donnée (data base)**

#### 1- Base de données :

Une **base de données (BD)** est un **ensemble structuré de données** mémorisées sur des supports accessibles par un ordinateur pour satisfaire simultanément plusieurs utilisateurs de façon sélective et en temps très court.

#### 2- Système de Gestion de Bases de Données (SGBD) :

C'est l'outil permettant de créer de nouvelle bases de données, de mettre à jour ces données (ajouter, modifier, supprimer), de rechercher efficacement des données spécifiques et de fournir différentes manières de les visualiser (à l'écran ou sur papier).

*Exemple* : Oracle, DB2, MySQL, Access, SQL Server, ...

#### 3- Avantage :

- Rapidité d'accès aux données
- La non-redondance : pas de répétition
- L'organisation des données



**Retenons :** Une base de données est un ensemble structuré de données enregistrées dans des tables, chaque table représente un objet du monde réel. (Une voiture, un élève, une personne...)

## II- Les composants d'une Base Données (BD) :

Colonnes

	Id-Eleve	Nom	Prénom	Date de naissance	Classe
+	E001	Ben salem	Nizar	09/09/2004	4 Eco 2
+	E002	Kefi	Rym	17/03/2005	4 Eco 2
+	E003	Abrougui	Amal	03/02/2005	4 Eco 1
+	E004	Sallami	Maroua	28/11/2004	4 Eco 1
+	E005	Choterana	Abir	27/04/2005	4 Math 2
+	E006	Abbassi	Maram	03/12/2004	4 Eco 3

Lignes

- ✓ Table : c'est une collection de données relatives à un sujet spécifique tel qu'un élève, un véhicule, un client, répartie **en colonnes (champs) et en lignes (enregistrements)**. On peut trouver plusieurs tables dans une même base de données.
- ✓ Une **colonne** d'une table est appelée **champs ou attribut** qui regroupe les informations de même type



- ✓ **Une ligne** d'une table est appelée **enregistrement** qui correspond à toutes les informations de la liste des cellules

### 1- Schéma d'une table :

Le schéma d'une table est donné par l'ensemble de ses champs. (attributs).

#### Exemple :

- Le schéma de la table **ELEVE** est :

ELEVE (ID\_ELEVE, Nom, Prénom, date\_naissance, classe)

De même pour les autres tables :

- Le schéma de la table **MATIERE** est :

MATIERE (ID\_MATIERE, Libelle\_matiere, coefficient)

- Le schéma de la table **NOTE** est :

NOTE (ID\_ELEVE#, ID\_MATIERE#, Note)

### 2- Schéma d'une base de données :

Le schéma d'une base de données est l'ensemble des schémas des tables de la base.

Donner Le schéma de la base de données « **GESTION DE NOTES** ».

#### Solution :

Le schéma de la base de données « **GESTION DE NOTES** » est :

ELEVE (ID\_ELEVE, Nom, Prenom, date de naissance, classe)

MATIERE (ID\_MATIERE, Libelle\_matiere, coefficient)

NOTE (ID\_ELEVE#, ID\_MATIERE#, Note)

### 3- Lancement du logiciel Access :

Démarrer le SGBD installé sur votre ordinateur Microsoft Access 2016.

**Access** est un logiciel qui gère des bases de données relationnelles. Ce type de logiciel est appelé système de gestion de bases de données relationnel (**SGBDR**).

### 4- Création une base de données vide

Créer la base de données nommée : « **gestion de notes** » :

- Base de donnée vide
- Préciser le nom et l'emplacement de la base
- Créer

Pour plus d'information regarder la vidéo de création d'une base de données sur youtube :



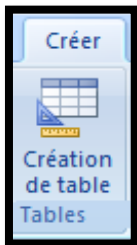
### 5- Création une table :


Soit la description des colonnes des tables de la base de données "gestion de note":

Nom	Description	Type	Taille	Format	Contrainte
<i>ID_eleve</i>	Le code d'un élève	Texte court	4		Non vide
<i>ID_matiere</i>	Le code d'une matière	Texte court	3		Non vide
<i>Nom</i>	<i>Nom</i> Nom d'un élève	Texte court	20		
<i>Prénom</i>	<i>Prénom</i> d'un élève	Texte court	10		
<i>Libelle_matiere</i>	<i>Libellé</i> d'une matière	Texte court	30		
<i>Date_naissance</i>	<i>date de naissance</i> d'un élève	Date/Heure		Date abrégé	
<i>classe</i>	Classe d'un élève	Texte court			
<i>Coef</i>	Coefficient d'une matière	Numérique	Entier long		Par défaut 1
<i>Note</i>	Note d'un élève dans une matière	Numérique	réel	Nombre général (2 chiffres après la virgule)	Supérieure ou égal à zéro et inférieure ou égale à 20

Créer les tables de la base de données en utilisant le tableau ci-dessus.

Pour créer une table, on doit suivre les étapes suivantes :



- Onglet « créer »
- Bouton « création de table »
- Créer les champs de la table et fixer pour chacun ces propriétés (nom, type, ...)
- Choisir la clé primaire de la table en cliquant sur l'outil  de la barre d'outils
- Fermer la table
- Enregistrer la table.

### 6- La clé primaire :

Dans une classe, est ce qu'on peut avoir deux élèves qui ont le même nom, prénom et date de naissance ?

#### Bien sûr oui

Qu'elle est le champ qui doit être unique pour chaque élève ?

Réponse :

L'ID-Eleve : qui représentera la clé primaire de la table ELEVE



#### a- Définition :

Une clé primaire **est un ou plusieurs champs** dont la ou les valeurs identifient de façon unique chaque enregistrement d'une table. Une clé primaire n'accepte pas les valeurs nulles.

#### Remarque :


*Une clé primaire peut être composée d'un ou par plusieurs champs et lie, généralement, les informations communes à plusieurs tables*

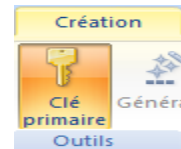
*Lorsqu'on ne précise pas clé primaire le SGBD ajoute automatiquement un nouveau champ et le considère comme clé primaire*

*La clé primaire est très importante dans une table car le SGBD empêche: Les doublons et les valeurs nulles pour la clé primaire*

**b- Créer une clé primaire**

Pour créer la clé primaire de la table :

- Sélectionnez le champ ou les champs concernés
- Cliquez sur le bouton clé primaire 

**7- Les relations :**

Quelle est la clé primaire de la table note ? Pourquoi ils sont 2 champs qui définies la clé de la table ?

**Parce que:**

Pour assurer la cohérence de cette BD, on doit appliquer des contraintes ou règle d'intégrité :

- Un élève peut avoir plusieurs notes : relation 1 élève-N note
- Une note concerne un seul élève : relation 1 note – 1 élève
- Une matière contient plusieurs notes : relation 1 matière-N note

Et pour assurer la cohérence de cette base de données, on peut appliquer les règles de gestion suivantes :

- Une note, dans une matière, n'est affectée qu'à un seul élève.
- Un élève peut avoir plusieurs notes.
- Une note est associée à une matière.
- Plusieurs notes peuvent être affectées à une matière pour plusieurs élèves.

**a- Définition :**

Une relation est un lien ou liaisons entre 2 tables d'une BD à l'aide de 2 champs en commun à ces 2 tables. Ces 2 champs sont dits : associés. Un tel champ est la clé primaire de l'une des tables et la clé étrangère dans l'autre table associée.

---

**Remarque :**

---

- Une relation est un lien entre 2 tables d'une BD à l'aide de 2 champs en commun.
  - L'avantage de relier des tables sur des champs en commun est d'éviter la redondance (avoir une même information plusieurs fois).
  - Les deux champs en commun se sont généralement une clé primaire et une clé étrangère
  - Une clé étrangère est une clé primaire dans sa table.
- 





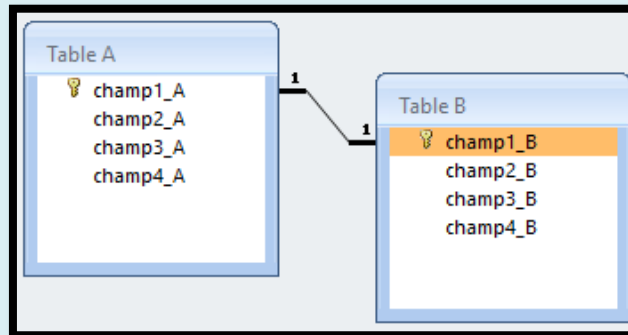
### b- Les types de relations :

Soit les 2 tables **Table A** et **Table B** d'une base de données. On distingue trois types de relations (appelées cardinalités) : "un- à- un", "un- à -plusieurs" et "plusieurs- à- plusieurs"

#### Relation un à un (1-1) :

Signifie qu'un enregistrement de la 1<sup>ère</sup> table ne peut correspondre qu'à un seul champ de la 2<sup>ème</sup> table.

*Exemple* : Un navire est commandé par un seul commandant et un commandant commande un seul navire On définit ainsi une relation de type 1-1.

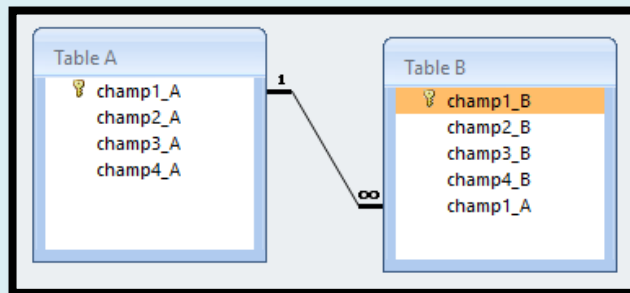


Une relation un à un (1-1) est créée si les deux champs liés sont des clés primaires.

#### Relation un à plusieurs (1-N) :

Signifie que chaque enregistrement de la 1<sup>ère</sup> table (contenant la clé primaire) peut être associé à plusieurs enregistrements de la 2<sup>ème</sup> table.

*Exemple* : Un élève peut avoir plusieurs notes alors qu'une note dans une matière ne peut être affectée qu'à un seul élève. On définit ainsi une relation de type 1-N.



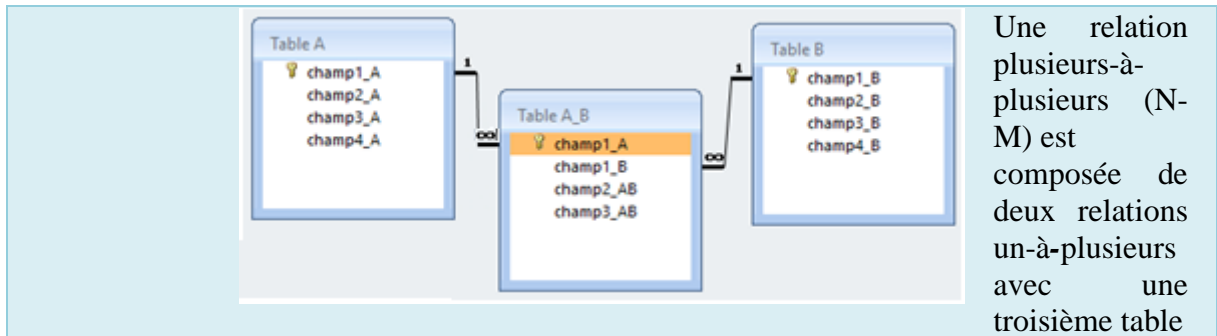
Une relation un à plusieurs (1-N) est créée si l'un des deux champs liés sont des clés primaires.

**Les clés étrangères / clés secondaires** Une clé étrangère est, généralement, une clé primaire provenant d'une autre Table pour définir une relation entre les deux tables en question.

#### Relation plusieurs à plusieurs (N-M) :

Signifie que chaque enregistrement de la 1<sup>ère</sup> table (contenant la clé primaire) peut être associé à plusieurs enregistrements de la 2<sup>ème</sup> table et inversement.

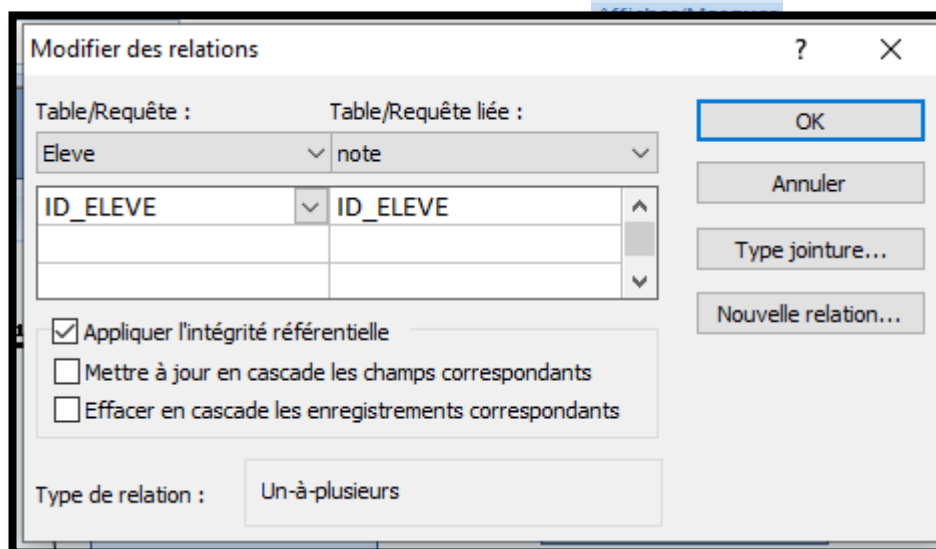
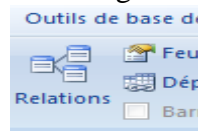
*Exemple* : Un enseignant enseigne plusieurs matières et une matière est enseignée par plusieurs enseignants



### c- Créer une relation :

Créer les relations possibles entre ses tables en admettant les règles décrites précédemment

- Onglet « **Outil de bases de donnée** »
- Bouton « **Relations** »

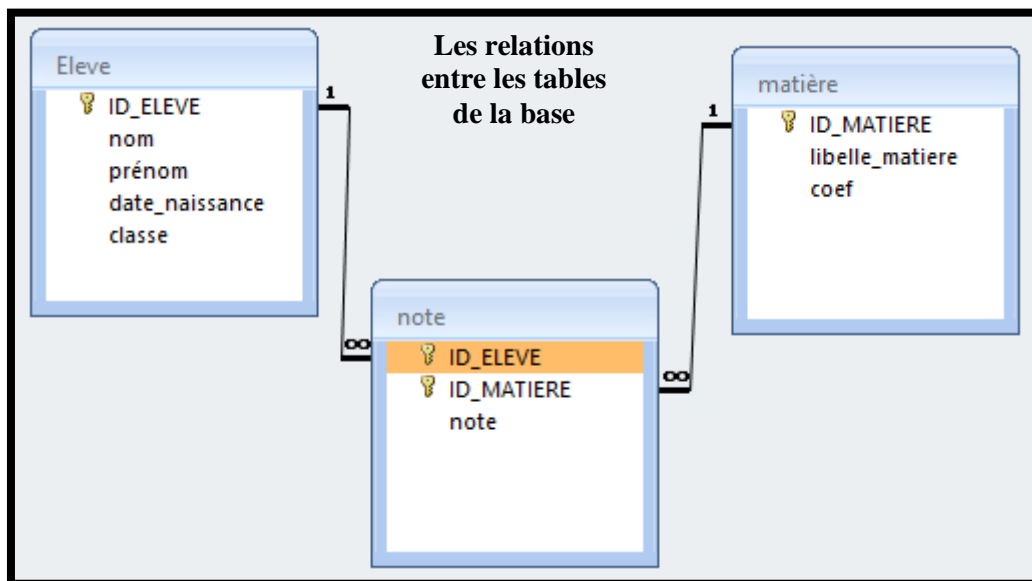
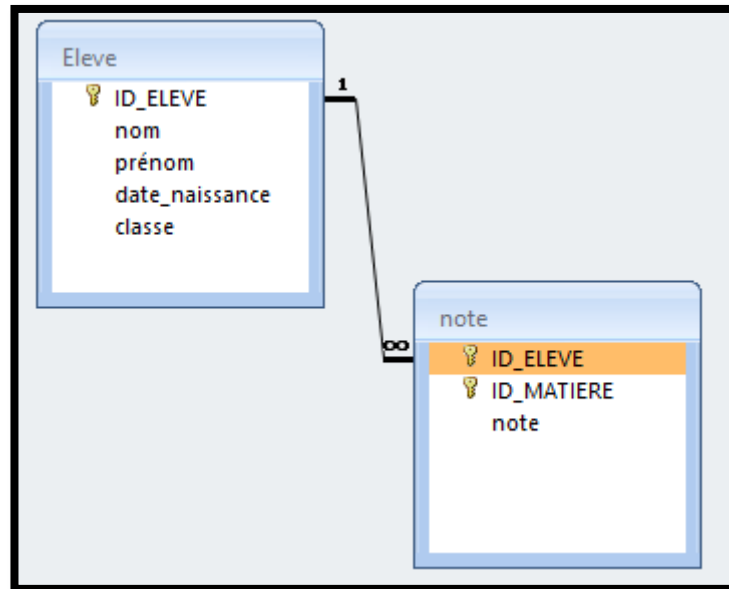


Sélectionner les clés primaires des tables et les glisser en maintenant le bouton gauche de la souris enfoncé vers les clés étrangères des autres tables

Par exemple : sélectionner la clé primaire **id\_eleve** de la table élève et le glisser, en maintenant le bouton gauche de la souris, vers la clé étrangère de la table **note**.

- Vérifier dans la boîte de dialogue affichée, les champs et l'intégrité référentielle.
- Valider ce paramètre en cliquant sur « **Créer** ».
- Enregistrer les modifications apportées.

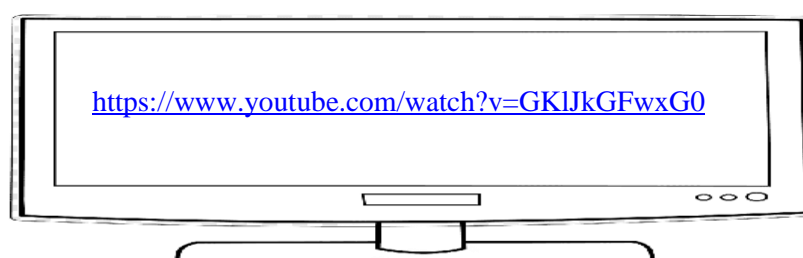
La relation sera représentée comme suit :



#### d- L'intégrité référentielle :

Signifie que lorsqu'on insère une ligne dans la table fille, le SGBD vérifie que la valeur saisie dans la colonne clé étrangère existe dans la colonne clé primaire de la table mère. C'est-à-dire qu'elle permet de s'assurer que chaque valeur de clé saisie dans la table fille existe bien dans la table mère.

Pour plus d'information regarder la vidéo de création des relations sur youtube



### e- Saisie des données dans une table :

Remplir les tables créées dans l'activité précédente par les données nécessaires.

- ✓ Sélectionner l'objet tables
- ✓ Double clic sur la table à remplir
- ✓ Dans la fenêtre qui apparaît saisir les différents enregistrements

+	E001	Ben salem	Nizar	09/09/2004	4 Eco 2
+	E002	Kefi	Rym	17/03/2005	4 Eco 2
+	E003	Abrougui	Amal	03/02/2005	4 Eco 1
+	E004	Sallami	Maroua	28/11/2004	4 Eco 1
+	E005	Choterana	Abir	27/04/2005	4 Math 2
+	E006	Abbassi	Maram	03/12/2004	4 Eco 3
+	E007	Boughanmi	Ahmed	03/06/2003	4 Eco 3
+	E008	Mizouni	Nejmaddine	15/08/2004	4 Math 1
+	E009	Mestiri	Nassira	14/10/2004	4 Sciences 1
+	E010	Majdoub	Wareda	19/01/2004	Eco 2

matière		
ID_MATIERE	libelle_matiere	coef
M01	Info	3
M02	lettre	1
M03	economie	3
M04	gestion	3
M05	science	3

note		
ID_ELEVE	ID_MATIERE	note
E001	M01	12
E001	M03	20
E002	M01	9,75
E003	M01	12,25
E003	M02	15,5
E003	M04	18,5
E003	M05	17,75
E004	M01	16

#### Remarque :



- Dans la saisie il ne faut laisser le champ clé vide.
- Il ne faut pas que deux enregistrements aient une même valeur de clé primaire.
- La table note qui contient la relation N-M sera la dernière à remplir car les valeurs de id\_eleve et id\_matiere doivent exister dans les tables mère (ELEVE et MATIERE) (c'est l'intégrité référentielles)

Pour plus d'information regarder la vidéo de l'insertion des données sur youtube :



### III- Création des requêtes :

#### 1- Définition :



Une requête est un moyen permettant d'interroger une BD et d'exploiter ses données (classer des données, rechercher des données, extraire des données, effectuer des calculs).

Il existe plusieurs types de requêtes dont les plus utilisées sont :

- ✓ **Les requêtes de sélection**
- ✓ **Les requêtes paramétrées**
- ✓ **Les requêtes d'ajout**
- ✓ **Les requêtes de mise à jour**
- ✓ **Les requêtes de calcul**
- ✓ **Les requêtes de suppression**



#### 2- Les requêtes de sélection

##### a- Créer une requête de sélection :



- Onglet « créer »
- Bouton « création de requête »
- Sectionner le type de requête

**Remarque :** Pour afficher toutes les colonnes, on peut glisser le champ "\*".

##### b- Exécuter une requête de sélection :

Le résultat d'une requête est sous forme d'une table qui peut être utilisée dans une autre requête

- Onglet « Accueil »
- Groupe « Résultats »
- Bouton « Exécuter »

Ou bien

- Double-cliquer sur le nom de la requête.
- Le résultat s'affiche sous la forme d'une table.

La sélection permet de rechercher des informations dans des BD généralement, une requête de sélection :

- S'applique soit à une seule table ou à plusieurs tables liées par des relations,
- Permet d'afficher des informations en appliquant un ou a plusieurs critères sur un ou plusieurs champs
- Permet d'enregistrer le résultat sous forme de tables

On peut utiliser dans les requêtes les opérateurs suivants :

=	<>	<	<=	>	>=
égal	différent	inférieur	Inférieur ou égal	supérieur	supérieur ou égal

ACCESS met à votre disposition d'autres operateurs :

- **Entre** : sélectionne les enregistrements pour lesquels la valeur d'un champ est comprise dans un intervalle de valeurs,
    - ✓ Entre "B" et "K"
    - ✓ Entre 50 et 120
    - ✓ Entre #01/01/2022# et #31/12/2022#
  - **Comme** : sélectionne les enregistrements contenant une donnée approximative
- Avec : \* : remplace plusieurs caractères au hasard  
 ? : remplace un seul caractère au hasard

**Question 1 :** Afficher la liste des élèves triée dans l'ordre croissant des noms

Il faut choisir « Croissant » sous le champ « Nom » sur la ligne Tri.

La table utilisé dans la requête

eleve

---

\*

- 🔑 Id-Eleve
- Nom
- Prénom
- Date de naissance
- Classe

Champ :	Id-Eleve	Nom	Prénom	Date de naissance	Classe
Table :	eleve	eleve	eleve	eleve	eleve
Tri :		Croissant			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :					
Ou :					

**Question 2 :** Afficher les élèves de la classe "4 ECO1"  
 Il faut écrire "4 ECO1 "sous le champ " Classe" sur la ligne Critère.



eleve

---

\*

- Id-Eleve
- Nom
- Prénom
- Date de naissance
- Classe

Champ :	Id-Eleve	Nom	Prénom	Date de naissance	Classe
Table :	eleve	eleve	eleve	eleve	eleve
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :					"4 eco 1"
Ou :					

**Question 3 :** Afficher les élèves qui ne sont pas en "4 ECO1"  
 Il faut écrire "<>4 ECO1" sous le champ "Classe" sur la ligne Critère. Ou bien pas "4 eco 1"

eleve

---

\*

- Id-Eleve
- Nom
- Prénom
- Date de naissance
- Classe

Champ :	Id-Eleve	Nom	Prénom	Date de naissance	Classe
Table :	eleve	eleve	eleve	eleve	eleve
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :					Pas "4 eco 1"
Ou :					

**Question 4 :** Afficher les élèves dont les noms commencent par la lettre « B »  
 Il faut écrire **Comme "B\*"** sous le champ « Nom » sur la ligne Critère.



eleve

---

\*

Id-Eleve

Nom

Prénom

Date de naissance

Classe

Champ :	Id-Eleve	Nom	Prénom	Date de naissance	Classe
Table :	eleve	eleve	eleve	eleve	eleve
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :		Comme "B*"			
Ou :					

**Question 5 :** Afficher les élèves nés après le 01/01/2005

Il faut écrire `>#01/01/2005#` sous le champ "Date de naissance" sur la ligne Critère.

eleve

---

\*

Id-Eleve

Nom

Prénom

Date de naissance

Classe

Champ :	Id-Eleve	Nom	Prénom	Date de naissance	Classe
Table :	eleve	eleve	eleve	eleve	eleve
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :				>#01/01/2005#	
Ou :					

**Question 6 :** Afficher les élèves dont les noms commencent par la lettre " A " et nés après le 01/01/2005

Il faut écrire « **Comme "A\*"** » sous le champ "Nom" et "`>#01/01/2005#`" sous le champ " Date de naissance", les deux sur la ligne Critère.





Champ :	Id-Eleve	Nom	Prénom	Date de naissance	Classe
Table :	eleve	eleve	eleve	eleve	eleve
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :		Comme "A*"		>#01/01/2005#	
Ou :					

**Question 7 :** Afficher les noms et les prénoms des élèves qui ont une note égale à 10  
 Dans ce cas, on va ajouter deux tables liées la table ELEVE et la table NOTE.  
 Ensuite, glisser les champs Nom et Prénom de la table ELEVE et le champ Notes de la table NOTE.  
 Ecrire 10 sous le champ " Notes" sur la ligne Critère.

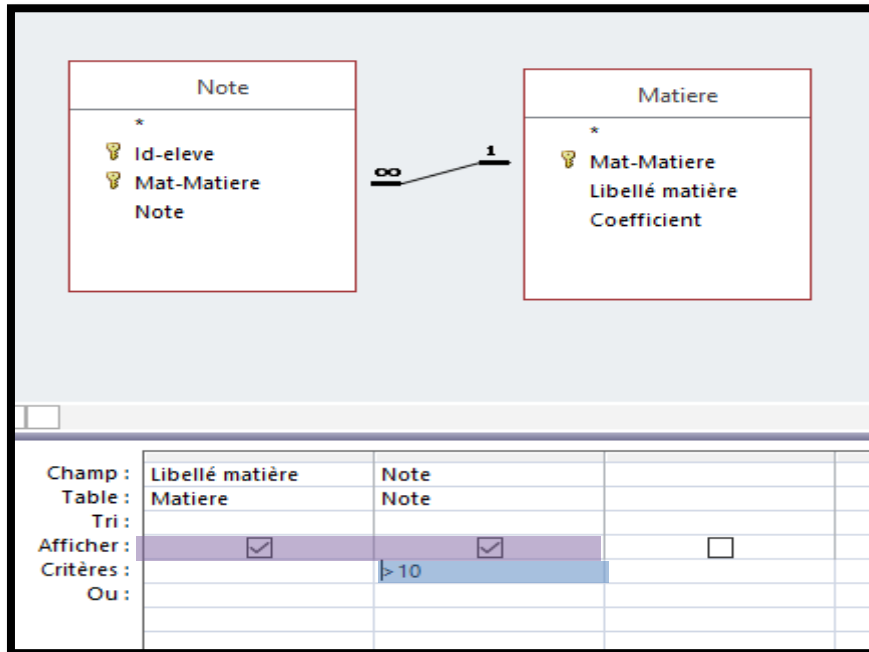
Champ :	Nom	Prénom	Note
Table :	eleve	eleve	Note
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :			=10
Ou :			

**Question 8 :** Afficher les libellés des matières dans lesquelles les élèves ont des notes supérieures à 10

Ajouter les deux tables MATIERE et la table NOTE qui sont liées.

Ensuite, glisser les champs Libellé\_matière de la table MATIERE et le champ Notes de la table NOTE.

Ecrire ">10" sous le champ "Notes" sur la ligne Critère.



### 3- Les requêtes paramétrées : (pour une donnée ou une valeur saisie par l'utilisateur)

Pour consulter les notes des élèves on créer une requête de chaque élève.

Non tout simplement.

On peut donner ou saisir l'IDeleve pour avoir ses notes.

Il suffit d'écrire un message **entre deux crochets ([ ])** sur la ligne Critère. ( c'est la requête paramétrées)

#### a- Définition :



Une requête paramétrée permet de créer à chaque fois une table résultat suivant un critère donné ou une valeur saisie par l'utilisateur dans l'exécution, donc la requête est écrite une seule fois mais le critère change pour chaque exécution, donc le résultat diffère d'une exécution à une autre.



#### Remarque :

Dans la grille de création de la requête, le paramètre doit être représenté par un message écrit entre crochets ([ Saisir ou donner.....])



**Question 9 :** Créer une requête paramétrée nommée « **Classe** » : permettant d'afficher les élèves **d'une classe donnée**.

Champ :	Id-Eleve	Nom	Prénom	Date de naissance	Classe
Table :	eleve	eleve	eleve	eleve	eleve
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :					[donner le nom de la classe]
Ou :					

**Question 10 :**

Afficher les notes **d'un élève donnée** dans chaque nom de matières

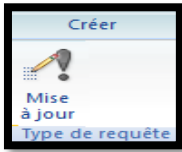
Champ :	Id-eleve	Libellé matière	Note
Table :	Note	Matiere	Note
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :	[donner ID de l'élève]		
Ou :			

#### 4- Les requêtes de mise à jour :

##### a- Définition

C'est une requête qui apporte des changements globaux à des enregistrements dans une ou plusieurs tables.

##### b- Création d'une requête de mise à jour :

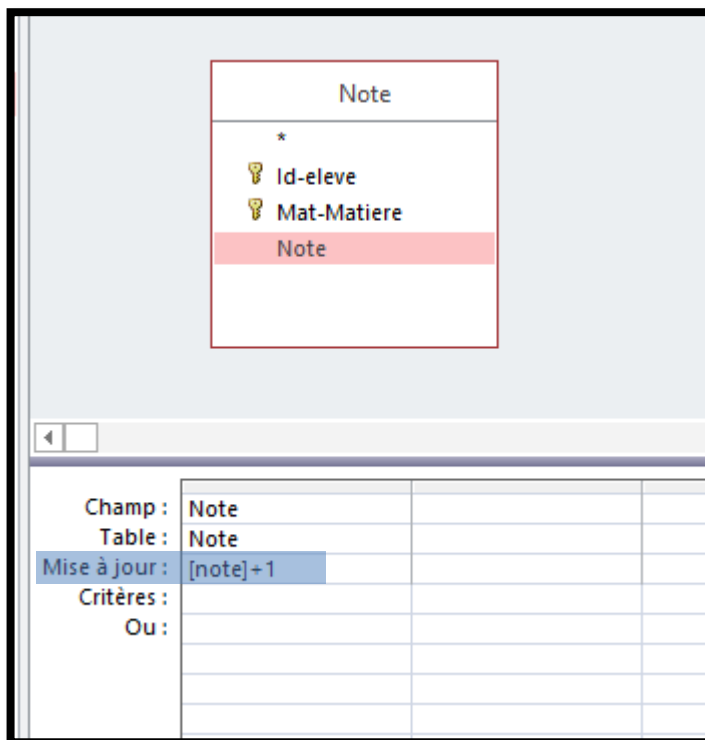


Pour créer une requête de mise à jour:

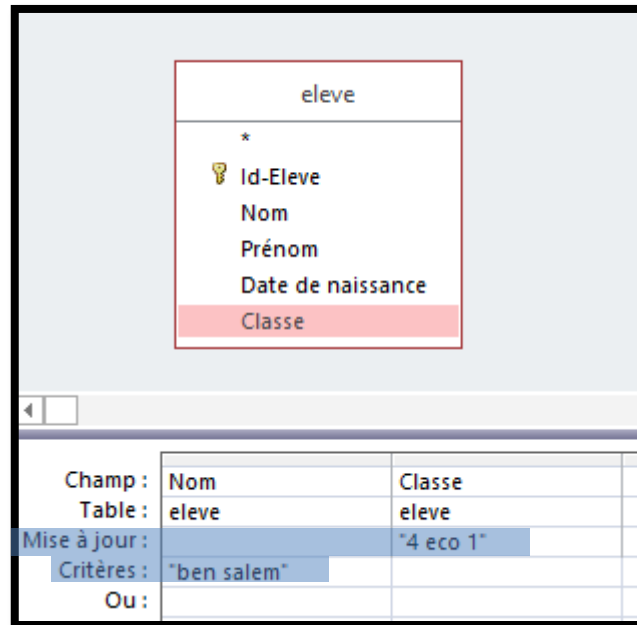
- Onglet « créer »
- Choisir « mise à jour »

**Question 11 :** Ecrire la requête qui permet d'augmenter de 1 point les notes des élèves

- Dans la ligne « Mise à jour », taper la modification : [Note]+ 1



**Question 12:** Ecrire la requête qui permet de modifier la classe, de l'élève ayant le nom "ben salah" en "4 eco 1"

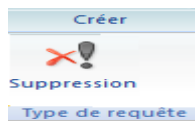


### 5- Les requêtes de suppression:

#### a- Définition

Elle permet de supprimer un groupe d'enregistrements d'une ou plusieurs tables

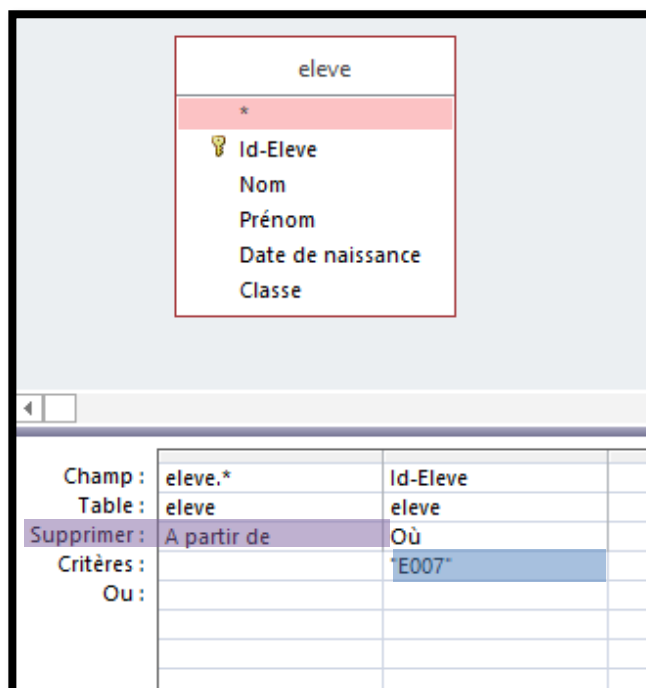
#### b- Création d'une requête de suppression



- Onglet « créer »
- Choisir requête de suppression

Dans la grille de création de la requête, fixer les champs et leurs tables à utiliser, cocher les cases des attributs à afficher et fixer les critères de suppression

**Question 13 :** Ecrire la requête qui permet de supprimer l'élève dont le code est "E007".



**Eleve .\*** :c'est-à-dire l'enregistrement entier.

Lorsque on supprime l'élève "E007", on supprimer la totalité de la ligne c'est pourquoi on écrit **eleve.\***

## 6- Les requêtes de calcul:

### a- Définition

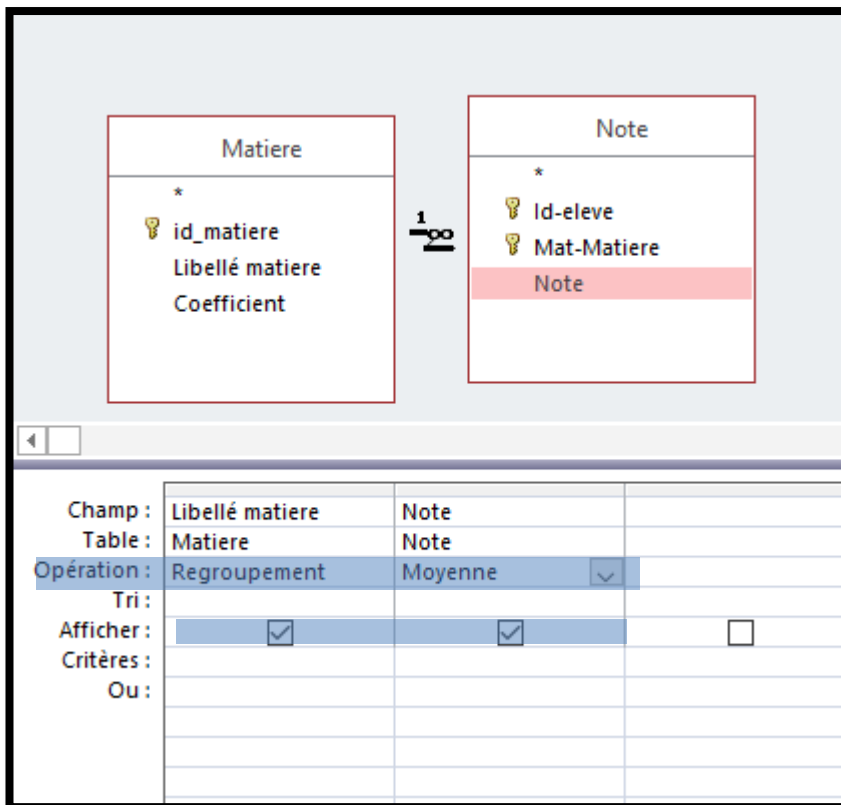
Elle permet d'effectuer des calculs (moyenne, compte, maximal, minimal d'une liste, somme...)

### b- Création d'une requête de calcul



- Onglet « créer »
- Bouton « création de requête »
- Onglet « créer »
- Cliquer sur « Totaux »

**Question 14 :** Écrire la requête qui permet d'afficher la moyenne générale des notes des élèves en mathématique.



Champ :	Libellé matiere	Note	
Table :	Matiere	Note	
Opération :	Regroupement	Moyenne	▼
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :			
Ou :			

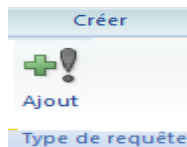
## 7- Les requêtes d'ajout:

### a- Définition

Elle permet d'ajouter un groupe d'enregistrements d'une ou plusieurs tables à la fin d'une ou plusieurs tables.

### b- Création d'une requête d'ajout

Pour créer une requête d'ajout :



- Onglet « créer »
- Choisir « création de requête »
- Choisir « Ajout »

**Remarque :**

*L'ajout de l'enregistrement se fait par défaut à la fin de la table*

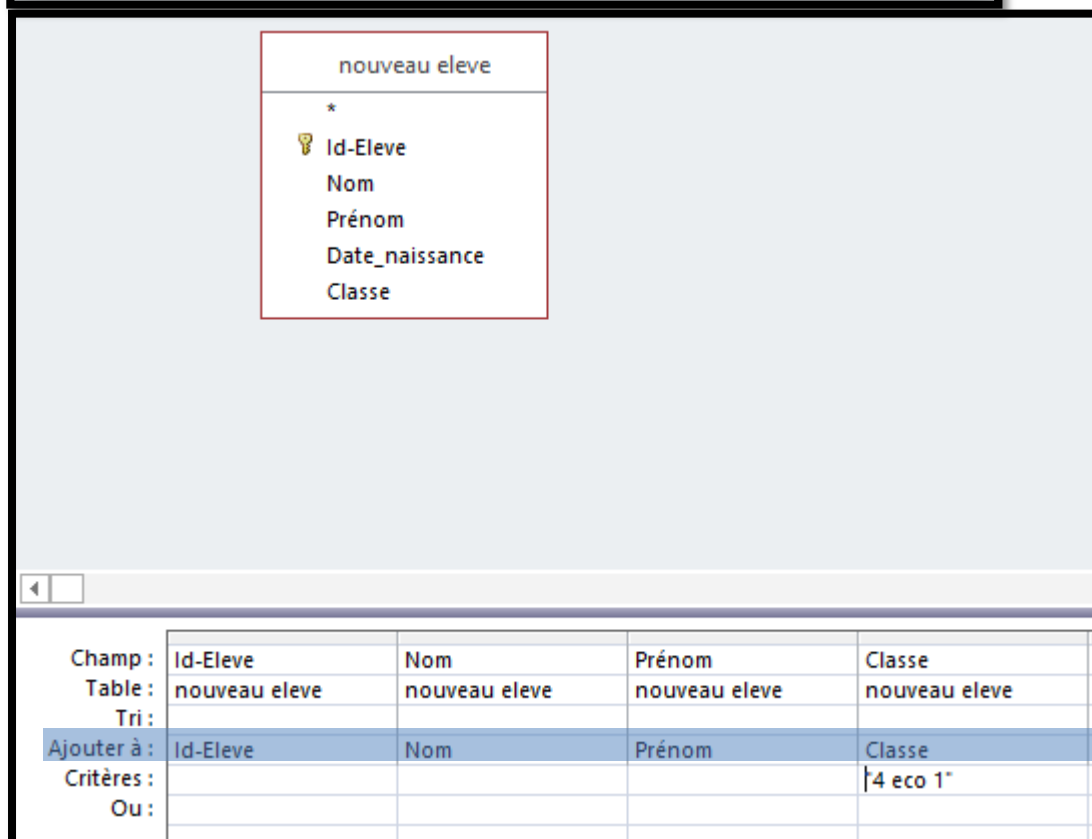
*Si la table contient des contraintes d'ordre, l'ajout se fait dans la bonne position*


**Question 15 :**

Soit la table « nouveau eleve » contenant des élèves provenant d'autres lycées.

ID_ELEVE	Nom	Prénom	Date_nais	Classe
E008	Barrani	Houda	10/08/2004	3 SI 1
E009	Msekni	Imen	20/11/2004	4 ECO 1
E010	Ben Ali	Malik	14/02/2005	2 SI 1
E011	Hedfi	Slim	01/05/2004	4 ECO 1
E012	<b>Ben Slimen</b>	<b>Mouna</b>	<b>25/05/2004</b>	<b>4 L 2</b>

Ajouter les nouveaux élèves qui sont en 4 Eco 1 à la table ELEVE.

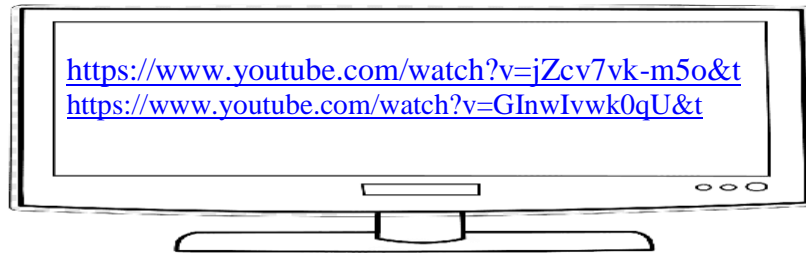



nouveau eleve				
*				
	Id-Eleve	Nom	Prénom	Date_naissance
				Classe

Champ :	Id-Eleve	Nom	Prénom	Classe
Table :	nouveau eleve	nouveau eleve	nouveau eleve	nouveau eleve
Tri :				
Ajouter à :	Id-Eleve	Nom	Prénom	Classe
Critères :				4 eco 1*
Ou :				

Pour plus d'information regarder la vidéo de création des requêtes sur youtube :



#### IV- Importer des données :



---

**Remarque :** dans l'importation ou l'exportation les types de fichiers quand utilisera sont : *texte (\*.txt), excel (\*.xlsx) et csv (\*.csv)*

---

##### 1- A partir d'une feuille Excel :

Importer à la table matière des données de la feuille "matiere" de la du classeur Excel nommé données.xlsx c:/bac2023

- Onglet « Données externes »
- Groupe « Importer »
- Bouton « Excel »
- Dans la fenêtre qui s'affiche, sélectionner la source et la destination des données et le mode et l'emplacement de stockage de données





Données externes - Feuille de calcul Excel

Sélectionner la source et la destination des données

Chemin du classeur

Veuillez spécifier la source de la définition des objets.

Nom fichier : C:\bac2023\ressources\_base note\donnees.xlsx Parcourir...

Spécifiez le mode et l'emplacement de stockage des données dans la base de données active.

**Importer les données sources dans une nouvelle table de la base de données active**  
Si la table spécifiée n'existe pas, Access la créera. Si la table existe déjà, Access risque de remplacer son contenu par des données importées. Les modifications des données sources ne seront pas prises en compte dans la base de données.

**Ajouter une copie des enregistrements à la table :**  Choisir le nom de la table

Si la table spécifiée existe, Access y ajoute les enregistrements. Si la table n'existe pas, Access la crée. Les modifications des données sources ne sont pas prises en compte dans la base de données.

**Lier à la source de données en créant une table attachée**  
Access crée une table qui préserve un lien avec les données sources dans Excel. Les modifications effectuées sur les données sources dans Excel sont répercutées dans la table liée. Toutefois, les données sources ne peuvent pas être modifiées dans Access.

OK Annuler

- Valider par **Ok**
- Sélectionner la feuille de données à importer puis cliquer sur « **suivant** »



Assistant Importation de feuille de calcul

Votre fichier de feuille de calcul contient plusieurs feuilles de calcul ou plages nommées. Laquelle voulez-vous garder ?

Afficher les feuilles de données  
 Afficher les plages nommées

matiere
eleve
note

Choisir le nom de feuille

Données exemples pour la feuille de données "matiere".

ID_matiere	libelle	coefficient
1mat5	Sport	1
3mat6	Histoire	1,5
4mat7	Géographie	1,5
5mat8	Français	1

Annuler < Précédent Suivant > Terminer

Assistant Importation de feuille de calcul

Microsoft Access peut utiliser vos en-têtes de colonnes comme noms de champs pour votre table. La première ligne spécifie-t-elle des en-têtes de colonnes ?

Première ligne contient les en-têtes de colonnes

Cocher cette option :

ID_matiere	libelle	coefficient
1mat5	Sport	1
2mat6	Histoire	1,5
3mat7	Géographie	1,5
4mat8	Français	1

Annuler < Précédent Suivant > Terminer

2- A partir un fichier texte ou CSV :

Pour importer des données d'un fichier texte(txt) ou comma-separated



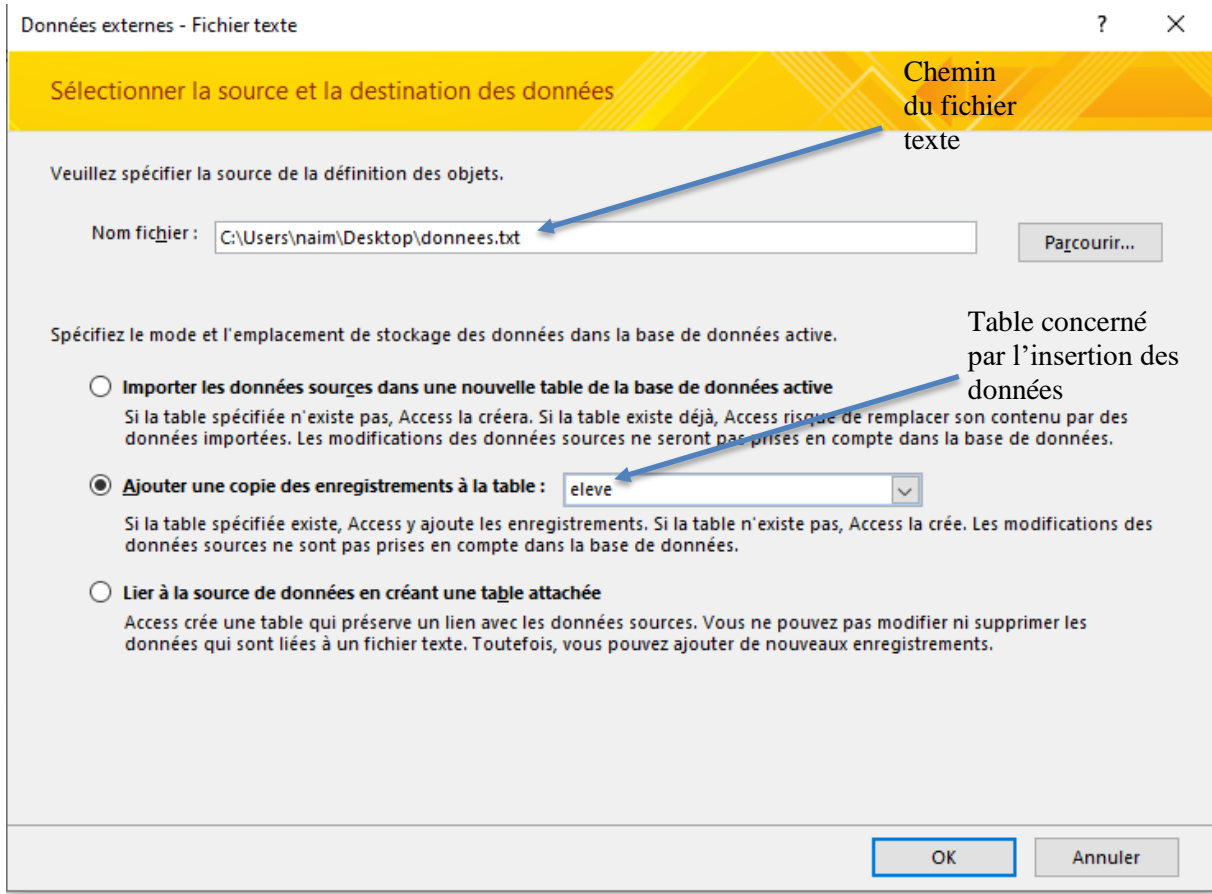
values (csv) dans à une table :

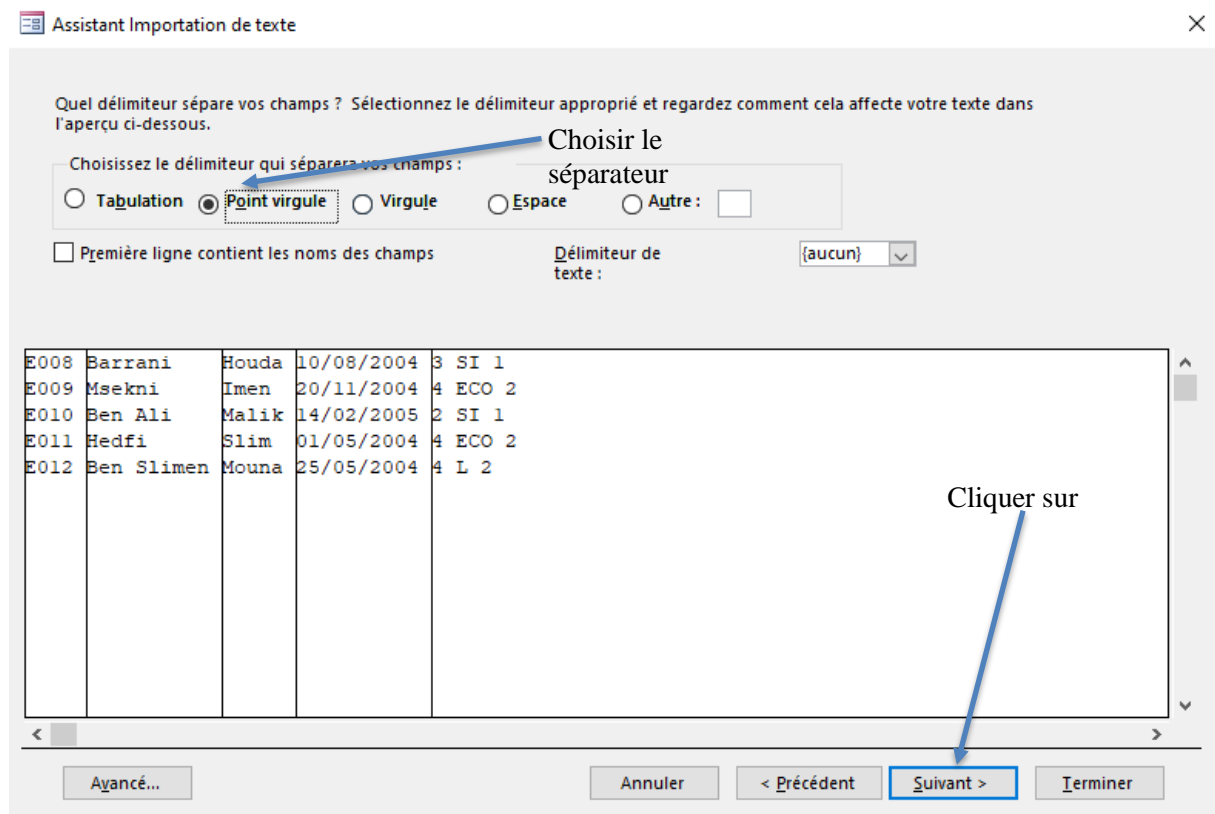
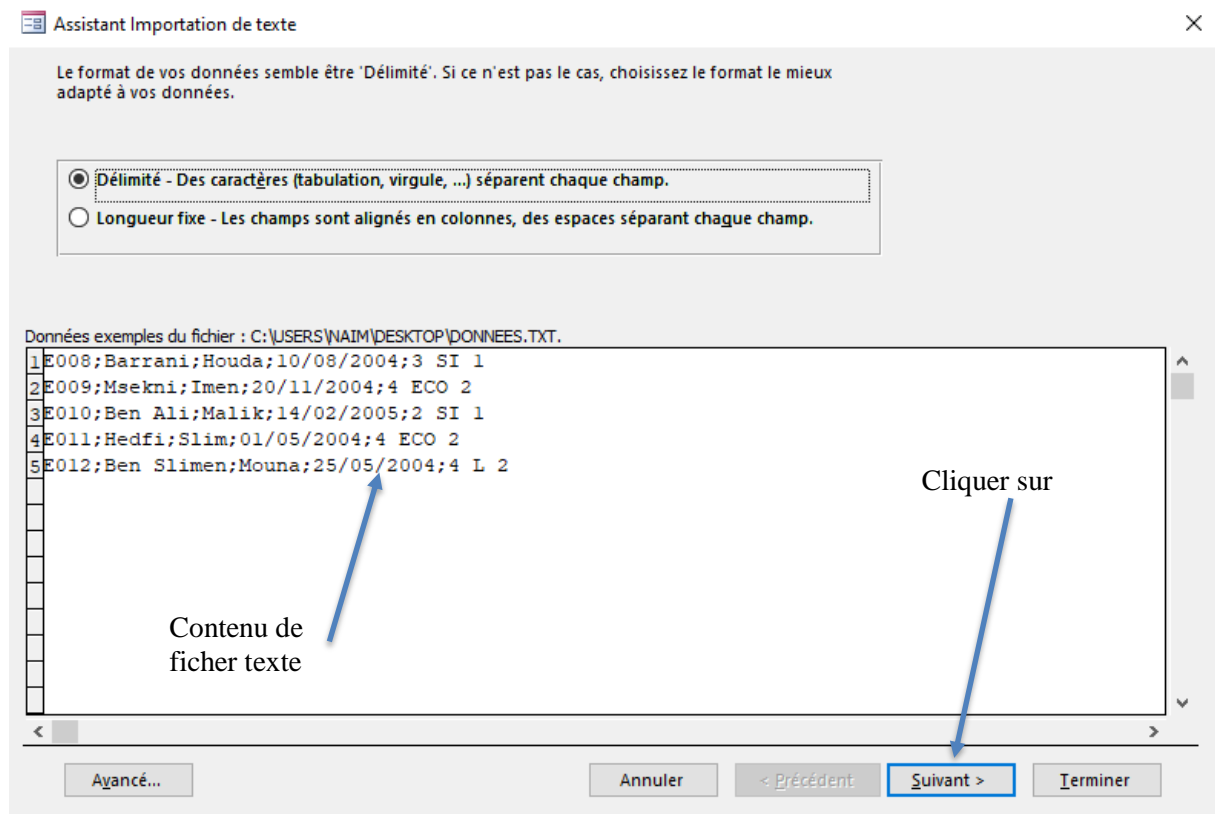
- Onglet « Données externes »
- Choisir « Fichier texte »

**Question :**

Importer les données du fichier nommé données.txt à la table "eleve"

- Séparateur entre les champs c'est le point-virgule ( ;)





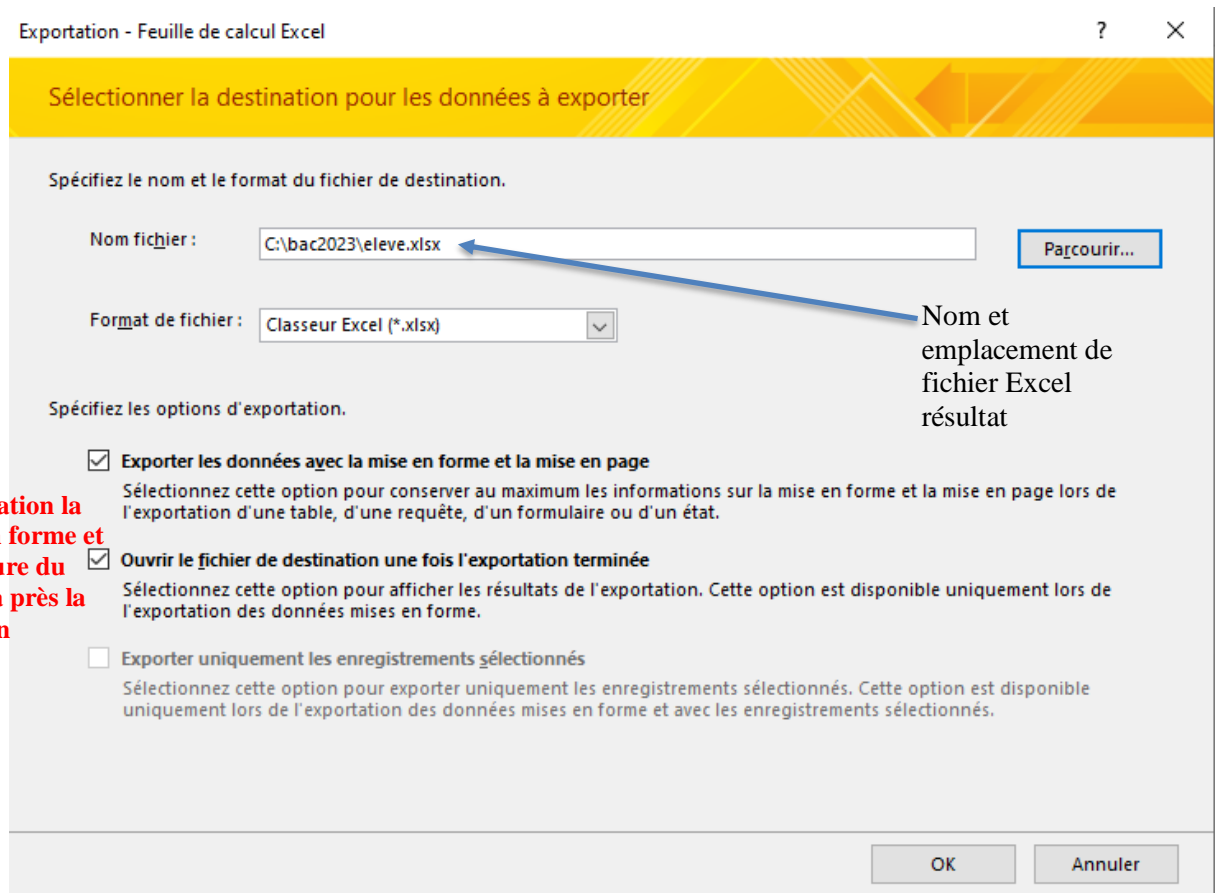
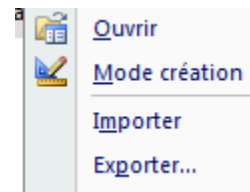
## V- Exportation des données d'une table ou une requête

### 1- Vers une feuille Excel

Exporter les données de la table " élève ", dans votre dossier de travail

du "C:/bac2023" et sous le nom de " élève.xlsx".

- Sélectionner la table ou la requête à exporter
- Dérouler le Menu contextuel (bouton droite de la souris)
- Dans la boîte de dialogue, utilisez le bouton Parcourir pour donner le nom du fichier à créer, et son emplacement.
- Cochez les cases à cocher qui suivent (elles sont explicites).
- Terminez en cliquant sur le bouton **OK**.



Exportation la mise en forme et ouverture du fichier à près la création

Nom et emplacement de fichier Excel résultat

Le fichier Excel est créé à l'emplacement choisi, et si vous avez coché la deuxième option, il est même directement ouvert dans Excel.

## 2- Vers un fichier csv ou txt

Pour exporter une table ou requête à un **Fichier txt ou csv** on doit parcourir par les étapes suivantes:

- Sélectionner la table ou le requête voulu
- Bouton droite de la souris
- Choisir le type de fichier : Fichier texte
- Dans la fenêtre qui s'affiche :
- **Modifier l'extension de fichier dans le cas de fichier csv**

### Question :

Exporter la table "eleve" dans un fichier "eleve.csv" dans le dossier "C:/bac2023" avec :

- Séparateur entre les champs : point-virgule (" ;")
- Symbole décimal : point (".")
- Code des pages : "utf-8".



Exportation - Fichier texte

Sélectionner la destination pour les données à exporter

Modifier l'extension du fichier de txt à csv

Spécifiez le nom et le format du fichier de destination.

Nom fichier :

Spécifiez les options d'exportation.

Exporter les données avec la mise en forme et la mise en page  
Sélectionnez cette option pour conserver au maximum les informations sur la mise en forme et la mise en page lors de l'exportation d'une table, d'une requête, d'un formulaire ou d'un état.

Ouvrir le fichier de destination une fois l'exportation terminée  
Sélectionnez cette option pour afficher les résultats de l'exportation. Cette option est disponible uniquement lors de l'exportation des données mises en forme.

Exporter uniquement les enregistrements sélectionnés  
Sélectionnez cette option pour exporter uniquement les enregistrements sélectionnés. Cette option est disponible uniquement lors de l'exportation des données mises en forme et avec les enregistrements sélectionnés.

OK Annuler

Assistant Exportation de texte

Cet Assistant vous permet de spécifier comment Microsoft Access doit exporter vos données. Quel format d'exportation souhaitez-vous ?

Délimité - Des caractères (tabulation, virgule, ...) séparent chaque champ.

Longueur fixe - Les champs sont alignés en colonnes, des espaces séparant chaque champ.

Format d'exportation exemple :

1	"E001";"Ben salem";"Nizar";9/9/2004 0:00:00;"4 Eco 2"
2	"E002";"Kefi";"Rym";17/3/2005 0:00:00;"4 Eco 2"
3	"E003";"Abrougui";"Amal";3/2/2005 0:00:00;"4 Eco 1"
4	"E004";"Sallami";"Maroua";28/11/2004 0:00:00;"4 Eco 1"
5	"E005";"Choterana";"Abir";27/4/2005 0:00:00;"4 Math 2"
6	"E006";"Abbassi";"Maram";3/12/2004 0:00:00;"4 Eco 3"
7	"E008";"Mizouni";"Nejmaddine";15/8/2004 0:00:00;"4 Math 1"
8	"E009";"Mestiri";"Nassira";14/10/2004 0:00:00;"4 Sciences 1"
9	"E010";"Majdoub";"Wareda";19/1/2004 0:00:00;"4 Eco 2"

Avancé... Annuler < Précédent Suivant > Terminer



Assistant Exportation de texte

Quel délimiteur sépare vos champs ? Sélectionnez le délimiteur approprié et regardez comment cela affecte votre texte dans l'aperçu ci-dessous.

Choisissez le délimiteur qui séparera vos champs :

Tabulation
  Point virgule
  Virgule
  Espace
  Autre :

Inclure les noms des champs sur la première ligne
 Délimiteur de texte :

```

"E001";"Ben salem";"Nizar";9/9/2004 0:00:00;"4 Eco 2"
"E002";"Kefi";"Rym";17/3/2005 0:00:00;"4 Eco 2"
"E003";"Abrougui";"Amal";3/2/2005 0:00:00;"4 Eco 1"
"E004";"Sallami";"Maroua";28/11/2004 0:00:00;"4 Eco 1"
"E005";"Choterana";"Abir";27/4/2005 0:00:00;"4 Math 2"
"E006";"Abbassi";"Maram";3/12/2004 0:00:00;"4 Eco 3"
"E008";"Mizouni";"Nejmaddine";15/8/2004 0:00:00;"4 Math 1"
"E009";"Mestiri";"Nassira";14/10/2004 0:00:00;"4 Sciences 1"
"E010";"Majdoub";"Wareda";19/1/2004 0:00:00;"4 Eco 2"
    
```

Cliquer sur : pour modifier page des codes et symbole décimale

Avancé... Annuler < Précédent Suivant > Terminer

Eleve Spécification d'exportation

Format du fichier :  Délimité  Longueur fixe

Séparateur de champs :  Délimiteur de texte :

Langue : Français Page de codes : Unicode (UTF-8)

Dates, heures et nombres

Ordre de date : JMA  Années (quatre chiffres)  Zéros non significatifs

Délimiteur de date : / Symbole décimal : .

Informations sur le champ :

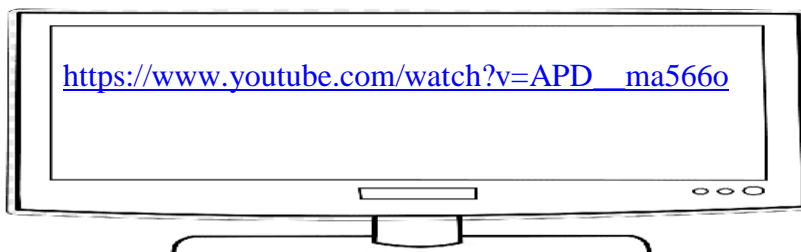
Nom du champ			
Id-Eleve			
Nom			
Prénom			
Date_naissance			
Classe			
*			

Modifier séparateur (;)

Modifier symbole décimal (.)

Modifier page de code : UTF-8

Avancé... Annuler < Précédent Suivant > Terminer



## PROTOTYPE 2021(PARTIE ACCESS)

Afin de gérer les locations des robes, le responsable du service informatique de l'espace **Bella** propose la base de données simplifiée intitulée "**Gestion\_Location**" définie par le schéma textuel suivant :

**Client** (CIN, NomPrenom, DateNais, Adresse, Tel)  
**TypeRobe** (CodeType, Designation)  
**Robe** (CodeRobe, Description, Couleur, CodeType#, Prix)  
**Location** (CIN#, CodeRobe#, DateLocation, DateRetour)

Les données de la table "**Client**" sont représentées comme suit :

Table : Client				
CIN	NomPrenom	DateNais	Adresse	Tel
19458833	Souhir Touzri	29/02/2000	4, Rue d'Alger	(+216) 44222333
01245888	Mehdi Beji	20/09/1945	17, Rue 14 Janvier	(+216) 20304050
12812855	Abla Bkalti	15/10/1996	21, Route Gabes	(+216) 21121314
27755144	Adnen Mestiri	04/09/2002	Im 9, Apt 6. Place Pasteur	(+216) 99887766
05528077	Sofien Nabli	24/01/1973	Place de l'indépendance	(+216) 98877665

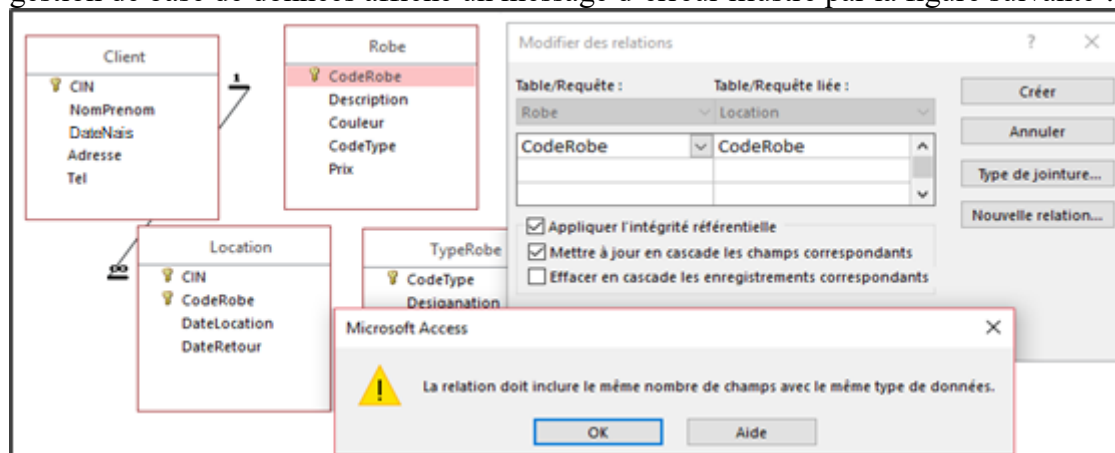
1) Pour chacun des champs de la table "**Client**", encadrer le type approprié à partir de la liste indiquée dans la colonne "**Type de données**" du tableau suivant :

Nom du champ	Type de données
CIN	Texte - Numérique - Date/Heure
NomPrenom	Texte - Numérique - Date/Heure
DateNais	Texte - Numérique - Date/Heure
Adresse	Texte - Numérique - Date/Heure
Tel	Texte - Numérique - Date/Heure

2) En se référant au schéma textuel de la base de données "**Gestion\_Location**", écrire que représente le champ "**CodeType**" pour chacune des tables suivantes :

- ✓ La table "**TypeRobe**" : .....
- ✓ La table "**Robe**" : .....

3) Lors de la création des relations entre les tables de cette base de données, le système de gestion de base de données affiche un message d'erreur illustré par la figure suivante :





Identifier la cause de cette erreur.

.....

.....

.....

4) Soit les grilles des requêtes nommées "R1", "R2", "R3" et "R4" suivantes :

R1			
Champ :	CodeRobe	Description	CodeType
Table :	Robe	Robe	TypeRobe
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :			"T2"
Ou :			

R2			
Champ :	NomPrenom	Tel	DateRetour
Table :	Client	Client	Location
Tri :	Croissant		
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :			#27/04/2021#
Ou :			

R3		
Champ :	Prix	Designation
Table :	Robe	TypeRobe
Mise à jour :	[Prix]*1,05	
Critères :		"Mariée"
Ou :		

R4			
Champ :	Prix	Designation	Couleur
Table :	Robe	TypeRobe	Robe
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :			[Donner couleur :]
Ou :			

Pour chacune des propositions ci-dessous, préciser le nom et le type de la requête.

a. Pour afficher par ordre **croissant** des noms et prénoms des clients qui ont retourné, le 27/04/2021, les robes qu'ils ont louées.

Nom de la requête :

Type de la requête :

b. Pour augmenter de **5%** le prix des robes de type "**Mariée**".

Nom de la requête :

Type de la requête :

c. Pour afficher le **code** et la **description** des robes dont le code type est "**T2**".

Nom de la requête :

Type de la requête :



d. Pour afficher la liste des **prix** et les **désignations** des robes pour une couleur donnée.  
Nom de la requête :

Type de la requête :

5) Compléter la grille de création de la requête **R5** suivante :

**R5:** Affiche les informations (**NomPrenom, Description, Couleur, Prix, DateLocation, DateRetour**) relatives aux clients ayant loué des robes en 2021.

<i>Champs :</i>						
<i>Table:</i>						
<i>Tri :</i>						
<i>Afficher:</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<i>Critères:</i>						
<i>Ou:</i>						

**CORRECTION PROTOTYPE 2021 :PARTIE ACCESS)**

Afin de gérer les locations des robes, le responsable du service informatique de l'espace **Bella** propose la base de données simplifiée intitulée "**Gestion\_Location**" définie par le schéma textuel suivant :

**Client** (CIN, NomPrenom, DateNais, Adresse, Tel)  
**TypeRobe** (CodeType, Designation)  
**Robe** (CodeRobe, Description, Couleur, CodeType#, Prix)  
**Location** (CIN#, CodeRobe#, DateLocation, DateRetour)

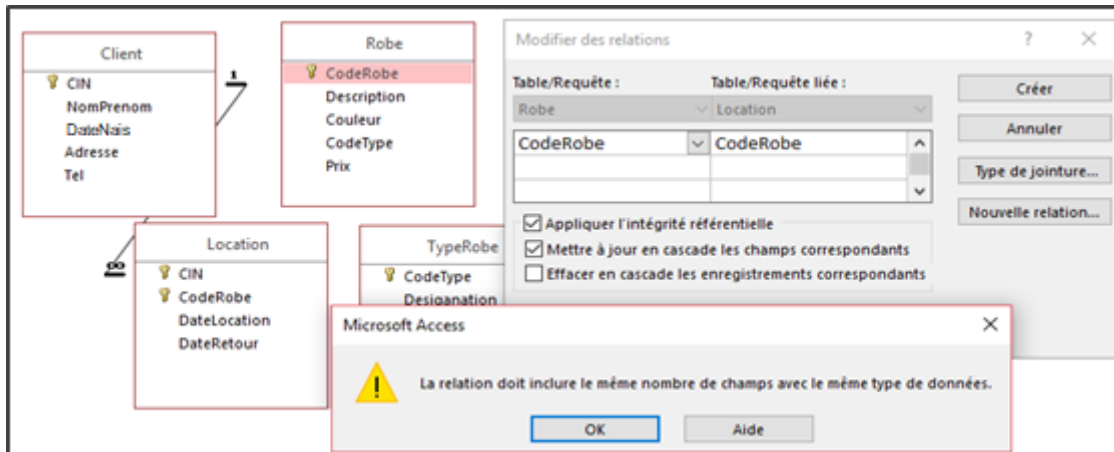
Les données de la table "**Client**" sont représentées comme suit :

Table : Client				
CIN	NomPrenom	DateNais	Adresse	Tel
19458833	Souhir Touzri	29/02/2000	4, Rue d'Alger	(+216) 44222333
01245888	Mehdi Beji	20/09/1945	17, Rue 14 Janvier	(+216) 20304050
12812855	Abla Bkalti	15/10/1996	21, Route Gabes	(+216) 21121314
27755144	Adnen Mestiri	04/09/2002	Im 9, Apt 6. Place Pasteur	(+216) 99887766
05528077	Sofien Nabli	24/01/1973	Place de l'indépendance	(+216) 98877665

- 1- Pour chacun des champs de la table "**Client**", encadrer le type approprié à partir de la liste indiquée dans la colonne "**Type de données**" du tableau suivant :

Nom du champ	Type de données
CIN	Texte - Numérique - Date/Heure
NomPrenom	Texte - Numérique - Date/Heure
DateNais	Texte - Numérique - Date/Heure
Adresse	Texte - Numérique - Date/Heure
Tel	Texte - Numérique - Date/Heure

- 2- En se référant au schéma textuel de la base de données "**Gestion\_Location**", écrire que représente le champ "**CodeType**" pour chacune des tables suivantes :
- ✓ La table "**TypeRobe**" : **clé primaire**
  - ✓ La table "**Robe**" : **clé étrangère**
- 3- Lors de la création des relations entre les tables de cette base de données, le système de gestion de base de données affiche un message d'erreur illustré par la figure suivante :



Identifier la cause de cette erreur.

**Les deux champs CIN de la table client et CIN de la table location doivent être de même type et propriété pour qu'on peut créer une relation entre ces deux champs.**

4- Soit les grilles des requêtes nommées "R1", "R2", "R3" et "R4" suivantes :

R1			
Champ :	CodeRobe	Description	CodeType
Table :	Robe	Robe	TypeRobe
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :			"T2"
Ou :			

R2			
Champ :	NomPrenom	Tel	DateRetour
Table :	Client	Client	Location
Tri :	Croissant		
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :			#27/04/2021#
Ou :			

R3	
Champ :	Prix
Table :	Robe
Mise à jour :	[Prix]*1,05
Critères :	"Mariée"
Ou :	

R4			
Champ :	Prix	Desigation	Couleur
Table :	Robe	TypeRobe	Robe
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :			[Donner couleur :]
Ou :			

Pour chacune des propositions ci-dessous, préciser le nom et le type de la requête.

- a- Pour afficher par ordre **croissant** des noms et prénoms des clients qui ont retourné, le **27/04/2021**, les robes qu'ils ont louées.



Nom de la requête : **R2**

Type de la requête : **Sélection**

b- Pour augmenter de **5%** le prix des robes de type "**Mariée**".

Nom de la requête : **R3**

Type de la requête : **Mise à jour**

c- Pour afficher le **code** et la **description** des robes dont le code type est "**T2**".

Nom de la requête : **R1**

Type de la requête : **sélection**

d- Pour afficher la liste des **prix** et les **désignations** des robes pour une couleur donnée.

Nom de la requête : **R4**

Type de la requête : **sélection paramétrée**

e- Compléter la grille de création de la requête **R5** suivante :

**R5:**

Affiche les informations (**NomPrenom**, **Description**, **Couleur**, **Prix**, **DateLocation**, **DateRetour**) relatives aux clients ayant loué des robes en 2021.

*Champs :*

*Table:*

*Tri :*

*Afficher:*

*Critères:*

*Ou:*

<b>NomPrenom</b>	<b>Description</b>	<b>Couleur</b>	<b>Prix</b>	<b>DateLocation</b>	<b>DateRetour</b>
Client	typeRobe	robe	Robe	location	Location
X	X	X	X	X	X
				>=#01/01/2021# ET <=#31/12/2021#	

## B- ANALYSE DE DONNEES

### I- introduction :

L'analyse de données est un domaine issu du monde des statistiques qui vise à faire le lien entre les différentes données statistiques pour les classer, les décrire et les analyser de manière brève.

L'objectif de l'analyse des données est d'extraire une information statistique qui permet de cerner plus précisément le profil de la donnée. Les résultats obtenus permettent ensuite d'optimiser la stratégie de la société en question en ajustant certains points.

Pour analyser les données en Python, on utilise la bibliothèque Pandas. Cette bibliothèque doit être initialement importée. Puis, les données à analyser sont initialement stockées dans **un DataFrame** (un tableau de données). Pandas offre diverses fonctionnalités pour :

- Charger les données dans **un DataFrame**
- Explorer les données (obtention d'informations/filtrage)
- Manipuler les données (Ajouter/Mettre à jour/Supprimer des données)
- Trier selon un ou plusieurs critères dans un ordre croissant ou décroissant

Les données à analyser sont chargées, souvent, à **partir de fichiers ayant divers formats (Excel (.xlsx) ; texte (.csv ou .txt))** dans **un DataFrame**. On peut aussi, dans certains cas, créer, explicitement **un DataFrame**.

**Un DataFrame c'est un tableau de données à deux dimensions dont les lignes et colonnes sont indexées (en commençant par 0).**

### II- importation de la bibliothèque pandas :

Pour importer une bibliothèque dans Python il existe plusieurs écritures :

Méthode	Importation
1	<code>from pandas import *</code>
Exemple appel avec le commande <code>read_csv</code>	
<code>read_csv</code>	
2	<code>import pandas</code>
Exemple	
<code>Pandas.read_csv</code>	
3	<code>import pandas as ps</code>
Exemple	
<code>Ps.read_csv</code>	

### III- Chargement des données d'un DataFrame : Avec le mot `id_DataFrame` : nom de la DataFrame

#### 1- Création d'un DataFrame :

Question :

Créer un DataFrame nommé "**matiere**" à partir du tableau de données ci-dessous :

ID_matiere	Nom_matiere	coefficient
mat1	informatique	1
mat2	économie	3
mat3	Math	2

Pour créer un DataFrame à partir un ensemble de données on utilise la fonction **DataFrame()** qui a la forme générale :



```
#importation de la bibliothèque pandas
from pandas import *
Id_DataFrame=DataFrame({
    "id_colonne1": [valeur1,...,Valeurn],
    ... ,
    "id_colonneN": [valeur1,...,valeurn]
})
```



**Remarque :**

*Lorsqu'une colonne contient un type texte court il faut mettre les données entre deux guillemets ("").*

*Exemple les deux colonnes `ID_matiere` et `Nom_matiere` sont de type texte court.*

```
#importation des bibliothéque pandas
from pandas import *
matiere=DataFrame({
    "ID_matiere":["mat1","mat2","mat3"],
    "Nom_matiere":["informatique","économie","coefficient"],
    "coefficient":[1,3,2]
})
#affichage de la DataFrame
print(matiere)
```

Le résultat affiché : création d'un DataFrame matiere

	ID_matiere	Nom_matiere	coefficient
0	mat1	informatique	1
1	mat2	économie	3
2	mat3	coefficient	2

## 2- Importation d'un DataFrame :

On peut charger des données dans un DataFrame à partir soit :

- Un fichier Excel (xlsx)
- Un fichier csv(csv)
- Un fichier texte (txt)

### a- A partir d'un fichier Excel

Pour importer des données à partir d'un fichier Excel (xlsx) on utilise la fonction `read_excel()` et dans la forme générale est la suivante :



```
#importation de la bibliothèque pandas
from pandas import *
Id_DataFrame=read_excel("Chemin / Nom_fichier.xls","Nom_Feuille")
```

### Question :

Importer les données de la feuille **notes** du classeur nommé **note** qui se trouve dans votre dossier de travail (c:\bac2023\numero) dans un DataFrame nommé **noteexcel** et l'afficher.





```
#importation des bibliothèque pandas
from pandas import *
noteexcel = read_excel ("note.xlsx","notes")
#affichage de la DataFrame
print(noteexcel)
```

### Le résultat affiché : DataFrame noteexcel

	Nom	prenom	id_eleve	libelle	note
0	Ben	salah	Sami	ele002	gestion 17.0
1	Ben	salah	Sami	ele002	Math 19.5
2	Ben	salah	Sami	ele002	economie 16.5
3	Ben	salah	Sami	ele002	informatique 18.0
4	hammi	Hela	Hela	ele003	gestion 17.5
5	hammi	Hela	Hela	ele003	Math 15.5
6	hammi	Hela	Hela	ele003	economie 18.5
7	hammi	Hela	Hela	ele003	informatique 17.5
8	Sassi	Ali	Ali	ele001	gestion 13.5
9	Sassi	Ali	Ali	ele001	Math 15.0
10	Sassi	Ali	Ali	ele001	economie 5.0
11	Sassi	Ali	Ali	ele001	informatique 19.5

### b- A partir d'un fichier csv

Pour importer des données à partir d'un fichier csv (csv) on utilise la fonction `read_csv()`, elle a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
Id_DataFrame = pd.read_csv("Chemin / Nom_fichier.csv", sep="son symbole", decimal="son symbole")
```

### Question :

Importer dans un nouveau DataFrame nommé `notecsv` les données du fichier `note.csv` et afficher avec :

- Séparateur : le symbole ";"
- Symbole décimal "."



- Chemin : ton dossier de travail c:\bac2023\numero

```
#importation des bibliothéque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
#affichage de la DataFrame
print(notecsv)
```

### Le résultat affiché : DataFrame notecsv

	Nom	prenom	id_eleve	libelle	note
0	Ben	salah	Sami	ele002	gestion 17.0
1	Ben	salah	Sami	ele002	Math 19.5
2	Ben	salah	Sami	ele002	economie 16.5
3	Ben	salah	Sami	ele002	informatique 18.0
4	hammi	Hela	ele003	gestion	17.5
5	hammi	Hela	ele003	Math	15.5
6	hammi	Hela	ele003	economie	18.5
7	hammi	Hela	ele003	informatique	17.5
8	Sassi	Ali	ele001	gestion	13.5
9	Sassi	Ali	ele001	Math	15.0
10	Sassi	Ali	ele001	economie	5.0
11	Sassi	Ali	ele001	informatique	19.5

### c- A partir d'un fichier texte

Pour importer des données à partir d'un fichier texte (txt) on utilise la fonction `read_table()`, elle a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
Id_DataFrame = pd.read_table("Chemin/Nom_fichier.txt",sep="son symbole", encoding="code")
```

### Question :

Importer dans un nouveau DataFrame nommé **notetxt** les données du fichier note.txt et afficher avec :

- Séparateur : le symbole ";"
- Page de code : utf-8
- Chemin : ton dossier de travail c:\bac2023\numero

```
#importation des bibliothèque pandas
from pandas import *
notetxt = read_table("note.txt",sep=";",encoding="utf-8")
#affichage de la DataFrame
print("affichage des données du la DataFrame notetxt")
print(notetxt)
```

```
affichage des données du la DataFrame notetxt
  id_eleve id_matiere  note
0   ele001      mat1  19.5
1   ele001      mat2   5.0
2   ele001      mat3  15.0
3   ele001      mat4  13.5
4   ele002      mat1  18.0
5   ele002      mat2  16.5
6   ele002      mat3  19.5
7   ele002      mat4  17.0
8   ele003      mat1  17.5
9   ele003      mat2  18.5
10  ele003      mat3  15.5
11  ele003      mat4  17.5
```

---

**Remarque :**

Les deux  fichiers Python et le fichier importation doivent être dans le même dossier de travail.

---



## IV- Information d'un DataFrame :

	Nom	prenom	id_eleve	libelle	note
0	Ben	salah	Sami	ele002	gestion 17.0
1	Ben	salah	Sami	ele002	Math 19.5
2	Ben	salah	Sami	ele002	economie 16.5
3	Ben	salah	Sami	ele002	informatique 18.0
4	hammi	Hela	ele003	gestion	17.5
5	hammi	Hela	ele003	Math	15.5
6	hammi	Hela	ele003	economie	18.5
7	hammi	Hela	ele003	informatique	17.5
8	Sassi	Ali	ele001	gestion	13.5
9	Sassi	Ali	ele001	Math	15.0
10	Sassi	Ali	ele001	economie	5.0
11	Sassi	Ali	ele001	informatique	19.5

Figure 1: contenu du fichier notecsv

Avec : `id_DataFrame` : Nom du DataFrame

### 1- Dimension d'un DataFrame :

Pour avoir les dimensions d'un DataFrame (nombre de ligne et de colonne) on utilise la fonction **shape** qui a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
id_DataFrame.shape
```

#### Question :

Afficher les dimensions du DataFrame "notecsv"

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
#affichage de la DataFrame
print("affichage des dimensions du la DataFrame notecsv")
print(notecsv.shape)
```

```
affichage des dimensions du la DataFrame notecsv
```

```
(12, 5)
```

12 : nombre de lignes  
5 : nombre de colonnes

## 2- Taille d'un DataFrame :

Pour avoir la taille d'un DataFrame (nombre de ligne et de colonne) on utilise la fonction **size** et elle a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
id_DataFrame.size
```

### Question :

Afficher la taille du DataFrame "notecsv"

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
#affichage de la DataFrame
print("affichage des dimensions du la DataFrame notecsv")
print(notecsv.shape)
print("affichage de la taille du la DataFrame notecsv")
print(notecsv.size)
```

```
affichage des dimensions du la DataFrame notecsv
(12, 5)
affichage de la taille du la DataFrame notecsv
60 = 12 lignes et 5 colonnes
```

## 3- Informations relatives d'un DataFrame :

Pour afficher des informations sur les colonnes d'un DataFrame leurs comptes leurs types, on utilise la fonction **info()** , elle a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
print(id_DataFrame.info())
```

### Question :

Afficher les informations sur les colonnes du DataFrame "notecsv"



```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
#affichage de la DataFrame
print("affichage des informations relatives des colonnes")
print(notecsv.info())
```

```
affichage des informations relatives des colonnes
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12 entries, 0 to 11
Data columns (total 5 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Nom          12 non-null    object
1   prenom       12 non-null    object
2   id_eleve     12 non-null    object
3   libelle      12 non-null    object
4   note         12 non-null    float64
dtypes: float64(1), object(4)
memory usage: 608.0+ bytes
None
```

#### 4- Statistiques descriptives d'un DataFrame :

Pour avoir statistiques descriptives d'un DataFrame un résumé rapide on utilise la fonction `describe()` , elle a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
print(id_DataFrame.describe())
```

#### Question :

Afficher un résumé rapide du DataFrame "notecsv".



```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
#affichage de la DataFrame
print("affichage d'un résumé rapide")
print(notecsv.describe())
```

```
affichage d'un résumé rapide
              note
count      12.000000
mean       16.083333
std         3.918681
min         5.000000
25%        15.375000
50%        17.250000
75%        18.125000
max        19.500000
```

#### 5- Les noms des colonnes d'un DataFrame :

Pour afficher les noms de colonnes d'un DataFrame on utilise la fonction **columns**, elle a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
print(id_DataFrame.columns)
```

#### Question :

Afficher les noms des colonnes du DataFrame "notecsv"

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("affichage des noms de colonnes ")
print(notecsv.columns)
```

```
affichage des noms de colonnes
Index(['Nom', 'prenom', 'id_eleve', 'libelle', 'note'], dtype='object')
```

## V- Manipulation d'un DataFrame :

### 1- Renommage d'une ou plusieurs colonnes d'un DataFrame :

Pour renommer une ou plusieurs colonnes d'un DataFrame on utilise la fonction **rename()**, qui a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *

Id_DataFrame = Id_DataFrame.rename(columns={
"colonne1": "nouveau_nom_colonne1",
...,
"colonneN": "nouveau_nom_colonneN"
})
```

### Question :

Renommer la colonne "libelle" en " nom matiere " du DataFrame notecsv

```
#importation des bibliothèque pandas
from pandas import *
notecsv=notecsv.rename(columns={"libelle": "Nom matiere"})
print(notecsv)
```

affichage du DataFrame après changement du nom de la colonne libelle

	Nom	prenom	id_eleve	nom matiere	note
0	Ben	salah	Sami	ele002	gestion 17.0
1	Ben	salah	Sami	ele002	Math 19.5
2	Ben	salah	Sami	ele002	economie 16.5
3	Ben	salah	Sami	ele002	informatique 18.0
4	hammi	Hela	ele003	gestion	17.5
5	hammi	Hela	ele003	Math	15.5
6	hammi	Hela	ele003	economie	18.5
7	hammi	Hela	ele003	informatique	17.5
8	Sassi	Ali	ele001	gestion	13.5
9	Sassi	Ali	ele001	Math	15.0
10	Sassi	Ali	ele001	economie	5.0
11	Sassi	Ali	ele001	informatique	19.5

Nom de la colonne a changée

### 2- Supprimer des lignes d'un DataFrame :

Pour supprimer une ou plusieurs lignes d'un DataFrame on utilise la fonction :

**drop ([numéro de ligne]), elle a la forme générale suivante :**



```
#importation de la bibliothèque pandas
from pandas import *

Id_DataFrame=id_DataFrame.drop([indice_ligne])
```





## Question :

Supprimer la première ligne du **DataFrame "notecsv"**

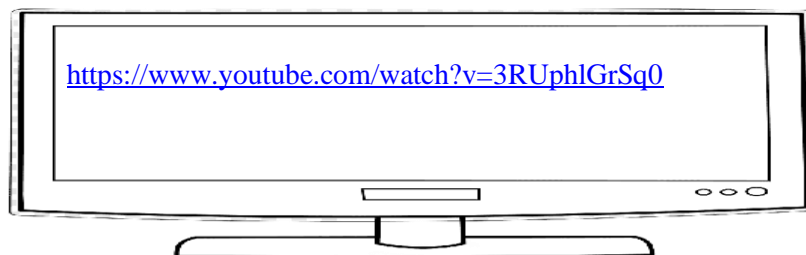
```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
notecsv=notecsv.drop([0])
print("affichage du DataFrame après la suppression de la première ligne ")
print(notecsv)
```

```
affichage du DataFrame après la suppression de la première ligne
      Nom prenom id_eleve      libelle  note
1   Ben salah   Sami  ele002      Math  19.5
2   Ben salah   Sami  ele002  economie  16.5
3   Ben salah   Sami  ele002  informatique  18.0
4   hammi      Hela  ele003      gestion  17.5
6   hammi      Hela  ele003  economie  18.5
7   hammi      Hela  ele003  informatique  17.5
8   Sassi      Ali  ele001      gestion  13.5
9   Sassi      Ali  ele001      Math  15.0
10  Sassi      Ali  ele001  economie  5.0
11  Sassi      Ali  ele001  informatique  19.5
```

## Question :

Supprimer ligne N°2et la ligne N°5 du **DataFrame "notecsv"**

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
notecsv=notecsv.drop([3,5])
print(notecsv)
```





	Nom	prenom	id_eleve	libelle	note
0	Ben salah	Sami	ele002	gestion	17.0
1	Ben salah	Sami	ele002	Math	19.5
2	Ben salah	Sami	ele002	economie	16.5
4	hammi	Hela	ele003	gestion	17.5
6	hammi	Hela	ele003	economie	18.5
7	hammi	Hela	ele003	informatique	17.5
8	Sassi	Ali	ele001	gestion	13.5
9	Sassi	Ali	ele001	Math	15.0
10	Sassi	Ali	ele001	economie	5.0
11	Sassi	Ali	ele001	informatique	19.5

### 3- Supprimer des colonnes d'un DataFrame :

Pour supprimer une ou plusieurs colonnes d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *
```

```
Id_DataFrame=Id_DataFrame.drop(columns=["nom_col_1", ..., "nom_col_N"])
```

#### Question :

Supprimer la colonne nommé "**prenom**" du DataFrame "notecsv" et afficher le DataFrame après modification

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
notecsv=notecsv.drop(columns=["prenom"])
print("affichage du DataFrame après la suppression de la colonne prenom ")
print(notecsv)
```

```
affichage du DataFrame après la suppression de la colonne prenom
   Nom id_eleve libelle note
0  Ben salah  ele002  gestion 17.0
1  Ben salah  ele002   Math 19.5
2  Ben salah  ele002  economie 16.5
3  Ben salah  ele002  informatique 18.0
4  hammi     ele003  gestion 17.5
5  hammi     ele003   Math 15.5
6  hammi     ele003  economie 18.5
7  hammi     ele003  informatique 17.5
8  Sassi     ele001  gestion 13.5
9  Sassi     ele001   Math 15.0
10 Sassi     ele001  economie 5.0
11 Sassi     ele001  informatique 19.5
```

**Question :**

Supprimer les colonne "Nom" et "prenom" de la DataFrame "notecsv" et afficher le DataFrame après modification

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
notecsv=notecsv.drop(columns=["Nom", "prenom"])
print(notecsv)
```

	id_eleve	libelle	note
0	ele002	gestion	17.0
1	ele002	Math	19.5
2	ele002	economie	16.5
3	ele002	informatique	18.0
4	ele003	gestion	17.5
5	ele003	Math	15.5
6	ele003	economie	18.5
7	ele003	informatique	17.5
8	ele001	gestion	13.5
9	ele001	Math	15.0
10	ele001	economie	5.0
11	ele001	informatique	19.5

**4- Ajouter une ou plusieurs colonnes d'un DataFrame :**

Pour ajouter une ou plusieurs colonnes d'un DataFrame on utilise la fonction qui a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
```

```
Id_DataFrame["id_Colonne"]=[liste des valeurs]
```

**Question :**

Ajouter la colonne coefficient au DataFrame "notecsv" dont les valeurs sont : Coefficient = [3,3,3,1,3,3,3,1,3,3,,3,1,3,3,3,1] et afficher le DataFrame



```

affichage du DataFrame après l'ajout d'une nouvelle colonne
  Nom prenom id_eleve libelle note coefficient
0 Ben salah Sami ele002 gestion 17.0 3
1 Ben salah Sami ele002 Math 19.5 3
2 Ben salah Sami ele002 economie 16.5 3
3 Ben salah Sami ele002 informatique 18.0 1
4 hammi Hela ele003 gestion 17.5 3
5 hammi Hela ele003 Math 15.5 3
6 hammi Hela ele003 economie 18.5 3
7 hammi Hela ele003 informatique 17.5 1
8 Sassi Ali ele001 gestion 13.5 3
9 Sassi Ali ele001 Math 15.0 3
10 Sassi Ali ele001 economie 5.0 3
11 Sassi Ali ele001 informatique 19.5 1

```

### 5- Modifier la valeur d'une cellule d'un DataFrame

Pour modifier le contenu d'une cellule d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *
```

```
Id_DataFrame.loc[N° ligne,"Nom_colonne"]=valeur
```

#### Question :

Changer la note de matière "**gestion**" l'élève "Ben salah Sami" à 17.5

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
notecsv.loc[0,"note"]=17.5
print("affichage du résultat après modification")
print(notecsv)
```

```
affichage du résultat après modification
```

	Nom	prenom	id_eleve	libelle	note
0	Ben salah	Sami	ele002	gestion	17.5
1	Ben salah	Sami	ele002	Math	19.5
2	Ben salah	Sami	ele002	economie	16.5
3	Ben salah	Sami	ele002	informatique	18.0
4	hammi	Hela	ele003	gestion	17.5
5	hammi	Hela	ele003	Math	15.5
6	hammi	Hela	ele003	economie	18.5
7	hammi	Hela	ele003	informatique	17.5
8	Sassi	Ali	ele001	gestion	13.5
9	Sassi	Ali	ele001	Math	15.0
10	Sassi	Ali	ele001	economie	5.0
11	Sassi	Ali	ele001	informatique	19.5

#### 6- Modifier une ligne d'un DataFrame :

Pour modifier les données d'une ligne d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *
```

```
Id_DataFrame.loc[N° ligne]=[liste_ valeur]
```

#### Question :

Modifier la première ligne par les données suivantes :

- Nom : Hajy
- Prénom : Selem
- Id\_elev : ele004
- Libelle : sport
- Note : 19



```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
notecsv.loc[0]=["Haji","selem","ele004","sport",19]
print("affichage du résultat après modification de la ligne zéro")
print(notecsv.loc[0])
```

```
affichage du résultat après modification de la ligne zéro
Nom          Haji
prenom       selem
id_eleve     ele004
libelle      sport
note         19.0
Name: 0, dtype: object
```

## VI. Affichage des données d'un DataFrame :

### 1- Affichage des données d'un DataFrame :

La fonction **print** permet d'afficher sur l'écran

Pour afficher les données d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *

print(Id_DataFrame)
```

#### Question :

Afficher les données du DataFrame "notecsv"

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print(notecsv)
```

```
      Nom prenom id_eleve libelle note
0  Ben salah Sami ele002 gestion 17.0
1  Ben salah Sami ele002 Math 19.5
2  Ben salah Sami ele002 economie 16.5
3  Ben salah Sami ele002 informatique 18.0
4  hammi Hela ele003 gestion 17.5
5  hammi Hela ele003 Math 15.5
6  hammi Hela ele003 economie 18.5
7  hammi Hela ele003 informatique 17.5
8  Sassi Ali ele001 gestion 13.5
9  Sassi Ali ele001 Math 15.0
10 Sassi Ali ele001 economie 5.0
11 Sassi Ali ele001 informatique 19.5
```

**a. Affichage du contenu d'une colonne d'un DataFrame :**

Pour afficher les données d'une colonne d'un DataFrame on utilise la commande suivante :



```
#importation de la bibliothèque pandas
from pandas import *

print(Id_DataFrame["nom_colonne"])
# ou bien
print(Id_DataFrame.nom_colonne)
# pour afficher plusieurs colonnes
print([[id_colonne1,id_colonne2...]])
```

**Question :**

Afficher les données de la colonne "note" du DataFrame "notecsv"

```
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("le contenu de la colonne note du DataFrame notecsv")
print(notecsv["note"])
```

```
le contenu de la colonne note du DataFrame notecsv
0      17.0
1      19.5
2      16.5
3      18.0
4      17.5
5      15.5
6      18.5
7      17.5
8      13.5
9      15.0
10     5.0
11     19.5
```

**Question :**

Afficher les deux colonnes "nom" et "note" du DataFrame "notecsv"



```
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("le contenu 2 colonnes Nom et note du DataFrame notecsv")
print(notecsv[["Nom","note"]])
```

```
le contenu 2 colonnes nom et note du DataFrame notecsv
      Nom  note
0  Ben salah  17.0
1  Ben salah  19.5
2  Ben salah  16.5
3  Ben salah  18.0
4    hammi  17.5
5    hammi  15.5
6    hammi  18.5
7    hammi  17.5
8    Sassi  13.5
9    Sassi  15.0
10   Sassi   5.0
11   Sassi  19.5
```

## 2- Affichage des données d'une ligne d'un DataFrame :

Pour afficher les données d'une ligne d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *

print(Id_DataFrame.iloc[numero_ligne])
```

### Question :

Afficher les données de la 4<sup>ème</sup> ligne du DataFrame "notecsv"

```
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("le contenu de la 4ème ligne du DataFrame notecsv")
print(notecsv.iloc[3])
```

```
le contenu de la 4ème ligne du DataFrame notecsv
Nom          Ben salah
prenom          Sami
id_eleve      ele002
libelle      informatique
note          18.0
Name: 3, dtype: object
```



### 3- Affichage le contenu d'une cellule d'un DataFrame :

Pour afficher les données d'une ligne d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *
```

```
print(Id_DataFrame.iloc[numero_ligne,numero_colonne])
```

#### Question :

Afficher les données de la cellule ligne 3 colonne 5

```
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("le contenu de la cellule ligne 3 colonne 4 DataFrame notecsv")
print(notecsv.iloc[3,4])
```

```
le contenu de la cellule ligne 3 colonne 4 DataFrame notecsv
18.0
```

### 4- Affichage n premières lignes d'un DataFrame :

Pour afficher les n premières lignes d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *
```

```
print(Id_DataFrame.head(n))
```

#### Question :

Afficher les 5 premières lignes du DataFrame "notecsv"

```
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("les cinq premiers lignes du DataFrame notecsv")
print(notecsv.head(5))
```

```
les cinq premiers lignes du DataFrame notecsv
      Nom prenom id_eleve      libelle  note
0  Ben salah      Sami  ele002      gestion  17.0
1  Ben salah      Sami  ele002           Math  19.5
2  Ben salah      Sami  ele002      economie  16.5
3  Ben salah      Sami  ele002  informatique  18.0
4      hammi      Hela  ele003      gestion  17.5
```



### 5- Affichage n dernières lignes d'un DataFrame :

Pour afficher les n derniers lignes d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *
```

```
print(Id_DataFrame.tail(n))
```

#### Question :

Afficher les 4 dernières lignes du DataFrame "notecsv"

```
from pandas import *
notecsv = read_csv("note.csv", sep=";", decimal=".")
print("les quatre dernier lignes du DataFrame notecsv")
print(notecsv.tail(4))
```

```
les quatre dernier lignes du DataFrame notecsv
      Nom prenom id_eleve      libelle  note
8   Sassi      Ali  ele001      gestion 13.5
9   Sassi      Ali  ele001         Math 15.0
10  Sassi      Ali  ele001      economie  5.0
11  Sassi      Ali  ele001  informatique 19.5
```



**Remarque :** les fonctions `head()` et `tail()` sans valeur entre parenthèse affichent respectivement les 5 premières et les 5 dernières lignes du DataFrame.

### 6- Affichage d'une plage de données d'un DataFrame :

Pour afficher une plage donnée d'une colonne d'un DataFrame :



```
#importation de la bibliothèque pandas
from pandas import *
```

```
print(Id_DataFrame.nomcolonne[position_initiale:position_finale])
```

**Question :**

Afficher les valeurs de la colonne "note" de la ligne 1 jusqu'à la ligne 3 du DataFrame "notecsv"

```
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("les informations de la colonne note entre 1 et 4 du DataFrame notecsv")
print(notecsv.note[1:4])
```

```
les informations de la colonne note entre 1 et 4 du DataFrame notecsv
1    19.5
2    16.5
3    18.0
Name: note, dtype: float64
```

**V- Tri :**

Trier c'est organiser une liste de valeurs suivant un ou plusieurs critères et selon un ordre croissant ou décroissant

**1- Tri d'un DataFrame selon les données :**

Pour trier un DataFrame selon les données d'une ou plusieurs colonnes et dans l'ordre croissant ou décroissant on écrit la commande suivante :



```
#importation de la bibliothèque pandas
from pandas import *
#Dans une nouvelle DataFrame
DataFrame_Trié=id_DataFrame.sort_values
(
    by=["Nom_col_1", "Nom_col_2"],
    ascending=[True, False]
)

# Ordre croissant : Ascending=True
#Ordre décroissant :ascending=False
```



## Question :

Trier les données du DataFrame "notecsv" dans l'ordre croissant des notes dans un nouveau DataFrame nommé "notetri" et l'afficher

```
notecsv = read_csv("note.csv", sep=";", decimal=".")
notetri=notecsv.sort_values(
    by=["note"],
    ascending=True)
print("affichage du DataFrame trier par les notes des élèves")
print(notetri)
```

```
affichage du DataFrame trier par les notes des élèves
  Nom prenom id_eleve  libelle  note
10  Sassi    Ali  ele001  economie  5.0
8   Sassi    Ali  ele001  gestion  13.5
9   Sassi    Ali  ele001  Math    15.0
5   hammi   Hela  ele003  Math    15.5
2   Ben salah Sami  ele002  economie  16.5
0   Ben salah Sami  ele002  gestion  17.0
4   hammi   Hela  ele003  gestion  17.5
7   hammi   Hela  ele003  informatique  17.5
3   Ben salah Sami  ele002  informatique  18.0
6   hammi   Hela  ele003  economie  18.5
1   Ben salah Sami  ele002  Math    19.5
11  Sassi    Ali  ele001  informatique  19.5
```

## Question :

Trier les données du DataFrame "notecsv" dans l'ordre croissant des notes et dans l'ordre décroissant des noms dans un nouveau DataFrame nommé "notetri" et l'afficher.

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv", sep=";", decimal=".")
notetri=notecsv.sort_values(
    by=["note", "Nom"],
    ascending=[True, False])
print("affichage du DataFrame trier par les notes des élèves et les nom")
print(notetri)
```

```
affichage du DataFrame trier par les notes des élèves et les nom
  Nom prenom id_eleve  libelle  note
10  Sassi    Ali  ele001  economie  5.0
8   Sassi    Ali  ele001  gestion  13.5
9   Sassi    Ali  ele001  Math    15.0
5   hammi   Hela  ele003  Math    15.5
2   Ben salah Sami  ele002  economie  16.5
0   Ben salah Sami  ele002  gestion  17.0
4   hammi   Hela  ele003  gestion  17.5
7   hammi   Hela  ele003  informatique  17.5
3   Ben salah Sami  ele002  informatique  18.0
6   hammi   Hela  ele003  economie  18.5
11  Sassi    Ali  ele001  informatique  19.5
1   Ben salah Sami  ele002  Math    19.5
```



## VII. Filtrage des données d'un DataFrame

C'est affichage des données suivant un ou plusieurs critères (afficher les élèves qui ont une note supérieure à 15...)

Les opérateurs de comparaisons (**Op\_Comparaison**) et logique (**op\_logique**) sont :

Les opérateurs de comparaison						Les opérateurs logique				
==	<	<=	>	>=	!=	isin()	ET	&	OU	

### 1- Filtrage selon une seule condition :

Filtrer c'est afficher seulement des données selon un ou plusieurs critères.

Pour afficher les données d'un DataFrame suivant un seul critère on utilise la commande suivante :



```
#importation de la bibliothèque pandas
from pandas import *

print(id_DataFrame[id_DataFrame["Nom_colonne"] Op_Comparaison Valeur])
```

### Question :

Afficher la liste des élèves qui ont une note supérieure à 16 du DataFrame "notecsv"

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("affichage la liste des élèves qui ont une note>=16")
print(notecsv[notecsv["note"]>=16])
```

```
affichage la liste des élèves qui ont une note>=16
      Nom prenom id_eleve      libelle  note
0  Ben salah   Sami   ele002      gestion  17.0
1  Ben salah   Sami   ele002         Math  19.5
2  Ben salah   Sami   ele002      economie  16.5
3  Ben salah   Sami   ele002  informatique  18.0
4      hammi   Hela   ele003      gestion  17.5
6      hammi   Hela   ele003      economie  18.5
7      hammi   Hela   ele003  informatique  17.5
11     Sassi   Ali   ele001  informatique  19.5
```

## 2- Filtrage selon deux conditions :

Pour afficher les données d'un DataFrame suivant deux critères on utilise la commande suivante :



```
#importation de la bibliothèque pandas
from pandas import *

print(id_DataFrame[(id_DataFrame["Nom_col_1"] Op_ComparaisonValeur1) Op_logique
(id_DataFrame["Nom_col_2"] Op_Comparaison Valeur2)])
```

### Question :

Afficher la liste des élèves qui ont des notes supérieures à 15 dans la matière "gestion"

```
#importation des bibliothèque pandas
from pandas import *
notecsv = read_csv("note.csv",sep=";",decimal=".")
print("affichage la liste des élèves qui ont une note>=15 dans gestion")
print(notecsv[(notecsv["note"] >=15)& (notecsv["libelle"] =="gestion")])
```

```
affichage la liste des élèves qui ont une note>=15 dans gestion
      Nom prenom id_eleve  libelle  note
0  Ben salah   Sami   ele002  gestion  17.0
4     hammi   Hela   ele003  gestion  17.5
```

## VIII. Les fonctions statistiques

Les fonctions statistiques	
Nom	Rôle
mean()	Calcul la moyenne d'une série de valeurs
max()	Détermine le maximum d'une série
min()	Détermine le minimum d'une série
sum()	Calcul la somme d'une série de valeurs
count	Calcul le nombre d'éléments dans une série non vide

Pour utiliser ces fonctions ci-dessus on utilise la commande suivant qui a la forme générale suivante :



```
#importation de la bibliothèque pandas
from pandas import *
# 1ère méthode
print(df.nom_colonne.nom_fonction())
# 2ème méthode
print(df["nom_colonne"].nom_fonction())
```

### Question :

Calculer et afficher la note maximale et la note minimale du DataFrame "notecsv"

```
#importation des bibliothèques pandas
from pandas import *
notecsv = read_csv("note.csv", sep=";", decimal=".")
print("affichage de la note max et la note min dans notecsv")
print("notemax=", notecsv.note.max(), "note min=", notecsv["note"].min())
```

```
affichage de la note max et la note min dans notecsv
notemax= 19.5 note min= 5.0
```

## IX. Graphiques :

Pour créer un graphique importer la bibliothèque graphique :

---

```
import matplotlib.pyplot as plt
```

---

### 1- Graphique à barres

Pour représenter graphiquement en barre la colonne "Nom\_colonne\_ordonnée" en fonction de "Nom\_colonne\_Abscisse"



```
from pandas import *
import matplotlib.pyplot as plt

id_DF.plot.bar(
    x="Nom_colonne_Abscisse", y="Nom_colonne_Ordonnée",
    title="Titre_Graphique",
    color="couleur_Graphique"
)
```

- Pour afficher un graphique utiliser la commande **show()** :



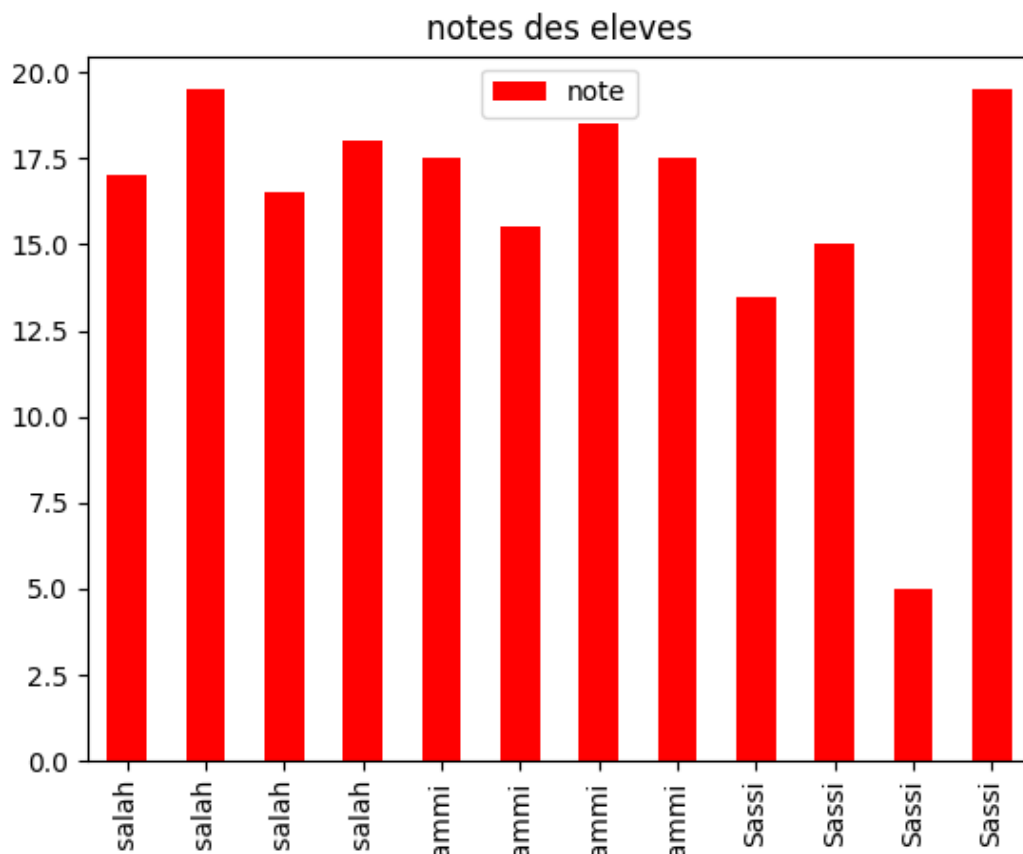
```
from pandas import *  
import matplotlib.pyplot as plt
```

**Question :**

Créer un graphique en barre des notes des élèves du DataFrame "notecsv"

- Titre : notes des élèves
- Couleur : rouge

```
#importation des bibliothèque pandas  
from pandas import *  
import matplotlib.pyplot as plt  
notecsv = read_csv("note.csv",sep=";",decimal=".")  
notecsv.plot.bar(x="Nom",  
                 y="note",  
                 title="notes des eleves",  
                 color="red"  
                )  
plt.show()
```





## 2- Graphique en courbe

Pour représenter graphiquement en courbe la colonne "Nom\_colonne\_ordonnée" en fonction de "Nom\_colonne\_Abscisse"



```
from pandas import *
import matplotlib.pyplot as plt
```

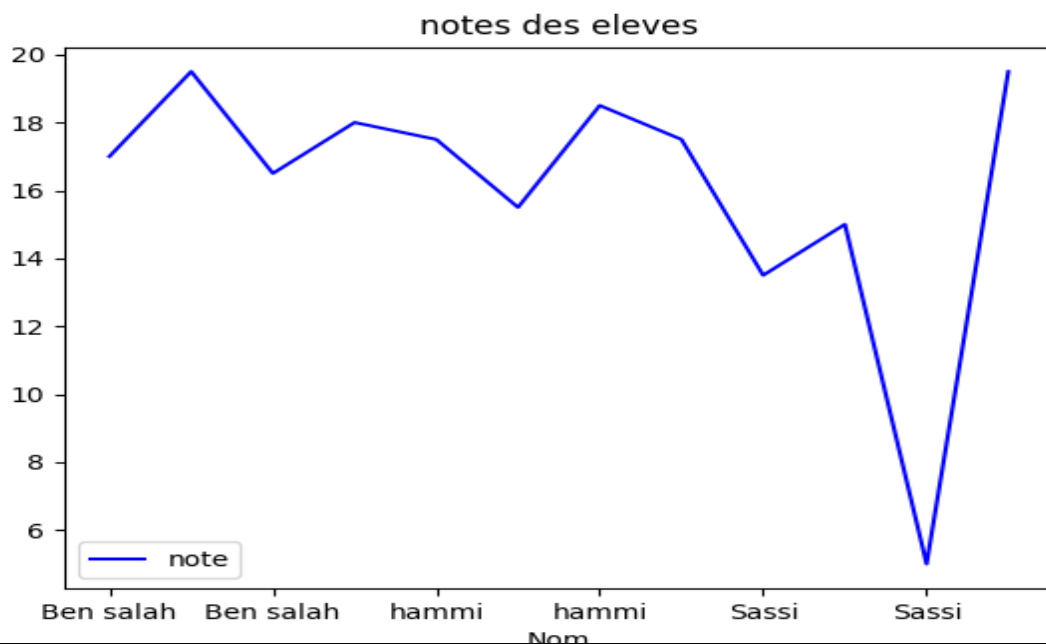
```
id_DF.plot.line(
x="Nom_colonne_Abscisse",y="Nom_colonne_Ordonnée",
title="Titre_Graphique",color="couleur_Graphique")
plt.show()
```

Question :

Créer un graphique en barre des notes des élèves du DataFrame "notecsv"

- Titre : notes des élèves
- Couleur : bleu

```
#importation des bibliothèque pandas
from pandas import *
import matplotlib.pyplot as plt
notecsv = read_csv("note.csv",sep=";",decimal=".")
notecsv.plot.line(x="Nom",
                  y="note",
                  title="notes des eleves",
                  color="blue"
                  )
```



**X. Création d'une colonne calculable :**

Pour créer une colonne à partir d'autres colonnes on utilise la commande suivante :



```
from pandas import *
import matplotlib.pyplot as plt

nouvelle_colonne=id_DA["nom_colonne1"] op_arithmétique id_DF["nom_colonne2"]
```

**Question :**

Créer une colonne nommée `totalobt` = `note`\* `coefficient` du DataFrame "notecsv" pour chaque élève

```
#importation des bibliothèques pandas
from pandas import *
import matplotlib.pyplot as plt
notecsv = read_csv("note.csv", sep=";", decimal=".")
#création d'une colonne coefficient
notecsv["coefficient"]=[3,2,3,1,3,2,3,1,3,2,3,1]
#création d'une colonne à partir d'un calcul
notecsv["noteobt"]=notecsv["coefficient"]*notecsv["note"]
print("affichage du DataFrame")
print(notecsv)
```

```
affichage du DataFrame
```

	Nom	prenom	id_eleve	libelle	note	coefficient	noteobt
0	Ben	salah	Sami	ele002	gestion	17.0	3 51.0
1	Ben	salah	Sami	ele002	Math	19.5	2 39.0
2	Ben	salah	Sami	ele002	economie	16.5	3 49.5
3	Ben	salah	Sami	ele002	informatique	18.0	1 18.0
4	hammi	Hela	ele003	gestion	17.5	3 52.5	
5	hammi	Hela	ele003	Math	15.5	2 31.0	
6	hammi	Hela	ele003	economie	18.5	3 55.5	
7	hammi	Hela	ele003	informatique	17.5	1 17.5	
8	Sassi	Ali	ele001	gestion	13.5	3 40.5	
9	Sassi	Ali	ele001	Math	15.0	2 30.0	
10	Sassi	Ali	ele001	economie	5.0	3 15.0	
11	Sassi	Ali	ele001	informatique	19.5	1 19.5	

**XI- Nettoyage d'un DataFrame :**

Voici un contenu d'un DataFrame

	Pays	superficie
0	Russie	17098246
1	Canada	9984670
2	USA	9600000
3	Chine	9833517
4	Brésil	8515767
5	Australie	7692024
6	Russie	17098246
7	Canada	9984670



Pour supprimer ou nettoyer un DataFrame des doublons :

```
from pandas import *
Résultat = Id_DataFrame . drop_duplicates ( )
```

```
from pandas import *
stat_pays=DataFrame({
    "Pays":["Russie","Canada","USA","Chine","Brésil","Australie","Russie","Canada"],
    "superficie":[17098246,9984670,9600000,9833517,8515767,7692024,17098246,9984670 ]
})

print(stat_pays)
stat_pays=stat_pays.drop_duplicates()
print("liste des pays sans duplications")
print(stat_pays)
```

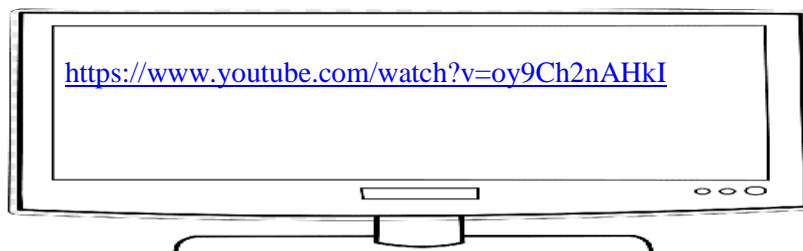
	Pays	superficie
0	Russie	17098246
1	Canada	9984670
2	USA	9600000
3	Chine	9833517
4	Brésil	8515767
5	Australie	7692024
6	Russie	17098246
7	Canada	9984670

liste des pays sans duplications

	Pays	superficie
0	Russie	17098246
1	Canada	9984670
2	USA	9600000
3	Chine	9833517
4	Brésil	8515767
5	Australie	7692024



<https://www.youtube.com/watch?v=oy9Ch2nAHkI>



## TP de révision

Code	Nom	Prix
Act1	Aérobic	10
Act2	Karaté	15.5
Act3	Musculation	15
Act4	Danse orientale	8
Act1	Aérobic	10
Act4	Danse orientale	8

Questions :

- 1- Lancer un éditeur de "Thonny" et créer un fichier nommé sport.py
- 2- Créer un nouveau DataFrame nommé "sport" d'après le tableau ci-dessus
- 3- Supprimer les doublons du DataFrame "sport"
- 4- Afficher le nombre d'éléments dans le DataFrame "sport"
- 5- Afficher les informations relatives à ce DataFrame
- 6- Ajouter les deux lignes suivantes :

Code	Nom	Prix
Act5	Danse classique	18
Act6	box	7

- 7- Renommer les colonnes :

Ancien nom	Nouveau nom
Code	Activité
Nom	Nom_act

- 8- Afficher les noms de colonne du DataFrame "sport"
- 9- Le prix de karaté à augmenter de 10 d. afficher cette ligne
- 10- Ajouter une nouvelle colonne nommé "nbr\_seance" qui égale à [6,5,4,6,4,7] pour chaque mois.
- 11- Ajouter une nouvelle colonne prix\_payer = prix\*nbr\_seance
- 12- Trier le DataFrame "sport" dans l'ordre décroissant des prix à payer dans un nouveau DataFrame nommé sport\_trier
- 13- Afficher le DataFrame "sport\_trier"
- 14- Afficher le nombre d'activité
- 15- Afficher la somme des totaux à payer
- 16- Afficher les activités dont le nombre de séance inférieur à 5
- 17- Afficher la liste des activités dont le prix est entre [10,15]
- 18- Créer un graphique en courbe représentant les prix des activités selon des noms des activités :
  - Le nom : "variation des prix"
  - Couleur : rouge
  - Réunies



## Correction du TP :

```

1 from pandas import *
2 import matplotlib.pyplot as plt
3 """Création d'un DataFrame sport
4 """
5 sport=DataFrame({
6
7     "Code":["Act1","Act2","Act3","Act4","Act1","Act4"],
8     "Nom":["Aerobic","Karaté","Musclation","Danse orientale","Aerobic","Danse orientale"],
9     "Prix":[10,15.5,15,8,10,8]
10
11 })
12 """Suppression des doublons du DataFrame sport
13 """
14 sport=sport.drop_duplicates()
15
16 print("le nombre des éléments dans le DataFrame =",sport.size)
17 print("les information sur le dataframe sport sont :")
18 print(sport.info())
19
20 ajout deux nouveaux ligne
21 """
22 sport.loc[4]=["Act5","Danse classique",18]
23 sport.loc[5]=["Act6","box",7]
24 print("renommage des colonnes")
25 sport=sport.rename(columns={"Code":"activite",
26                             "Nom":"Nom_act"})
27 print(sport.columns)
28 """Modification du prix Karaté"""
29 sport.loc[3,"Prix"]=10
30 print("affichage de ligne qui a été modifier")
31 print(sport.iloc[3])
32 print("ajout d'une nouvelle colonne séance")
33 sport["seance"]=[6,5,4,6,4,7]
34 print("création d'une nouvelle colonne calculable prix à payer")
35 sport["prix_payer"]=sport["seance"]*sport["Prix"]
36 print(sport)
37 """tri du dataframe sport """
38 sport_trier=sport.sort_values(by=["prix_payer"],ascending=False)
39 print(sport_trier)
40 print("nombre d'activité=",sport["activite"].count())
41 print("le montant Total=",sport[["prix_payer"]].sum())
42 """
43 affichage des ligne dont le nombre de séance >5
44 """
45 print(sport[sport["seance"]>5])
46 """
47 Affichage des lignes don le prix de séance entre [10,15]
48 """
49 print(sport[(sport["Prix"]>=10) & (sport["Prix"]<=15)])
50 """
51 Création d'un graphique en courbe
52 """
53 sport.plot.line(x="Nom_act",
54                 y="Prix",
55                 title="activités",
56                 color="red")
57 plt.show()
58

```



## Devoir 1

### EXERCICE N°1:

Le responsable informatique de l'hôtel veut gérer les réservations effectués pendant la saison de l'été, pour cela, il a exporté la table **Réservation** dans un fichier

- "Réservation.csv" situé sous D:/
- Un point-virgule en séparateur de champs
- Point en séparateur décimal.

Pour pouvoir analyser les données et prendre des décisions, il a utilisé la bibliothèque "Pandas" du langage de programmation « Python ».

1) Ecrire le script nécessaire qui permet d'importer le fichier « **Réservation.csv** » dans un DataFrame nommé « **res** » puis l'afficher.

	Ref_res	Ref_clt	Ref_cui	Salle	Date_res	Heure_res	Montant
0	R1	A1	B1	Séminaire	04/03/2021	12:00	4000
1	R2	A4	B1	Mariage	11/05/2021	20:00	1680
2	R3	A1	B1	Séminaire	12/05/2021	12:00	3600
3	R4	A2	B1	Salon	28/07/2021	19:00	1150
4	R5	A2	B2	Séminaire	30/01/2021	12:00	2240
5	R6	A1	B2	Salon	20/04/2021	20:00	1680

2) Relier chaque méthode par le résultat qui lui correspond :

Méthode	Résultat																							
<code>print(res.shape)</code>	<table border="1"> <thead> <tr> <th>Ref_res</th> <th>Ref_clt</th> <th>Ref_cui</th> <th>Salle</th> <th>Date_res</th> <th>Heure_res</th> <th>Montant</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>R5</td> <td>A2</td> <td>B2</td> <td>Séminaire</td> <td>30/01/2021</td> <td>12:00</td> <td>2240</td> </tr> <tr> <td>5</td> <td>R6</td> <td>A1</td> <td>B2</td> <td>Salon</td> <td>20/04/2021</td> <td>20:00</td> <td>1680</td> </tr> </tbody> </table>	Ref_res	Ref_clt	Ref_cui	Salle	Date_res	Heure_res	Montant	4	R5	A2	B2	Séminaire	30/01/2021	12:00	2240	5	R6	A1	B2	Salon	20/04/2021	20:00	1680
Ref_res	Ref_clt	Ref_cui	Salle	Date_res	Heure_res	Montant																		
4	R5	A2	B2	Séminaire	30/01/2021	12:00	2240																	
5	R6	A1	B2	Salon	20/04/2021	20:00	1680																	
<code>print(res.tail(2))</code>	(6, 7)																							
<code>print(res.size)</code>	Index(['Ref_res', 'Ref_clt', 'Ref_cui', 'Salle', 'Date_res', 'Heure_res', 'Montant'], dtype='object')																							
<code>print(res.columns)</code>	42																							

3) Soit le script ci-dessous :

```
print(res[res['Montant'] > 3000])
```

Donner le rôle de ce script : .....

4) Compléter le script ci-dessous pour chercher la somme des montants des réservations :

```
print(res[.....]......())
```

5) Pour afficher la première ligne du DataFrame, encadrer la bonne réponse:

```
print(res.iloc [0])
```

```
print (res.iloc [1])
```

6) On désire modifier le nom du colonne " Ref\_clt" par "Code\_clt" .

En tapant la commande suivante :

```
res.rename(columns={'Ref_clt': 'Code_clt'})
print(res)
```



Le DataFrame "res" s'affiche comme suit :

	Ref_res	Ref_clt	Ref_cui	Salle	Date_res	Heure_res	Montant
0	R1	A1	B1	Séminaire	04/03/2021	12:00	4000
1	R2	A4	B1	Mariage	11/05/2021	20:00	1680
2	R3	A1	B1	Séminaire	12/05/2021	12:00	3600
3	R4	A2	B1	Salon	28/07/2021	19:00	1150
4	R5	A2	B2	Séminaire	30/01/2021	12:00	2240
5	R6	A1	B2	Salon	20/04/2021	20:00	1680

Apporter les modifications nécessaires à la commande proposée pour effectuer le changement du nom de colonne.

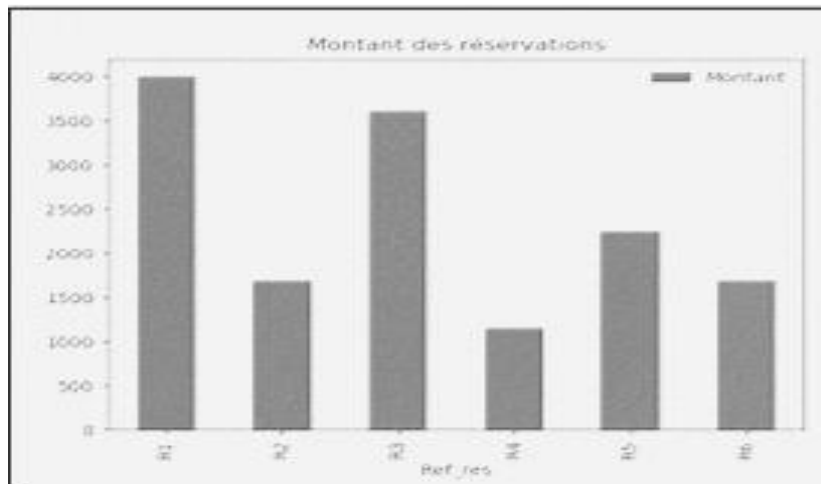
2- Le script suivant: colorie la bonne réponse

```
resultat=res.sort_values(by=['Montant'], ascending=True)
print(resultat)
```

- Permet de trier le DataFrame resultat par ordre croissant selon la colonne Montant
- Permet de trier le DataFrame res par ordre croissant selon la colonne Montant
- Permet de trier le DataFrame resultat par ordre décroissant selon la colonne Montant

3- Compléter le script pour créer le graphique suivant :

```
import matplotlib.pyplot as plt
res.plot.....(x='.....',y='.....',
color='red', title='Montant des réservations')
plt.show()
```





## Devoir 2

Afin d'avoir une idée claire sur les locations des robes réalisées pendant l'année 2021, Le responsable du service informatique de l'espace **Bella** a exporté le résultat de la requête "R5" dans le fichier "R5.csv" situé à la racine C. Pour analyser les données exportées, on utilise la bibliothèque Pandas du langage de programmation Python.

1. Parmi les propositions ci-dessous, encrer le numéro du script à écrire afin d'importer le contenu du fichier "R5.csv" dans un DataFrame nommé "resultat". [Voir page ci-dessus](#)

Numéro du script	Script
1	<code>resultat = pandas.read_csv ("c:\R5.xlsx", sep = ";")</code>
2	<code>Resultat=pandas.DataFrame({"NomPrenom","Description","Couleur", "Prix","DateLocation", "DateRetour"})</code>
3	<code>resultat = pandas.read_csv ("c:\R5.csv", sep = ";")</code>
4	<code>resultat = pandas.read_excel ("c:\R5.xlsx", "client")</code>

2. Afin d'afficher des informations sur le DataFrame "resultat", compléter le tableau ci-dessous en associant à chaque numéro de méthode la lettre correspondante au type d'affichage.

Méthode	Numéro	Lettre	Type d'affichage
1. shape	1	.....	a. Affichage du nombre d'éléments d'un DataFrame
2. size	2	.....	b. Affichage des informations concernant un DataFrame
3. describe ()	3	.....	c. Affichage des dimensions d'un DataFrame
4. info ()	4	.....	d. Affichage du résumé rapide d'un DataFrame

3. Compléter la commande ci-dessous permettant d'afficher le nombre de locations réalisées pendant l'année 2021.

```
print ("Le nombre de locations en 2021 est :", resultat
["....."]......)
```

4. Donner le rôle du script suivant :

```
Resultat_nouveau=resultat.sort_values(by = ['Prix'], ascending=False)
```

.....  
 .....  
 .....  
 .....





.....  
 .....

5. Lors de l'exécution d'un script de génération d'un graphique, l'erreur suivante s'affiche :

```
Traceback (most recent call last):
  File "C:\Gestion Location.py", line 2, in <module>

    plt.show()
NameError: name 'plt' is not defined
```

A- Compléter le script ci-dessous par l'une des commandes suivantes pour éviter l'affichage de cette erreur.

Liste des commandes :

- import pandas
- import pandas as plt
- import matplotlib.pyplot as plt
- import matplotlib.pyplot as pd

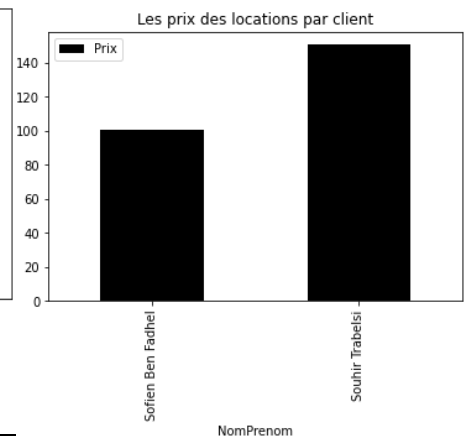
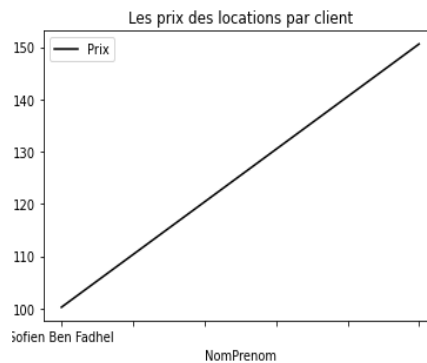
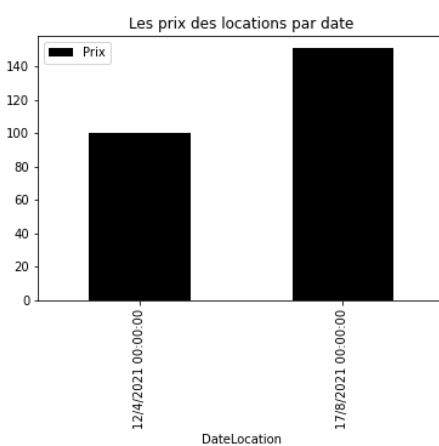
Script à corriger :

```
.....

resultat.plot.bar(x="NomPrenom",y="Prix",title="Les prix des locations par client",color="black")

plt.show()
```

a. Soit les trois graphiques ci-dessous (1, 2 et 3), donner le numéro correspondant au graphique affiché suite à l'exécution du script précédent : .....



- 1 -	- 2 -	- 3 -

## Correction Devoir 1 :(pandas)

### EXERCICE N°1:

Le responsable informatique de l'hôtel veut gérer les réservations effectués pendant la saison de l'été, pour cela, il a exporté la table **Réservation** dans un fichier

- "Réservation.csv" situé sous D:/
- Un point-virgule en séparateur de champs
- Point en séparateur décimal.

Pour pouvoir analyser les données et prendre des décisions, il a utilisé la bibliothèque "Pandas" du langage de programmation « Python ».

- 1- Ecrire le script nécessaire qui permet d'importer le fichier « **Réservation.csv** » dans un DataFrame nommé « **res** » puis l'afficher.

```
from pandas import *
res= read_csv("reservation.csv",sep=";",decimal=".")
print(res)
```

	Ref_res	Ref_clt	Ref_cui	Salle	Date_res	Heure_res	Montant
0	R1	A1	B1	Séminaire	04/03/2021	12:00	4000
1	R2	A4	B1	Mariage	11/05/2021	20:00	1680
2	R3	A1	B1	Séminaire	12/05/2021	12:00	3600
3	R4	A2	B1	Salon	28/07/2021	19:00	1150
4	R5	A2	B2	Séminaire	30/01/2021	12:00	2240
5	R6	A1	B2	Salon	20/04/2021	20:00	1680

- 2- Relier chaque méthode par le résultat qui lui correspond :

Méthode	Résultat																							
<code>print(res.shape)</code>	<table border="1"> <thead> <tr> <th>Ref_res</th> <th>Ref_clt</th> <th>Ref_cui</th> <th>Salle</th> <th>Date_res</th> <th>Heure_res</th> <th>Montant</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>R5</td> <td>A2</td> <td>B2</td> <td>Séminaire</td> <td>30/01/2021</td> <td>12:00</td> <td>2240</td> </tr> <tr> <td>5</td> <td>R6</td> <td>A1</td> <td>B2</td> <td>Salon</td> <td>20/04/2021</td> <td>20:00</td> <td>1680</td> </tr> </tbody> </table>	Ref_res	Ref_clt	Ref_cui	Salle	Date_res	Heure_res	Montant	4	R5	A2	B2	Séminaire	30/01/2021	12:00	2240	5	R6	A1	B2	Salon	20/04/2021	20:00	1680
Ref_res	Ref_clt	Ref_cui	Salle	Date_res	Heure_res	Montant																		
4	R5	A2	B2	Séminaire	30/01/2021	12:00	2240																	
5	R6	A1	B2	Salon	20/04/2021	20:00	1680																	
<code>print(res.tail(2))</code>	(6, 7)																							
<code>print(res.size)</code>	Index(['Ref_res', 'Ref_clt', 'Ref_cui', 'Salle', 'Date_res', 'Heure_res', 'Montant'], dtype='object')																							
<code>print(res.columns)</code>	42																							

- 3- Soit le script ci-dessous :

```
print(res[res['Montant'] > 3000])
```

Donner le rôle de ce script : **afficher la liste des montant > 3000 du DataFrame res**

- 4- Compléter le script ci-dessous pour chercher la somme des montants des réservations :

```
print(res["Montant"].sum())
```

- 5- Pour afficher la première ligne du DataFrame, encadrer la bonne réponse:

<code>print(res.iloc [0])</code>	<code>print (res.iloc [1])</code>
----------------------------------	-----------------------------------

- 6- On désire modifier le nom du colonne " Ref\_clt" par "Code\_clt" .

En tapant la commande suivante :



```
res.rename(columns={'Ref_clt':'Code_clt'})
print(res)
```

Le DataFrame "res" s'affiche comme suit :

	Ref_res	Ref_clt	Ref_cui	Salle	Date_res	Heure_res	Montant
0	R1	A1	B1	Séminaire	04/03/2021	12:00	4000
1	R2	A4	B1	Mariage	11/05/2021	20:00	1680
2	R3	A1	B1	Séminaire	12/05/2021	12:00	3600
3	R4	A2	B1	Salon	28/07/2021	19:00	1150
4	R5	A2	B2	Séminaire	30/01/2021	12:00	2240
5	R6	A1	B2	Salon	20/04/2021	20:00	1680

Apporter les modifications nécessaires à la commande proposée pour effectuer le changement du nom de colonne.

```
from pandas import *
res.rename(columns={'Ref_clt':'code_clt'}, inplace=True)
print(res)
```

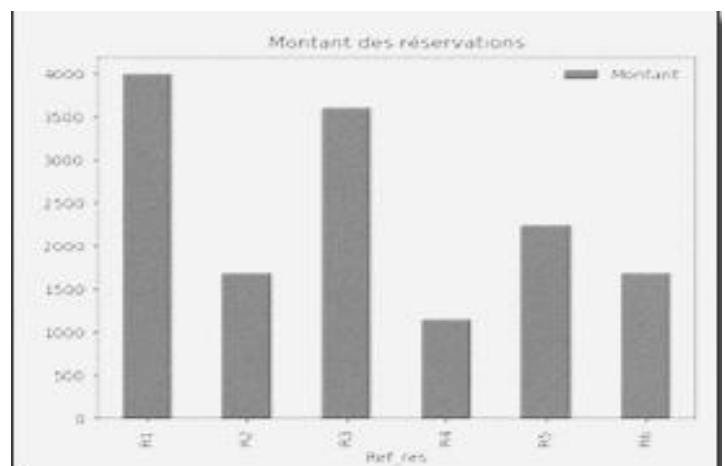
7- Le script suivant: colorie la bonne réponse

```
resultat=res.sort_values(by=['Montant'], ascending=True)
print(resultat)
```

- Permet de trier le DataFrame resultat par ordre croissant selon la colonne Montant
- Permet de trier le DataFrame res par ordre croissant selon la colonne Montant
- Permet de trier le DataFrame resultat par ordre décroissant selon la colonne Montant

8- Compléter le script pour créer le graphique suivant :

```
import matplotlib.pyplot as plt
res.plot.bar(x="Ref_res",
             y="Montant",
             color="green",
             title="Montant des réservations")
plt.show()
```



## Correction devoir 2 : (pandas)

Afin d'avoir une idée claire sur les locations des robes réalisées pendant l'année 2021, Le responsable du service informatique de l'espace **Bella** a exporté le résultat de la requête "R5" dans le fichier "R5.csv" situé à la racine C. Pour analyser les données exportées, on utilise la bibliothèque Pandas du langage de programmation Python. [Voir page ci-dessus](#)

Parmi les propositions ci-dessous, encrer le numéro du script à écrire afin d'importer le contenu du fichier "R5.csv" dans un DataFrame nommé "resultat".

- Séparateur entre les données ";"
- Décimal="."

Numéro du script	Script
1	<code>resultat = pandas.read_csv ("c:\R5.xlsx", sep = ";")</code>
2	<code>Resultat=pandas.DataFrame({"NomPrenom","Description","Couleur", "Prix","DateLocation", "DateRetour"})</code>
3	<code>resultat = pandas.read_csv ("c:\R5.csv", sep = ";",decimal=".")</code>
4	<code>resultat = pandas.read_table ("c:\R5.txt", sep=";",decimal=".")</code>

1- Afin d'afficher des informations sur le DataFrame "resultat", compléter le tableau ci-dessous en associant à chaque numéro de méthode la lettre correspondante au type d'affichage.

Méthode	Numéro	Lettre	Type d'affichage
1. shape	1	c	a- Affichage du nombre d'éléments d'un DataFrame
2. size	2	a	b- Affichage des informations concernant un DataFrame
3. describe ()	3	d	c- Affichage des dimensions d'un DataFrame
4. info ()	4	b	d- Affichage du résumé rapide d'un DataFrame

2- Compléter la commande ci-dessous permettant d'afficher le nombre de locations réalisées pendant l'année 2021.

```
print ("Le nombre de locations en 2021 est :",  
resultat ["prix"].count())
```

3- Donner le rôle du script suivant :

```
Resultat_nouveau=resultat.sort_values(by = ["Prix"],
```

`ascending=False)`

Permet de trier les données du DataFrame "resultat" dans un nouveau Dataframe nommé "Resultat\_nouveau" suivant les prix dans l'ordre décroissant

4- Lors de l'exécution d'un script de génération d'un graphique, l'erreur suivante s'affiche :

```
Traceback (most recent call last):
  File "C:\Gestion Location.py", line 2, in <module>
    plt.show()
NameError: name 'plt' is not defined
```

a- Compléter le script ci-dessous par l'une des commandes suivantes pour éviter l'affichage de cette erreur.

*Liste des commandes :*

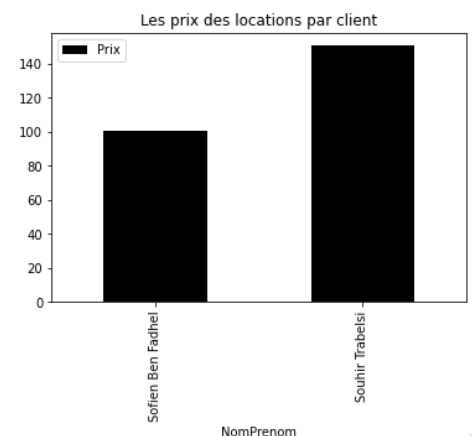
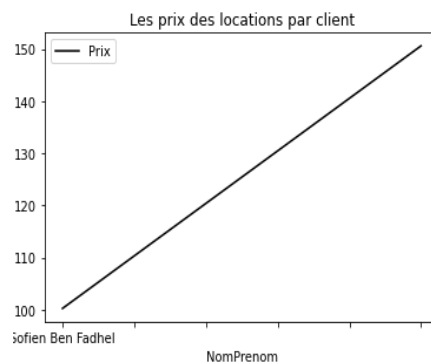
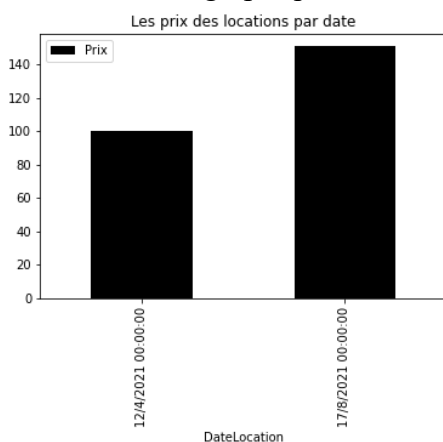
```
import pandas
import pandas as plt
import matplotlib.pyplot as plt
import matplotlib.pyplot as pd
```

*Script à corriger :*

```
import matplotlib.pyplot as plt .....
resultat.plot.bar(x="NomPrenom",y="Prix",title="Les prix des locations par client",color="black")
plt.show()
Car il a utilisé l'alias plt
```

b- Soit les trois graphiques ci-dessous (1, 2 et 3), donner le numéro correspondant au

graphique affiché suite à l'exécution du script précédent : **3**



## correction Bac théorique 2022

### Partie I (13 points)

Un enseignant veut présenter aux élèves les différentes notions géographiques en rapport avec les fleuves. Pour cela et à l'aide d'un SGBD, il a créé la base de données simplifiée intitulée "Gestion\_Fleuves" décrite par le schéma textuel suivant :

**Continent** (NumContinent, NomContinent)

**Pays** (CodePays, NomPays, Capitale, Superficie, NbHabitants, NumContinent#)

**Fleuve** (CodeFleuve, NomFleuve, Longueur, Débit)

**Traversée** (CodeFleuve#, CodePays#, Distance)

Soit la description des colonnes des tables de cette base de données :

Nom	Description	Type
NumContinent	Numéro d'un continent	Numérique
NomContinent	Nom d'un continent	Texte
CodePays	Code d'un pays	Texte
NomPays	Nom d'un pays	Texte
Capitale	Capitale d'un pays	Texte
Superficie	Superficie d'un pays en Km <sup>2</sup>	Numérique

Nom	Description	Type
NbHabitants	Nombre d'habitants d'un pays	Numérique
CodeFleuve	Code d'un fleuve	Texte
NomFleuve	Nom d'un fleuve	Texte
Longueur	Longueur totale d'un fleuve en Km	Numérique
Débit	Débit d'un fleuve en m <sup>3</sup> /s	Numérique
Distance	Longueur d'un fleuve dans un pays en Km	Numérique

Soient les extraits des tables de la base de données "Gestion\_Fleuves" suivants :

Table "Pays"					
CodePays	NomPays	Capitale	Superficie	NbHabitants	NumContinent
P001	Tunisie	Tunis	163 610	11 936 000	1
P002	Australie	Canberra	7 682 300	25 788 000	5
P003	Brésil	Brasilia	8 511 996	213 993 000	2
P004	Pérou	Ottawa	1 285 000	32 970 000	2
P005	Colombie	Bogota	1 143 000	50 880 000	2
P006	France	Paris	543 965	65 426 000	4
P007	Japon	Tokyo	377 832	126 051 000	3
P008	Algérie	Alger	2 382 000	43 850 000	1

Table "Continent"	
NumContinent	NomContinent
1	Afrique
2	Amérique
3	Asie
4	Europe
5	Océanie

Table "Fleuve"			
CodeFleuve	NomFleuve	Longueur	Débit
F001	Medjerda	460	1 000
F002	Amazone	6 436	209 000
F003	Loire	1 006	850
F004	Seine	775	563
F005	Amour	4 354	11 000
F006	Danube	2 850	6 500

Table "Traversée"		
CodeFleuve	CodePays	Distance
F001	P001	350
F001	P008	110
F002	P003	4 345
F002	P004	1 871
F002	P005	220
F003	P006	1 006

- 1) En se référant à cette base de données et pour chacune des propositions ci-dessous, compléter la colonne "Valide" par la lettre "V" dans le cas où la proposition est correcte ou par la lettre "F" dans le cas contraire.




Proposition	Valide
Le pays ayant le code "P007" est un pays africain.	F
La dernière colonne de la table "Pays" permet de gérer une relation directe entre la table "Pays" et la table "Continent" et de garantir la cohérence des données.	V
Un fleuve peut traverser un ou plusieurs pays.	V
Lors de l'ajout d'un nouveau fleuve nommé "Le Tigre" dans la table "Fleuve", on peut lui attribuer le code "F006".	F

- 2) En se référant à l'annexe ci-dessous, compléter le tableau suivant en associant à chaque message d'erreur (E1, E2 et E3) généré par le SGBD, l'opération de saisie (S1, S2 ou S3) et la cause (C1, C2 ou C3) correspondantes à cette erreur.

Message d'erreur	Opération de saisie	Cause de l'erreur
E1	S3	C1
E2	S1	C3
E3	S2	C2

## Annexe :

Les messages d'erreurs :

- E1.  Vous devez entrer une valeur dans le champ « Traversée.Distance ».
- E2.  Vous ne pouvez pas ajouter ou modifier un enregistrement car l'enregistrement associé est requis dans la table « Continent ».
- E3.  Une ou plusieurs valeurs sont interdites par la règle de validation « >0 » déterminée pour « Fleuve.Longueur ». Saisissez une valeur que l'expression acceptera pour ce champ.

Les opérations de saisie :

- S1. Modifier la valeur de la colonne "NumContinent" de la table "Pays" par "30" (comme illustré ci-dessous).

Pays					
CodePays	NomPays	Capitale	Superficie	NbHabitants	NumContinent
+ P001	Tunisie	Tunis	163610	11936000	30

- S2. Modifier la valeur de la colonne "Longueur" de la table "Fleuve" par "-460" (comme illustré ci-contre).

Fleuve			
CodeFleuve	NomFleuve	Longueur	Débit
+ F001	Medjerda	-460	1000

- S3. Ajouter un nouveau enregistrement à la table "Traversée" sans saisir la valeur de la colonne "Distance" (comme illustré ci-contre).

Traversée		
CodeFleuve	CodePays	Distance
+ F004	P007	

Les causes d'erreurs :

- C1. La propriété "Null Interdit", du champ concerné, a la valeur "Oui".  
 C2. La propriété "Valide si", du champ concerné, exige une valeur strictement positive.  
 C3. La valeur de la clé étrangère ne figure pas dans la colonne correspondante de la table source.

3) Soient les grilles des requêtes nommées "Requête1", "Requête2" et "Requête3" suivantes :

Requête1		Requête2	
Champ :	NomPays      NomContinent	Champ :	NbHabitants      NomContinent
Table :	Pays              Continent	Table :	Pays              Continent
Tri :		Mise à jour :	[NbHabitants]*1.01
Afficher :	<input checked="" type="checkbox"/>	Critères :	"Amérique"
Critères :	"Amérique"	Ou :	
Ou :			

Requête3					
Champ :	NomFleuve	Débit	Distance	NomContinent	NomPays
Table :	Fleuve	Fleuve	Traversée	Continent	Pays
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Critères :					[Donner un pays:]
Ou :					

Compléter le tableau ci-dessous en indiquant pour chaque rôle le nom de la requête et son type (Sélection / Suppression / Mise à jour / Sélection paramétrée / Suppression paramétrée / Mise à jour paramétrée).

Rôle de la requête	Nom de la requête	Type de la requête
Modifier le nombre d'habitants des pays de l'Amérique.	Requête 2	Mise à jour
Afficher les informations relatives aux fleuves qui traversent un pays dont le nom est donné.	Requête 3	Sélection paramétrée
Afficher les noms des pays de l'Amérique.	Requête 1	Sélection

4) Compléter les grilles de création des requêtes suivantes :

**Requête4** : Afficher les informations relatives aux différents pays (NomPays, Capitale, Superficie, NbHabitants, NomContinent).

Champs :	NomPays	Capital	Superficie	NbHabitants	NomContinent
Table :	pays	pays	pays	pays	Continent
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :					
Ou :					

**Requête5** : Afficher les noms et les superficies des pays traversés par le fleuve "Medjerda".

Champs :	NomPays	Superficie	NomFleuve		
Table :	pays	pays	Fleuve		
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Critères :			"Madjerda"		
Ou :					

**Requête6** : Afficher les informations relatives aux fleuves ayant un débit supérieur à un débit donné.

Champs :	CodeFleuve	NomFleuve	débitFleuve		
Table :	Fleuve	Fleuve	Fleuve		
Tri :					
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Critères :			>[donner le débit]		
Ou :					





## Partie II (7 points)

Pour analyser les données relatives aux caractéristiques géographiques et démographiques de quelques pays, l'enseignant a exporté le résultat de la requête "Requête4" dans le fichier "pays.csv" situé à la racine C. Pour ce faire, il a utilisé la bibliothèque "pandas" du langage de programmation Python pour créer un DataFrame nommé "pays".

- 1) Les propriétés et les méthodes **shape**, **size**, **describe()** et **info()** permettent d'afficher des informations relatives à un DataFrame. Pour chacun des deux affichages du DataFrame "pays", écrire ci-dessous la propriété ou la méthode utilisée.

```
RangeIndex: 8 entries, 0 to 7
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  ---            -
0   NomPays         8 non-null      object
1   Capitale        8 non-null      object
2   Superficie     8 non-null      int64
3   NbHabitants    8 non-null      int64
4   NomContinent   8 non-null      object
dtypes: int64(2), object(3)
memory usage: 288.0+ bytes
```

```
      Superficie  NbHabitants
count  8.000000e+00  8.000000e+00
mean   2.761213e+06  7.962724e+07
std    3.371614e+06  7.059519e+07
min    1.636100e+05  1.193600e+07
25%    5.024318e+05  3.142178e+07
50%    1.214000e+06  4.761940e+07
75%    3.707075e+06  1.273111e+08
max    8.511996e+06  2.161329e+08
```

Propriété ou Méthode utilisée : **info()**

Propriété ou Méthode utilisée : **describe()**

- 2) En utilisant la bibliothèque "pandas", écrire ou compléter les scripts ci-dessous permettant de réaliser les tâches demandées.

- a. Modifier le nombre d'habitants du septième pays du DataFrame "pays" par **12731000**.

```
pays.loc [ 6 , "NbHabitant" ] =
12731000
```

- b. Afficher devant chaque nom de pays sa capitale.

```
print(pays["NomPays"],pays["Capital"])
```

- c. Ajouter dans le DataFrame "pays" une nouvelle colonne nommée "Densite", dont les valeurs sont calculées par la formule suivante : **Densite = NbHabitants / Superficie**

```
pays [ "Densite" ] = pays [ "NbHabitant" ] / pays [ "Superficie" ]
```

- 3) Afin de trier le DataFrame "pays", l'enseignant applique le script suivant :

```
pays_tri = pays.sort_values ( by = ["NbHabitants"] , ascending = False )
```

Pour chacune des propositions ci-dessous, relatives à l'explication de ce script, mettre la lettre "V" dans la case correspondante dans le cas où la proposition est correcte ou la lettre "F" dans le cas contraire.

- |   |   |
|---|---|
| V | Le tri est appliqué sur le nombre d'habitants.                                |
| F | Le tri est réalisé en ordre croissant.  |
| F | Le résultat du tri est enregistré dans le DataFrame "pays".                   |
| V | Le nombre de colonnes du DataFrame trié est le même que le DataFrame initial. |

- 4) L'enseignant veut générer un graphique à barres rouges représentant les noms des pays en fonction de leurs superficies.

Pour chacune des propositions ci-dessous, mettre la lettre "V" dans la case correspondante dans le cas où la proposition permet de générer le graphique demandé ou la lettre "F" dans le cas contraire.

- ```
pays.plot.bar (x="NomPays", y="NbHabitants", title="Les superficies", color="black")
pays.plot.line (x="NomPays", y="Superficie", title="Les superficies", color="red")
pays.plot.bar (x="NomPays", y="Superficie", title="Les superficies", color="red")
pays.plot.bar (x="Superficie", y="NomPays", title="Les superficies", color="red")
```

|   |
|---|
| F |
| F |
| V |
| F |

**PROTOTYPE 2022****Épreuve pratique d'informatique**Section : **Économie et Gestion**Durée : **1heure**Coefficient : **0.5**

Afin de gérer le stockage des céréales collectées dans des silos, on se propose d'utiliser la base de données intitulée "**Gestion\_Cereale**" décrite par le schéma textuel simplifié suivant :

**Cereale** (CodeC, NomC, Prix)

**Silo** (CodeS, Nom, Adresse, Capacité)

**Collecte** (CodeC#, CodeS#, DateC, Quantité)

Soit la description des colonnes des tables de la base de données "**Gestion\_Cereale**":

| Nom      | Description                                                          | Type        | Taille      | Format                                       | Contrainte        |
|----------|----------------------------------------------------------------------|-------------|-------------|----------------------------------------------|-------------------|
| CodeC    | Le code d'une céréale                                                | Texte court | 4           |                                              | Non vide          |
| NomC     | Le nom d'une céréale                                                 | Texte court | 30          |                                              | Non vide          |
| Prix     | Le prix d'un Quintal d'une céréale en Dinar                          | Numérique   | Réel        | Nombre général (3 chiffres après la virgule) | Supérieur à zéro  |
| CodeS    | Le code d'un silo                                                    | Texte court | 4           |                                              | Non vide          |
| Nom      | Le nom d'un silo                                                     | Texte court | 30          |                                              | Non vide          |
| Adresse  | L'adresse d'un silo                                                  | Texte court | 50          |                                              |                   |
| Capacité | La capacité maximale de stockage d'un silo en Quintal.               | Numérique   | Entier long |                                              | Supérieure à zéro |
| DateC    | La date d'une collecte                                               | Date/Heure  |             | Date abrégé                                  | Non vide          |
| Quantité | La quantité, en Quintal, livrée à un silo à une date bien déterminée | Numérique   | Entier      |                                              | Supérieure à zéro |

**A) À l'aide du logiciel de gestion de base de données disponible :**

- 1) Créer, dans votre dossier de travail, la base de données "**Gestion\_Cereale**".
- 2) Créer les tables et les relations de cette base de données tout en respectant les types et les propriétés cités dans la description ci-dessus.
- 3) Remplir les tables par les données suivantes :

| Cereale |            |        |
|---------|------------|--------|
| CodeC   | NomC       | Prix   |
| C01     | Blé Dur    | 87.255 |
| C02     | Blé Tendre | 67.732 |
| C03     | Orge       | 56.500 |
| C04     | Triticale  | 56.500 |

| Silo  |          |          |          |
|-------|----------|----------|----------|
| CodeS | Nom      | Adresse  | Capacité |
| S01   | STBlé    | Jendouba | 255000   |
| S02   | CerNord  | Beja     | 270000   |
| S03   | ColKef   | Le Kef   | 189987   |
| S04   | StockBlé | Bizerte  | 265000   |

| Collecte |       |            |          |
|----------|-------|------------|----------|
| CodeC    | CodeS | DateC      | Quantité |
| C01      | S01   | 29/06/2021 | 4200     |

|     |     |            |      |
|-----|-----|------------|------|
| C02 | S02 | 29/06/2021 | 4500 |
| C03 | S03 | 30/06/2021 | 3500 |
| C04 | S04 | 30/06/2021 | 3900 |
| C01 | S02 | 16/07/2021 | 4900 |
| C02 | S01 | 28/07/2021 | 6230 |

4) Créer les requêtes suivantes :

**R1** : Afficher les noms et les quantités des céréales collectées le **30/06/2021**.

**R2** : Afficher la liste des silos (**Nom, Adresse, Capacité**) qui ont stocké une céréale de code donné.

**R3** : Afficher la somme des quantités collectées dans le silo de code "S01".

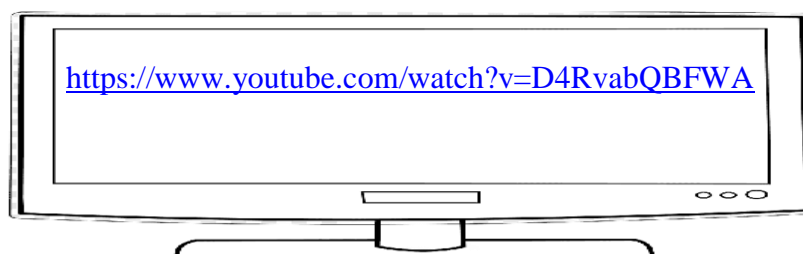
**R4** : Afficher la liste des céréales collectées  
(**CodeC, Prix, CodeS, Nom, Quantité, DateC**).

5) Exporter, dans votre dossier de travail et sous le nom de "**Collecte.csv**", le contenu de la table "**Collecte**" au format **csv** (virgule en séparateur de champs, point en séparateur décimal, conserver le numéro des lignes).

**B) À l'aide de l'environnement de développement pour Python disponible :**

- 1) Créer un script et l'enregistrer sous le nom "**Collecte.py**" dans votre dossier de travail.
- 2) En se référant à l'annexe de la page 3, choisir les méthodes adéquates, de la bibliothèque Pandas, pour écrire le script "**Collecte.py**" qui permet de/d' :
  - a) Importer le contenu de la table intitulée "**Collecte.csv**" et le stocker dans un DataFrame qu'on nommera "**Collecte**",
  - b) Afficher les informations relatives au DataFrame "**Collecte**" ainsi que les 6 premières lignes,
  - c) Remplacer la quantité de la 5<sup>ème</sup> ligne, dans le DataFrame "**Collecte**", par 5560 au lieu de 3500,
  - d) Afficher les informations relatives aux céréales dont la quantité collectée est supérieure ou égale à 4500.
  - e) Trier le DataFrame "**Collecte**" dans un nouveau DataFrame nommé "**Collecte\_Tri**" selon la quantité collectée dans l'ordre croissant. Afficher le nouveau DataFrame,
  - f) Calculer et d'afficher la moyenne des quantités collectées,
  - g) Générer puis d'afficher, à partir du DataFrame "**Collecte**", un graphique à barres rouges représentant, pour chaque date de collecte, la quantité collectée.

## Correction de prototype 2022



## ANNEXE 2023

### Les méthodes à utiliser avec la bibliothèque Pandas

| Catégorie                                                              | Syntaxe                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Création d'un DataFrame                                                | <pre>Id_DataFrame = pandas. DataFrame({"Id_Colonne1":["Val1",..., "ValN"],..., "Id_ColonneM":["Val1",..., "ValN"]})</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Importation d'un DataFrame                                             | <pre>Id_DataFrame = pandas.read_excel ("Chemin/Nom_Fichier.xlsx", "Nom_Feuille") Id_DataFrame = pandas.read_csv("Chemin/Nom_Fichier.extension", sep = "séparateur")</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Manipulation d'un DataFrame                                            | <pre>Id_DataFrame.shape ou bien print (Id_DataFrame.shape) Id_DataFrame.size ou bien print (Id_DataFrame.size) Id_DataFrame.info() ou bien print (Id_DataFrame.info()) Id_DataFrame.describe()ou bien print (Id_DataFrame.describe()) Id_DataFrame.columns ou bien print (Id_DataFrame.columns) Resultat = Id_DataFrame.rename (columns = {"NomColonne1":"NouveauNomColonne1", "NomColonne2":"NouveauNomColonne2",...}) Resultat = Id_DataFrame.drop ([N° Ligne1, ..., N° LigneN]) Resultat = Id_DataFrame.drop ( Id_DataFrame .index [N° LigneInitial : N° LigneFinal ]) Resultat = Id_DataFrame.drop ( columns = ["NomColonne1",..., "NomColonneN"])</pre>                                                                                                                                                           |
| Affichage des données d'un DataFrame                                   | <pre>Id_DataFrame ou bien print(Id_DataFrame) Id_DataFrame ["NomColonne"] ou bien print ( Id_DataFrame [ "NomColonne" ] ) Id_DataFrame [ [ "NomColonne1", "NomColonne2", ...] ] ou bien print Id_DataFrame [ [ "NomColonne1", "NomColonne2", ...] ] Id_DataFrame ["NomColonne"] [N° LigneInitial : N° LigneFinal ] ou bien print ( Id_DataFrame ["NomColonne"] [N° LigneInitial : N° LigneFinal ] ) Id_DataFrame.iloc [N° Ligne] ou bien print(Id_DataFrame.iloc[N° Ligne]) Id_DataFrame.head (n) print ( Id_DataFrame.head(n)) Id_DataFrame.tail (n) print ( Id_DataFrame.tail(n)) Id_DataFrame.iloc [ N° LigneInitial : N° LigneFinal ] ou bien print (Id_DataFrame.iloc [ N° LigneInitial : N° LigneFinal ] ) Id_DataFrame.iloc[ N° Ligne, N° Colonne ou bien print (Id_DataFrame.iloc[N° Ligne, N° Colonne])</pre> |
| Modification et ajout des données dans un DataFrame                    | <pre>Id_DataFrame.loc[N° ligne, "Nom_colonne"]=Valeur ( ou Formule) Id_DataFrame.loc[N° ligne]=[Liste_Valeur] Id_DataFrame["Id_Colonne"]=valeur ou (Formule)</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Nettoyage d'un DataFrame                                               | <pre>Résultat = Id_DataFrame . drop_duplicates ( ) Résultat = Id_DataFrame . dropna ( ) Résultat = Id_DataFrame . dropna ( axis=1 )</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Les fonctions statistiques                                             | <pre>mean ( ) – min ( ) - max ( )- sum ( ) – count ( )</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| Affichage des données d'un DataFrame selon une ou plusieurs conditions | <pre>print(Id_DataFrame[Id_DataFrame["Nom_Colonne"] Op_Comparaison valeur]) print(Id_DataFrame[(Id_DataFrame["Nom_Colonne"] Op_Comp Valeur) Op_Logique (Id_DataFrame["Nom_Colonne"] Op_Comp Valeur)])</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |

|                                 |                                                                                                                                                                                                                                                                                                                                                                           |
|---------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tri des éléments d'un DataFrame | <code>DataFrame_Trié = Id_DataFrame.sort_values(by = ["Nom_Colonne1", "Nom_Colonne2", ...], ascending = [ True/False, True/False, ...])</code>                                                                                                                                                                                                                            |
| Création d'un graphique         | <code>import matplotlib.pyplot as alias_matplotlib<br/>Id_DataFrame.plot.bar(x="Nom_Colonne_Abscisse",<br/>y="Nom_Colonne_Ordonnée", title="Titre_Graphique",<br/>color="couleur_Graphique")<br/>Id_DataFrame.plot.line(x="Nom_Colonne_Abscisse",<br/>y="Nom_Colonne_Ordonnée",<br/>title="Titre_Graphique",color="couleur_Graphique")<br/>alias_matplotlib.show()</code> |