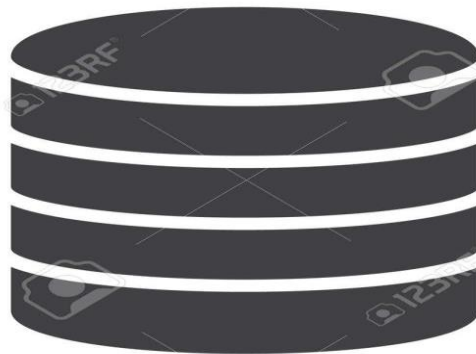




Projet : Base De Données



Réalisé par :

OUNDIR IMAD

Encadré par :

Mme. MHAIDRAT LEILA

M. AKOUAD SAMY

Plan :

- I. Introduction Base de données :
- II. Les types de bases de données :
- III. Pourquoi on utilise une base de données :
- IV. Les meilleurs logiciels de base données :
- V. Comprendre le contexte du projet :
 - ✓ Tache1 : Segmenter les taches entre nous.
 - ✓ Tache2 : Gestion des passagers.
 - ✓ Tache3 : Customer.
 - ✓ Exercice d'application.

I- Introduction Base de données :

a- C'est quoi une base de données en informatique ?

- Les bases de données informatiques sont utilisées dans un grand nombre d'entreprises pour stocker, organiser et analyser les données. ... Une base de données (que nous nommerons BDD par commodité) est une collection d'informations organisées afin d'être facilement consultables, gérables et mises à jour.



II- Les types de bases de données :

- **Parmi ces types :**

1- La base de données réseau :

Cette première évolution de la base de données hiérarchique permet des relations n-m (c'est à dire plusieurs parents/plusieurs enfants, contrairement à la base de données hiérarchique qui ne comprenait qu'un seul parent).

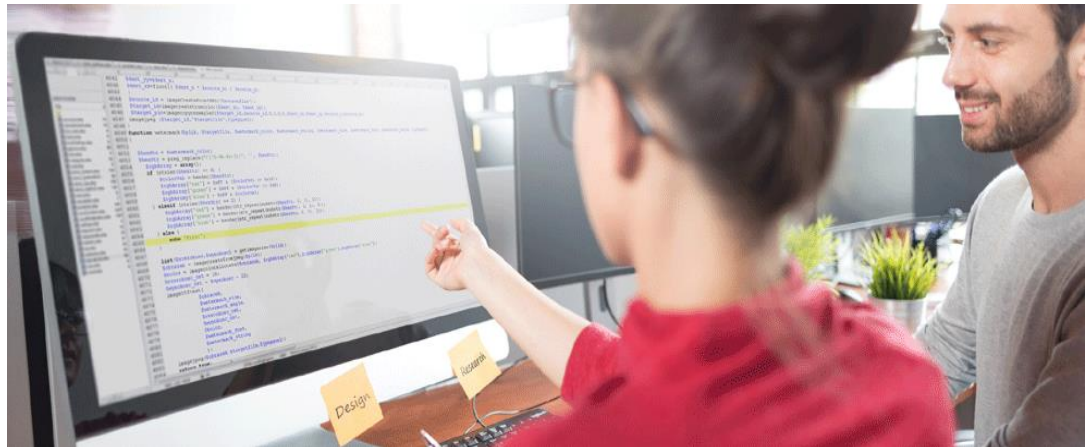
2- La base de données relationnelle :

C'est le type de base de données le plus répandu aujourd'hui. Basé sur l'algèbre relationnel, il est devenu la « norme » en informatique ces trente dernières années.

3- La base de données objet

Ce type de base de données n'existe pas encore de façon officielle et reste à l'heure actuelle au stade de « sujet de laboratoire ».

III- Pourquoi on utilise une base de données :



- Les bases de données permettent aux utilisateurs de centraliser et partagés leurs informations à tout moment. ... L'information est également obtenue plus rapidement et plus facilement, ce qui simplifie grandement le travail et vous permet d'avoir un meilleur service à la clientèle.

IV- Les meilleurs logiciels de base données :



A- Les 5 meilleurs logiciels de base de données

- MySQL. Véritable référence, MySQL est de loin le plus populaire des **logiciels de base de données** pour applications web. ...
- PostgreSQL. C'est « l'ancêtre » des **logiciels de base de données**. ...
- SAP HANA. ...
- Mongo DB. ...
- Microsoft SQL Server.

V- Comprendre le contexte du projet :

- Tache1 : « Segmenter les taches entre nous » :

Nous avons eu une petite réunion nous les membres du groupe, avons divise les taches et avons décidé de travailler avec **Trello** c'est une application de gestion de projet.

a- Définition Trello :

- **Trello** est un outil de gestion de projet en ligne, lancé en septembre 2011 et inspiré par la méthode Kanban de Toyota. Il repose sur une organisation des projets en planches listant des cartes, chacune représentant des tâches.

- Exemple Trello :



• Tache2 : Gestion des passagers :

On voudrait ajouter une table de passagers à la base BDVols. Un passager est décrit par son numéro, son nom et son prénom. Ces attributs seront respectivement Représentés par :
- Première étapes nous avons commencé par la création de la base de données, après nous avons créé un tableau s'appelle "passagers" dans ce tableau nous avons créé les trois champs NumeroP - Nom - Prénom

✓ Avion (NumeroA, Compagnie, Constructeur, Modèle, Capacité):

a-1 syntaxe pour créer tableau Avion:

```
CREATE TABLE Avions(NumeroA int(3) PRIMARY KEY, compagnie varchar(11), constructeur varchar(11), modele varchar(11), capacite int(10) );
```

✓ Horaire(NumeroH, VilleDépart, VilleArrivée, HeureDépart, HeureArrivée):

a-2 syntaxe pour créer tableau Horaire:

```
CREATE TABLE Horaire(NumeroH int(3) PRIMARY KEY, VilleDepart varchar(11), VilleArrivee(11), HeureDepart int(3), HeureArrivee int(3));
```

✓ Vol(NumeroV, JourSem, Jour, PlacesLibres, #Avion, #Horaire):

a-3 syntaxe pour créer tableau Vol:

```
use bdvols;
CREATE TABLE vol
(
    numeroV int PRIMARY KEY,
    jourSem Varchar(20),
    jour DATE,
    placeLibre INT ,
    avion INT(3),
    horair INT(3),
    CONSTRAINT avion_fk FOREIGN KEY (avion)
    REFERENCES avions (NumeroA),
    CONSTRAINT horair_fk FOREIGN KEY (horair)
    REFERENCES horaire (Horaire)
);
```

a-4 syntaxe pour créer base de données:

```
CREATE DATABASE BDVols;
```

a-5 syntaxe pour créer un champ NumeroP de type INT , créer un champ Nom de type VARCHAR(20).un champ Prénom de type VARCHAR(20):

```
USE BDVols ;
CREATE TABLE passagers (numeroP int (4),
nom varchar(20), prenom varchar (20));
```

II- ajoutez trois passagers dans la table :

- Adjonction des passagers Avec une seule requête INSERT INTO, ajoutez trois passagers dans la table de manière à obtenir :**NumeroP NOM PRENOM**:

- 1 -Thirion Eric
- 2- Gaston Lagaffe
- 3 -Clapton Eric

b-1 syntaxe pour insérer les trois passagers:

```
USE bdvols;
INSERT INTO passagers(numeroP,nom,prenom)
VALUES(1," Thirion "," Eric ");
INSERT INTO passagers(numeroP,nom,prenom)
VALUES(2," Gaston "," Lagaffe ");
INSERT INTO passagers(numeroP,nom,prenom)
VALUES(3," Clapton "," Eric ");
```

III- Correction de l'erreur « Nom et Prénom » :

Correction de l'erreur à la deuxième ligne Dans la deuxième ligne, le nom et le prénom sont inversés. Corrigez cette erreur en utilisant les requêtes DELETE et INSERT. « 2 Lagaffe Gaston »

c-1 syntaxe pour correction de l'erreur de la deuxième ligne :

*premièrement supprimé le champ :

```
USE bdvols;

delet from passagers where numeoP =2;

INSERT INTO passagers (nom,prenom)

VALUES ("Lagaffe "," Gaston ");
```

IV-Modification des noms et des prénoms :

- Modification des noms et des prénoms Sans utiliser les numéros des passagers et en utilisant uniquement la requête UPDATE, modifiez la table passager afin d'obtenir le résultat suivant :**NumeroP-Nom -Prenom**

- 1-Thirion Alex
- 2-Deferre Gaston
- 3- Clapton Alex

```
USE bdvols;
update passagers SET nom=" Thirion ", penom =" Alex ";
where id=1
update passagers SET nom=" Deferre ", penom =" Gaston ";
where id=2
```

```
update passagers SET nom=" Clapton ", penom =" Alex ";  
where id=3
```

- **Tache3 : Customer :**

1 : Sélection de colonnes Afficher une table contenant les colonnes CustomerName, Adress et City de la table Customers :

```
SELECT Customers (CustomerName, Adress, City )  
From Customers;
```

2 : Sélection de lignes Afficher une table contenant les lignes de la table Customers dont le City est London :

```
SELECT * FROM Customers where city ='London';
```

3 : Sélection multicritère Afficher les lignes de la table Customers dont lCountry est UK, City est London et CustomerID est supérieur de 19 :

```
SELECT * FROM Customers where country='UK' AND City='London' AND CustomerID>19 ;
```


• Exercice d'application :

➤ Voilà un modèle relationnel de la base infoPers:

- Ajouter un tableau s'appelle InfoProfessionnel dans ce tableau il faut saisir les champs :
- Un champ NumeroP .
- Un champ Nom.
- Un champ prenom.
- Un champ adresse.
- Un champ ville .
- Un champ telephone.

1-CREATE DATABASE infoPers;

2-CREATE TABLE myinfos (id int(10),nom varchar(20),prenom varchar(20),ville varchar(20),
telephone int(20));

3-INSERT INTO myinfos(id,nom,prenom,ville,telephone)
VALUES(1,"Imadani","noureddine","tinghir",0682275084);

INSERT INTO myinfos(id,nom,prenom,ville,telephone)
VALUES(2,"OUNDIR","IMAD","OUJDA",0635121357);

INSERT INTO myinfos(id,nom,prenom,ville,telephone)
VALUES(3,"BOUJEMA","YOUSSEF","OUJDA",0777630753);

INSERT INTO myinfos(id,nom,prenom,ville,telephone)
VALUES(4,"GROUZ","KHALIL","OUJDA",0644362697);

INSERT INTO myinfos(id,nom,prenom,ville,telephone)
VALUES(5,"CHAIBI","IMRANE","OUJDA",0632823950);

4-ajouter champs adresse et age:

ALTER TABLE myinfos ADD(age int(10), address varchar(20));

5-ajouter valeurs:

UPDATE myinfos

set age=22,address="doha"

WHERE id=5;

FIN

*Merci beaucoup pour
votre attention!*