

Les « Cuisines de Colmar » ont pour activité la fabrication et la livraison de plats cuisinés. Vous êtes chargé de l’exploitation de la base de données et votre mission va consister à réaliser un certain nombre de requêtes en langage SQL.

Vous disposez de la base de données sous MySQL ainsi que des compositions des tables en fin de fichier.

**Ce travail sera à rendre individuellement sur Moodle pour le 3 avril 2024 minuit.**

**Schéma relationnel de l’application**

**MLD**

ATELIER\_CUISINE (id, TelCuisine, idCuisinierResponsable)

Clef primaire : id

Clef étrangère : idCuisinierResponsable en référence à id de la table CUISINIER

CATEGORIE(id, libelle)

Clef primaire : id

CLIENT( id, nom, prenom, adresseRue, cp, ville)

Clef primaire : id

COMMANDE (id,DateCom,DateLivraison, idClient)

Clef primaire : id

Clef étrangère : idClient en référence à id de la table CLIENT

CUISINIER (id, nom, dateEmb, idCuisine)

Clef primaire : id

Clef étrangère : idCuisine en référence à id de la table ATELIER\_CUISINE

MENU (id,PrixMenu)

Clef primaire : id

PLAT (id, NomPlat, idCategorie)

Clef primaire : id

Clef étrangère : idCategorie en référence à id de la table CATEGORIE

COMPOSE (idMenu, idPlat)

Clef primaire : idMenu, idPlat

Clef étrangère : idMenu en référence à id de la table MENU

idPlat en référence à id de la table PLAT

LIGNE\_COMMANDE (idCommande, idMenu,QteCom)

Clef primaire : idCommande, idPlat

Clef étrangère : idMenu en référence à id de la table MENU

idCommande en référence à id de la table COMMANDE

PREPARER (idCuisine, idPlat, datePreparation, nombrePlats)

Clef primaire : idCuisine, idPlat, datePreparation

Clef étrangère : idCuisine en référence à id de la table ATELIER\_CUISINE

idPlat en référence à id de la table PLAT

MCD

Une image contenant texte, capture d’écran, diagramme, Police

Description générée automatiquement

# Partie A – Révisions sur les requêtes

1. Combien existe-t-il de plats par numéro de catégorie ?

SELECT idCategorie, COUNT(id) AS nb\_plats

FROM plat

GROUP BY idCategorie

A screenshot of a cell phone

Description automatically generated

1. Combien existe-t-il de plats par nom de catégorie ?

SELECT C.libelle AS Libellé\_cat, COUNT(P.id) AS nb\_plats\_par\_cat

FROM plat P

JOIN categorie C ON C.id = P.idCategorie

GROUP BY C.libelle

A screenshot of a computer

Description automatically generated

1. Quel est le menu le plus cher, le menu le moins cher et l’écart de prix entre les deux ? En une seule requête.

SELECT MAX(prixmenu) AS prix\_max, MIN(prixmenu) AS prix\_min, MAX(prixmenu) - MIN(prixmenu) AS ecart\_min\_max

FROM menu;

A close up of numbers

Description automatically generated

1. Quel est le chiffre d’affaires par nom de client ?

SELECT Cl.nom, SUM(prixmenu\*LC.qtecom) as Chiffre\_daffaires

FROM client Cl

JOIN commande C ON Cl.id = C.idClient

JOIN ligne\_commande LC ON LC.idCommande = C.id

JOIN menu M ON M.id = LC.idMenu

GROUP BY Cl.nom

A screenshot of a computer

Description automatically generated

1. Même question, mais en ne retenant que les clients qui ont un CA supérieur à 200 €

SELECT Cl.nom, SUM(prixmenu\*LC.qtecom) as `Chiffre d'affaires`

FROM client Cl

JOIN commande C ON Cl.id = C.idClient

JOIN ligne\_commande LC ON LC.idCommande = C.id

JOIN menu M ON M.id = LC.idMenu

GROUP BY Cl.nom

HAVING SUM(prixmenu\*LC.qtecom) > 200;

A screenshot of a computer

Description automatically generated

1. Combien de salariés ont été embauchés par année ?

SELECT YEAR(dateEmb) AS Ancienneté, COUNT(id) AS nb\_cuisiniers

FROM cuisinier CU

GROUP BY YEAR(dateEmb)

ORDER BY YEAR(dateEmb)

A screenshot of a table

Description automatically generated

1. Quels sont les menus (identifiant du menu) qui comportent plus de 4 plats ?

SELECT idMenu, COUNT(idPlat) AS nb\_plats

FROM compose

GROUP BY idMenu

HAVING COUNT(idPlat) > 4

A screenshot of a computer

Description automatically generated

1. On désire connaître le nombre d’années d’ancienneté par cuisinier (afficher le nom du cuisinier).

SELECT nom, YEAR(CURRENT\_DATE) - YEAR(dateEmb) AS ancienneté

FROM cuisinier;

A screenshot of a phone

Description automatically generated

1. Combien y a-t-il de cuisiniers par cuisine et par année d’ancienneté ?

SELECT A.id AS idCuisine, YEAR(C.dateEmb) AS Ancienneté, COUNT(C.id) AS nombre\_cuisiniers

FROM atelier\_cuisine A

JOIN cuisinier C ON A.id = C.idCuisine

GROUP BY A.id, YEAR(C.dateEmb)

A screenshot of a computer

Description automatically generated

1. Combien de menus différents ont été livré le 15 mars ?

SELECT COUNT(DISTINCT M.id) AS Différents\_Menus

FROM menu M

JOIN ligne\_commande LC ON M.id = LC.idMenu

JOIN commande C ON LC.idCommande = C.id

WHERE DAY(C.datelivraison) = 15 AND MONTH(C.datelivraison) = 3;

A close-up of a text

Description automatically generated

1. Combien de plats doit-on fabriquer par jour ?

SELECT ROUND(AVG(total\_journalier)) AS Plats\_A\_Fabriquer

FROM (

SELECT datePreparation, SUM(nombrePlats) AS total\_journalier

FROM PREPARER

GROUP BY datePreparation

) AS plats\_par\_jour;

A close-up of a red circle

Description automatically generated

1. Quel est le prix moyen d’un menu ?

SELECT ROUND(AVG(prixmenu), 2) AS Prix\_moyen\_menu

FROM menu

A close up of a number

Description automatically generated

# Partie B –Requêtes plus complexes

1. Quels sont les identifiants des clients qui ont commandé le menu numéro 1 ?

(Requête imbriquée)

Solution :

SELECT idClient

FROM COMMANDE

WHERE id IN (

SELECT idCommande

FROM ligne\_commande

WHERE idMenu = 1

);

A screenshot of a computer

Description automatically generated

1. Quels sont les noms des clients qui ont commandé le menu numéro 1 ?

(Requête imbriquée)

SELECT Cl.nom

FROM client Cl

WHERE Cl.id IN (

SELECT idClient

FROM commande Cm

WHERE Cm.id IN (

SELECT Lc.idCommande

FROM ligne\_commande Lc

WHERE idMenu = 1

)

);

A close up of words

Description automatically generated

1. Quel est le nom du responsable de l’atelier où travaille MENARDEAU ?

(Requête non imbriquée)

Solution à expliquer :

-- comme il y a 2 catégories de cuisiniers

-- on doit faire 2 jointures différentes

-- la 1ére identifie les responsables cuisiniers

-- la 2éme identifie les cuisiniers normaux

-- afin de trouver le responsable de la cuisine de MENARDEAU

-- on identifie d'abord l'atelier, puis le responsable

SELECT C\_Resp.nom

FROM cuisinier C\_Resp

JOIN atelier\_cuisine AC ON C\_Resp.id = AC.idCuisinierResponsable

JOIN cuisinier C ON AC.id = C.idCuisine

WHERE C.nom = 'MENARDEAU';

A close up of a word

Description automatically generated

1. Quels sont les clients qui habitent la même ville par ordre croissant des noms des clients (auto jointure) ?

SELECT c1.nom AS nom\_1, c2.nom nom\_2, c1.ville

FROM client c1

JOIN client c2 ON c1.ville = c2.ville

WHERE c1.nom != c2.nom

ORDER BY c1.nom, c2.nom

A table with many names

Description automatically generated

1. Quels sont les clients (noms) qui ont commandé des quantités supérieures à 10 (requête imbriquée) ?

SELECT C.nom

FROM client C

WHERE C.id IN (

SELECT CMD.idClient

FROM commande CMD

WHERE CMD.id IN (

SELECT LC.idCommande

FROM ligne\_commande LC

WHERE LC.qtecom > 10

)

)

A close up of a sign

Description automatically generated

1. Quels sont les cuisiniers (noms) qui ont préparé des plats le 15 mars ?

SELECT C.nom

FROM cuisinier C

WHERE C.idCuisine IN (

SELECT AC.id

FROM atelier\_cuisine AC

WHERE AC.id IN (

SELECT P.idCuisine

FROM preparer P

WHERE

MONTH(P.datePreparation) = 3 AND

DAY(P.datePreparation) = 15

)

)

A group of names on a white background

Description automatically generated

1. Quels sont les numéros et prix des menus préparés le 15 Mars ?

SELECT M.id, M.prixmenu

FROM menu M

WHERE M.id IN (

SELECT C.idMenu

FROM compose C

WHERE C.idPlat IN (

SELECT P.id

FROM plat P

WHERE P.id IN (

SELECT PR.idPlat

FROM preparer PR

WHERE

MONTH(PR.datePreparation) = 3 AND

DAY(PR.datePreparation) = 15

)

)

)

A screenshot of a cell phone

Description automatically generated

1. Quels sont les noms des clients qui ont pris un menu supérieur à 15 euros ?

SELECT C.nom

FROM client C

WHERE C.id IN (

SELECT CMD.idClient

FROM commande CMD

WHERE CMD.id IN (

SELECT LC.idCommande

FROM ligne\_commande LC

WHERE LC.idMenu IN (

SELECT M.id

FROM menu M

WHERE M.prixmenu > 15

)

)

)

A group of black text

Description automatically generated

# Quel est le menu le plus cher (requête imbriquée) ?

SELECT prixmenu

FROM menu

WHERE prixmenu IN (

SELECT MAX(prixmenu)

FROM menu

)

A screenshot of a computer

Description automatically generated

# Quels sont les menus qui ne sont pas les plus chers (requête imbriquée) ?

SELECT id, prixmenu

FROM menu

WHERE prixmenu < (

SELECT MAX(prixmenu)

FROM menu

);

A screenshot of a number

Description automatically generated

# Partie C – Requêtes actions

1. Un nouveau cuisinier a été embauché aujourd’hui : Joseph CHASLES, il travaille dans la cuisine n° 1 (Cherchez comment mettre la date du jour).

INSERT INTO cuisinier (nom, dateEmb, idCuisine)

VALUES ('CHASLES', CURRENT\_DATE, 1)

A close up of a word

Description automatically generated

1. Un nouveau client doit être saisi : DELAMARRE André. On ne connaît pas son adresse.

INSERT INTO client (nom, prenom)

VALUES ('DELAMARRE', 'André')



1. Quatre nouveaux plats doivent être saisis : ,mousse au chocolat, île flottante, salade de fruits, salade exotique. Ces plats sont classés dans la catégorie n°10.

INSERT INTO plat (nomplat, idCategorie)

VALUES ('mousse au chocolat', 10),

('île flottante', 10),

('salade de fruits', 10),

('salade exotique', 10);

A screenshot of a menu

Description automatically generated

1. Le plat n° 38 a changé de catégorie : il est maintenant en catégorie 10.

UPDATE plat

SET idCategorie = 10

WHERE id = 38;



1. Créez une table regroupant toutes les salades. Cette table a la même structure que la table PLAT. Appelez cette table « SALADE ». Insérez dans cette table toutes les salades.

-- creer la table

CREATE TABLE IF NOT EXISTS `salade` (

`id` int NOT NULL AUTO\_INCREMENT,

`nomplat` varchar(50) COLLATE utf8mb4\_general\_ci DEFAULT NULL,

`idCategorie` smallint DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `idCategorie` (`idCategorie`)

) ENGINE=InnoDB AUTO\_INCREMENT=49 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

ALTER TABLE `salade`

ADD CONSTRAINT `salade\_ibfk\_1` FOREIGN KEY (`idCategorie`) REFERENCES `categorie` (`id`);

INSERT INTO SALADE (id, nomplat, idCategorie)

SELECT id, nomplat, idCategorie

FROM PLAT

WHERE nomplat LIKE '%salade%';

A hand holding a piece of paper

Description automatically generated

A screenshot of a computer

Description automatically generated

Ceci sera effectué en deux requêtes, et il a 6 plats de salade ….

1. Effacez les enregistrements de cette table.

DELETE FROM salade

A screenshot of a chat

Description automatically generated

1. Enregistrez dans la table SALADE uniquement les salades servies en dessert (catégorie = 10), il en restera 3.

INSERT INTO salade (nomplat, idCategorie)

SELECT nomplat, idCategorie

FROM plat

WHERE idCategorie = 10 and nomplat LIKE '%salade%';

A screenshot of a computer

Description automatically generated

1. Le cuisinier n° 4 a démissionné. Modifiez la table cuisinier en conséquence.

DELETE FROM cuisinier

WHERE id = 4

-- Ou intialiser la date d’embauche à null

UPDATE cuisinier

SET dateEmb = NULL

WHERE id = 4;

1. Le menu n° 1 coûte maintenant 13 €.

UPDATE menu

SET prixmenu = 13

WHERE id = 1 ;

A screenshot of a computer

Description automatically generated

1. Le client n°6 a déménagé : il habite maintenant 56, bld de la Liberté – 35000 RENNES.

UPDATE client

SET adresseRue = '56, bld de la Liberté', cp = 35000, ville = 'RENNES'

WHERE id = 6;

1. Le plat n°1 n’existe plus. Que se passe-t-il lorsque nous essayons de le supprimer ? Pourquoi ?

La requête n’aucun effet sur la base de données si la donnée en question à supprimer existe dans une autre table tant qu’une clé étrangère.

1. Les prix des menus ont augmenté de 3%. Effectuez cette modification.

UPDATE menu

SET prixmenu = prixmenu \* 1.03;

1. Supprimez cette augmentation

UPDATE menu

SET prixmenu = prixmenu / 1.03;

1. Les menus qui contiennent des plats de la catégorie pâtisserie subissent une augmentation de 5%.

UPDATE menu

SET prixmenu = prixmenu \* 1.05

WHERE id IN (

SELECT DISTINCT C.idMenu

FROM compose C

JOIN plat P ON C.idPlat = P.id

JOIN categorie Cat ON P.idCategorie = Cat.id

WHERE Cat.libelle = 'pâtisserie'

);

Table ATELIER\_CUISINE

Table COMMANDE

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Table CATEGORIE

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Table COMPOSE

Une image contenant texte, capture d’écran, nombre

Description générée automatiquement

Une image contenant texte, capture d’écran, nombre

Description générée automatiquement

Une image contenant texte, capture d’écran, Police, blanc

Description générée automatiquement

Table CLIENT

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Table CUISINIER

Table Ligne\_Commande

Une image contenant texte, capture d’écran, nombre

Description générée automatiquement

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Table MENU

Une image contenant texte, capture d’écran, nombre

Description générée automatiquement

Table PLAT

Une image contenant texte, capture d’écran, nombre, document

Description générée automatiquement

Une image contenant texte, capture d’écran, nombre, document

Description générée automatiquement

Table PREPARER

Une image contenant texte, capture d’écran, nombre, menu

Description générée automatiquement

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement