EXO 1

CREATE TABLE utilisateur

(

  id INT PRIMARY KEY NOT NULL AUTO\_INCREMENT,

    titre VARCHAR(100),

durée\_en\_minute INT

date\_de\_sortie INT

)

INSERT INTO utilisateur (titre, duree\_en\_minute, date\_de\_sortie)

VALUES

('The Gentlemen', 113, 2020),

('Kuzco', 75, 2000),

('Joker', 122, 2019),

('Sweeney todd', 116, 2007),

('Pincesse mononoké', 134, 1997);

DELETE FROM utilisateur

WHERE date\_de\_sortie < '2010’ AND durée\_en\_minute < '120'

EXO 2

1 :

SELECT ville\_nom\_reel, ville\_surface

FROM `villes\_france\_free`

ORDER BY ville\_surface ASC

LIMIT 5

2 :

SELECT ville\_code\_postal, ville\_population\_2010

FROM `villes\_france\_free`

ORDER BY ville\_population\_2010 DESC

LIMIT 15

3 :

SELECT ville\_departement, ville\_nom\_reel

FROM `villes\_france\_free`

WHERE ville\_nom\_reel LIKE "p%"

4 :

SELECT ville\_departement, SUM(ville\_population\_2012)

FROM villes\_france\_free

GROUP BY ville\_departement

5 :

SELECT ville\_nom\_reel

FROM `villes\_france\_free`

WHERE ville\_population\_1999 > 20000

6 :

UPDATE villes\_france\_free

SET ville\_longitude\_dms = REPLACE(ville\_longitude\_dms, '+', '\*');

7 :

SELECT REVERSE (departement\_nom) AS 'nom\_inverse'

FROM departement

WHERE departement\_code LIKE "97%"

8 :

SELECT v.ville\_nom\_reel, v.ville\_surface, d.departement\_nom

FROM villes\_france\_free AS v

INNER JOIN departement AS d

ON ville\_id = departement\_id

ORDER BY ville\_surface DESC

LIMIT 5

9 :

SELECT v.ville\_departement, d.departement\_nom,

SUM(v.ville\_population\_2010) AS departement\_population

FROM villes\_france\_free AS v

LEFT JOIN departement AS d

ON v.ville\_departement = d.departement\_code

GROUP BY v.ville\_departement, d.departement\_nom

ORDER BY `departement\_population` DESC LIMIT 5;

EXO 3

1

SELECT

Categories.nom,

Marques.nom,

Produits.modele

FROM Produits

JOIN Categories

ON Categories.id = Produits.categorie\_id

JOIN Marques

ON Marques.id = Produits.marque\_id;

2

DELETE FROM Produits

WHERE marque\_id = (

SELECT id FROM Marques WHERE nom = 'Indesit'

);

3

UPDATE `Produits`

SET `prix`= prix - 50

WHERE prix > 600;

Exo 4

1

SELECT p.nom AS personnage, m.nom AS metier

FROM personnage AS p

JOIN metier AS m ON p.idMetier = m.id;

2

SELECT p.nom AS personnage, m.nom AS metier

FROM personnage AS p

LEFT JOIN metier AS m ON p.idMetier = m.id;

3

SELECT m.nom

FROM metier AS m

LEFT JOIN personnage AS p

ON m.id = p.idmetier

WHERE p.id IS NULL;

EXO 5 :

SELECT p.produit, AVG(note) AS note\_moyenne

FROM produit AS p

LEFT JOIN note ON p.id = note.produit\_id

GROUP BY p.produit;