

The background is a solid blue color. It is decorated with various abstract geometric elements: a vertical dashed white line on the left; a grid of small purple dots in the top left; concentric purple circles at the top center; a horizontal line with a small white circle at its left end in the top right; a dark blue and purple quarter-circle shape on the right; a horizontal line with a small black circle at its left end below the quarter-circle; a dotted purple circle and a solid white circle on the bottom left; a horizontal purple bar with a black dot at its right end below the dotted circle; three small dots (white, purple, purple) at the bottom center; a series of purple diagonal slashes on the bottom right; and two vertical bars, one dark blue and one white, on the far right.

MYSQL 2ITECH



Plan de cours



.....



Mise en place de l'environnement

- Démarrer un conteneur docker mysql 8
- Accessible port 3306
- Volume persistant
- Nom du conteneur: bdd
- Mode détaché
- Environnement: paramétrer MYSQL_ROOT_PASSWORD

Rappels: docker exec pour prendre le contrôle du conteneur.

Création database, user et table

```
Uploaded using RayThis Extension

-- Création d'une base de données;
CREATE DATABASE <name>;

-- Création d'un utilisateur
CREATE USER 'nouveau_utilisateur'@'localhost' IDENTIFIED BY 'mot_de_passe';
GRANT ALL PRIVILEGES ON <database_name>.* TO 'user'@'localhost';
FLUSH PRIVILEGES;

-- Suppression d'un utilisateur
DROP USER 'utilisateur'@'localhost';

-- Afficher les bases de données
SHOW DATABASES;

-- Suppression d'une base de données
DROP DATABASE <database_name>;

-- utiliser une base de données
USE <database_name>;
```

```
Uploaded using RayThis Extension

-- Créer une table
CREATE TABLE <table name> (
  id INT AUTO_INCREMENT PRIMARY KEY, -- obligatoire
  <column name> <data type>,
  <column name> <data type>
);

-- lister les tables
SHOW TABLES;

-- lister les colonnes d'une table
DESCRIBE <table name>;

-- supprimer une table
DROP TABLE <table name>;

-- types de données
INT
DOUBLE
VARCHAR
TEXT
DATE
TIME
DATETIME
ENUM
SET
BOOLEAN (synonyme de TINYINT(1))
```

Contraintes

```
Uploaded using RayThis Extension

-- spécifier une contrainte a la création de la table
CREATE TABLE IF NOT EXISTS `users` (
  -- PRIMARY KEY: la clé primaire est forcément unique, AUTO_INCREMENT: auto-incrémenté,
  id INT AUTO_INCREMENT PRIMARY KEY,
  -- NOT NULL = obligatoire
  username varchar(255) NOT NULL,
  -- UNIQUE = unique au sein de la table
  email varchar(255) NOT NULL UNIQUE,
  -- ENUM = liste de valeurs possibles, DEFAULT = valeur par défaut
  role ENUM('admin', 'author', 'subscriber') NOT NULL DEFAULT 'subscriber',
  -- CHECK = contrainte de valeur + condition
  age INT CHECK (age >= 18),
  -- CHECK = contrainte de valeur dans un tableau
  gender char(1) CHECK (gender IN ('m', 'f')),
  -- DATETIME = date et heure, CURRENT_TIMESTAMP = valeur par défaut
  created_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
  -- ON UPDATE CURRENT_TIMESTAMP = mise à jour automatique
  updated_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  -- contraintes nommées en fin de ligne (bonne pratique)
  CONSTRAINT users_username_unique UNIQUE (username) -- UNIQUE = unique au sein de la table
);
```

Modifier la structure d'une table

```
Uploaded using RayThis Extension

-- Ajout d'une nouvelle colonne "address"
ALTER TABLE users ADD address VARCHAR(255);

-- Modification du type de données de la colonne "role" de VARCHAR à ENUM
ALTER TABLE users MODIFY role ENUM('admin', 'author', 'subscriber');

-- Ajout d'une contrainte UNIQUE à la colonne "age"
ALTER TABLE user ADD CONSTRAINT age_unique UNIQUE (age);

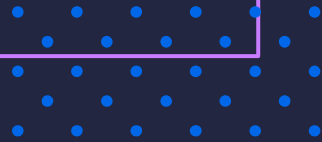
-- Renommage de la colonne "email" en "user_email"
ALTER TABLE users CHANGE email user_email VARCHAR(255);

-- Suppression de la colonne "address"
ALTER TABLE user DROP COLUMN address;

-- Modification de la taille de la colonne "username"
ALTER TABLE user MODIFY username VARCHAR(100);

-- Suppression de la contrainte UNIQUE de la colonne "age"
ALTER TABLE user DROP INDEX age_unique;
```

Insertion de données



Uploaded using RayThis Extension

```
INSERT INTO <table name> (column1, column2, ...) VALUES (value1, value2, ...);
```



Lecture de données

```
Uploaded using RayThis Extension

-- tous les champs, toutes les lignes
SELECT * FROM <table>;

-- champs sélectionnés, toutes les lignes
SELECT <field1>, <field2>, ... FROM <table>;

-- tous les champs, lignes sélectionnées
SELECT * FROM <table> WHERE <condition>;

-- comparaison sur l'id
SELECT * FROM <table> WHERE id = 1;

-- multiple conditions
SELECT * FROM <table> WHERE id = 1 AND name = 'toto';

-- OR
SELECT * FROM <table> WHERE id = 1 OR name = 'toto';

-- comparaison complexe sur un champ de texte
SELECT * FROM <table> WHERE name LIKE 'toto%'; -- commence par toto
-- % = n'importe quoi
```

```
Uploaded using RayThis Extension

-- Autres opérateurs

-- =, <, >, <=, >=, BETWEEN, IN, NOT IN, IS NULL, IS NOT NULL
-- AND, OR, NOT

-- Valeurs de comparaison utiles
-- '%' = n'importe quoi
-- '_' = un seul caractère, n'importe lequel
-- 'NOT NULL' = non nul
-- 'IS NULL' = nul
-- 'IN' = dans une liste
-- BETWEEN = entre deux valeurs

-- Exemples
SELECT * FROM user WHERE (age > 18 AND role = "subscriber") OR role = "admin";
SELECT * FROM user WHERE role IS NULL;
SELECT * FROM user WHERE role IN ("admin", "author");
SELECT * FROM user WHERE age BETWEEN 18 AND 25;
```


Mise à jour / Suppression

Uploaded using RayThis Extension

```
-- UPDATE et SET sont les 2 mots clés permettant la mise à jour
UPDATE <table> SET <column> = <value>; -- met à jour toutes les entrées de la table

-- Par défaut, UPDATE/SET s'appliquent à toutes la table,
-- L'utilisation de WHERE permet de cibler les mise à jour et s'utilise comme pour le SELECT
UPDATE users SET role = 'admin' WHERE id = 1;

-- vous pouvez spécifier plusieurs colonnes
UPDATE products SET price = 19.99, stock_quantity = 100 WHERE product_id = 1001;

-- Les expressions fonctionnent comme pour le INSERT
UPDATE orders
SET total_amount = (unit_price * quantity) --récupère les valeurs actuelles de unit_price et
quantity dans la table
WHERE order_id = 500;
```

Uploaded using RayThis Extension

```
-- supprime toutes les entrées de la table
DELETE FROM <table>;

-- Le where permet de restreindre les suppressions
-- fonctionnent comme pour SELECT et UPDATE
DELETE FROM employees WHERE employee_id = 101;
```

ORDER BY

```
Uploaded using RayThis Extension

-- Tri par ordre ascendant sur un texte
SELECT * FROM products ORDER BY product_name ASC;

-- Tri par ordre descendant sur un texte
SELECT * FROM products ORDER BY product_name DESC;

-- Tri par ordre ascendant sur un nombre (même résultat que précédemment)
SELECT * FROM products ORDER BY product_price ASC;

-- Tri sur plusieurs colonnes
SELECT * FROM products ORDER BY product_price ASC, product_name DESC;

-- Tri en ignorant la casse
SELECT * FROM products ORDER BY product_name COLLATE NOCASE ASC;

-- Tri en ignorant les accents
SELECT * FROM products ORDER BY product_name COLLATE NOACCENTS ASC;

-- Tri en ignorant les accents et la casse
SELECT * FROM products ORDER BY product_name COLLATE NOACCENTS NOCASE ASC;

-- Tri avec expression
SELECT * FROM products ORDER BY price * discount ASC;

-- Tri avec filtre
SELECT * FROM products WHERE price > 10 ORDER BY price * discount ASC;
```

Les fonctions mathématiques



Uploaded using RayThis Extension

```
-- Compter le nombre d'employés dans la table "employees"
```

```
SELECT COUNT(*) FROM employees;
```

```
-- Calculer la moyenne des salaires des employés
```

```
SELECT AVG(salary) FROM employees;
```

```
-- Arrondir la moyenne des salaires à deux décimales
```

```
SELECT ROUND(AVG(salary), 2) FROM employees;
```

Les alias



Uploaded using RayThis Extension

-- Alias de colonne pour rendre le résultat plus lisible

```
SELECT first_name AS prénom, last_name AS nom FROM employees;
```

-- Alias pour une valeur calculée

```
SELECT order_id, (unit_price * quantity) AS total FROM order_details;
```

-- Alias pour renommer le résultat de la fonction mathématique d'agrégation

```
SELECT AVG(salary) AS moyenne_salaire FROM employees;
```

Les views

Uploaded using RayThis Extension

```
-- créer une vue simple, avec uniquement les colonnes spécifiées
CREATE VIEW vue_clients AS
  SELECT customer_id, customer_name, email
  FROM customers;

-- créer une vue avec une clause WHERE
CREATE VIEW vue_clients AS
  SELECT customer_id, customer_name, email
  FROM customers
  WHERE customer_id > 1000;

-- créer une vue avec une clause ORDER BY
CREATE VIEW vue_clients AS
  SELECT customer_id, customer_name, email
  FROM customers
  ORDER BY customer_name;

-- Modifier une vue
ALTER VIEW vue_clients AS
  SELECT customer_id, customer_name, email

-- Supprimer une vue
DROP VIEW vue_clients;

-- Afficher les vues existantes
SHOW FULL TABLES IN classicmodels WHERE TABLE_TYPE LIKE 'VIEW';
```

Relations et clés étrangères

```
Uploaded using RayThis Extension

CREATE TABLE orders (
  id INT PRIMARY KEY,
  order_date DATETIME,
  status VARCHAR(20),
);

CREATE TABLE order_details (
  id INT PRIMARY KEY,
  order_id INT, -- Clé étrangère
  quantity INT,
  price DECIMAL(10, 2),
  -- liaison avec la table orders sur la clé primaire id
  FOREIGN KEY (order_id) REFERENCES orders(id)
);

-- récupérer les commandes avec leurs détails
SELECT * FROM orders JOIN order_details ON orders.id = order_details.order_id;
```

Jouer avec les JOIN

Uploaded using RayThis Extension

```
CREATE TABLE orders (  
  id INT PRIMARY KEY,  
  order_date DATETIME,  
  status VARCHAR(20),  
);  
  
CREATE TABLE order_details (  
  id INT PRIMARY KEY,  
  order_id INT, -- Clé étrangère  
  quantity INT,  
  price DECIMAL(10, 2),  
  -- liaison avec la table orders sur la clé primaire id  
  FOREIGN KEY (order_id) REFERENCES orders(id)  
);  
  
-- récupérer les commandes avec leurs détails  
SELECT * FROM orders JOIN order_details ON orders.id = order_details.order_id;
```

Contraintes d'intégrité référentielles

```
Uploaded using RayThis Extension

-- valeurs par défaut
-- constante numérique ou chaîne de caractère
-- NULL
-- CURRENT_TIMESTAMP
-- CURRENT_DATE
-- CURRENT_TIME

-- Intégrité à la suppression d'une clé étrangère
-- ON DELETE CASCADE -- supprime les enregistrements liés
-- ON DELETE SET NULL -- met la valeur à NULL
-- ON DELETE SET DEFAULT -- met la valeur par défaut si elle existe sinon NULL
-- ON DELETE RESTRICT -- refuse la suppression
-- ON DELETE NO ACTION -- refuse la suppression

CREATE TABLE parent (
  id INT NOT NULL,
  PRIMARY KEY (id)
);

CREATE TABLE enfant (
  id INT NOT NULL,
  parent_id INT,
  PRIMARY KEY (id),
  FOREIGN KEY (parent_id) REFERENCES parent(id) ON DELETE CASCADE -- si on supprime un parent,
on supprime aussi les enfants
);

-- Intégrité à la mise à jour d'une clé étrangère
-- (rarement utile, seulement dans le cas où la clé primaire peut changer dans la table
référéncée)
-- ON UPDATE CASCADE
-- ON UPDATE SET NULL
-- ON UPDATE SET DEFAULT
-- ON UPDATE RESTRICT
-- ON UPDATE NO ACTION
```


Contraintes d'intégrité référentielles

```
Uploaded using RayThis Extension

-- valeurs par défaut
-- constante numérique ou chaîne de caractère
-- NULL
-- CURRENT_TIMESTAMP
-- CURRENT_DATE
-- CURRENT_TIME

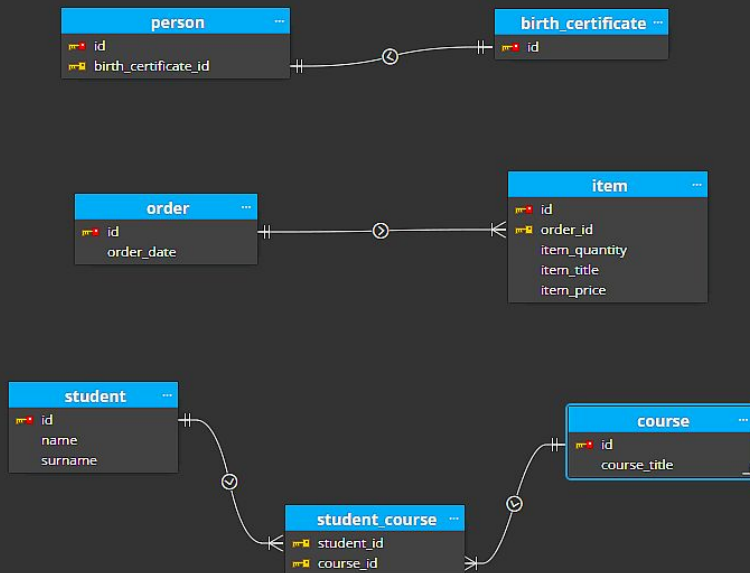
-- Intégrité à la suppression d'une clé étrangère
-- ON DELETE CASCADE -- supprime les enregistrements liés
-- ON DELETE SET NULL -- met la valeur à NULL
-- ON DELETE SET DEFAULT -- met la valeur par défaut si elle existe sinon NULL
-- ON DELETE RESTRICT -- refuse la suppression
-- ON DELETE NO ACTION -- refuse la suppression

CREATE TABLE parent (
  id INT NOT NULL,
  PRIMARY KEY (id)
);









CREATE TABLE enfant (
  id INT NOT NULL,
  parent_id INT,
  PRIMARY KEY (id),
  FOREIGN KEY (parent_id) REFERENCES parent(id) ON DELETE CASCADE -- si on supprime un parent,
on supprime aussi les enfants
);

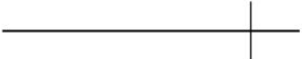

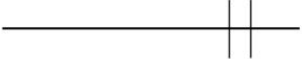



-- Intégrité à la mise à jour d'une clé étrangère
-- (rarement utile, seulement dans le cas où la clé primaire peut changer dans la table
référéncée)
-- ON UPDATE CASCADE
-- ON UPDATE SET NULL
-- ON UPDATE SET DEFAULT
-- ON UPDATE RESTRICT
-- ON UPDATE NO ACTION
```

Théorie: type de relations



Théorie: type de relations

1 : 1	
1 : 0..1	
1 : N	
1 : 1..N	
1 : 0..N	
N : N	
1..N : 1..N	
0..N : 0..N	

Symbol	Meaning	Number
	One	N/A
	Many	N/A
	Mandatory-One	Exactly one
	Optional-One	Zero or one
	Mandatory-Many	One or More
	Optional-Many	Zero or more