

# PHP

Programmation orientée objet et bases de données



SQL

# SQL

1. État de vos connaissances
2. Structure d'un serveur SQL
3. Installation et mise en place d'une base de données
4. Compréhension des types
5. Création des tables
6. Insertion des données
7. Sélection des données
8. Notions de jointure
9. Gestion des données
10. Optimisation des requêtes

## Serveur

Schéma 1

Table 1

ID	Firstname	Lastname
1	David	Patiasvili
2	Chuck	Norris
3	Jean	Dupont

Table 2

ID	Contact_Id	Date	Statut	Amount
1	1	2020-12-12	Validé	100.00
2	1	2020-12-10	En attente	85.50
3	2	2020-11-12	En attente	200.00

Table 3

ID	Invoice_id	Name	Amount
1	1	Nuit d'hotel	100.00
2	2	Nuit d'hotel	65.00
3	2	Petit-déjeuner	20.50
4	3	Nuit d'hotel	155.00
5	3	Diner	45.00

Schéma 2

Table 1

ID	Firstname	Lastname
1	David	Patiasvili
2	Chuck	Norris
3	Jean	Dupont

Table 2

ID	Contact_Id	Date	Statut	Amount
1	1	2020-12-12	Validé	100.00
2	1	2020-12-10	En attente	85.50
3	2	2020-11-12	En attente	200.00

Table 3

ID	Invoice_id	Name	Amount
1	1	Nuit d'hotel	100.00
2	2	Nuit d'hotel	65.00
3	2	Petit-déjeuner	20.50
4	3	Nuit d'hotel	155.00
5	3	Diner	45.00

# Les différents types (numériques)

Type	Nombre d'octets	Minimum	Maximum
TINYINT	1	-128	127
SMALLINT	2	-32768	32767
MEDIUMINT	3	-8388608	8388607
INT	4	-2147483648	2147483647
BIGINT	8	-9223372036854775808	9223372036854775807

Nombres décimaux :

- NUMERIC et DECIMAL (PRECISION, NB DECIMAL) : **chaîne**
- FLOAT (PRECISION, NB DECIMAL) : **nombre approché**
- DOUBLE : **nombre approché**

**Les plus utilisés : INT, BIGINT, DECIMAL, FLOAT, DOUBLE**

# Les différents types (alphanumériques)

Texte	CHAR(5)	Mémoire requise	VARCHAR(5)
' '	' '	5 octets	' '
'tex'	'tex '	5 octets	'tex'
'texte'	'texte'	5 octets	'texte'
'texte trop long'	'texte'	5 octets	'texte'

## À savoir :

- VARCHAR/CHAR <= 255
- UTF-8 = 2 octets

Type	Longueur maximale	Mémoire occupée
TINYTEXT	2 <sup>8</sup> octets	Longueur de la chaîne + 1 octet
TEXT	2 <sup>16</sup> octets	Longueur de la chaîne + 2 octets
MEDIUMTEXT	2 <sup>24</sup> octets	Longueur de la chaîne + 3 octets
LONGTEXT	2 <sup>32</sup> octets	Longueur de la chaîne + 4 octets

# Les différents types (autres)

- **ENUM** : Valeurs parmi une liste de valeurs (valeurs ou labels) : 65535 valeurs possibles
- **SET** : Valeurs parmi une liste de valeurs

À NE PAS UTILISER

- **DATE** : YYYY-MM-DD
- **DATETIME** : YYYY-MM-DD HH:MM:SS
- **TIME** : HH:MM:SS
- **TIMESTAMP** : (s) depuis 1/1/1970



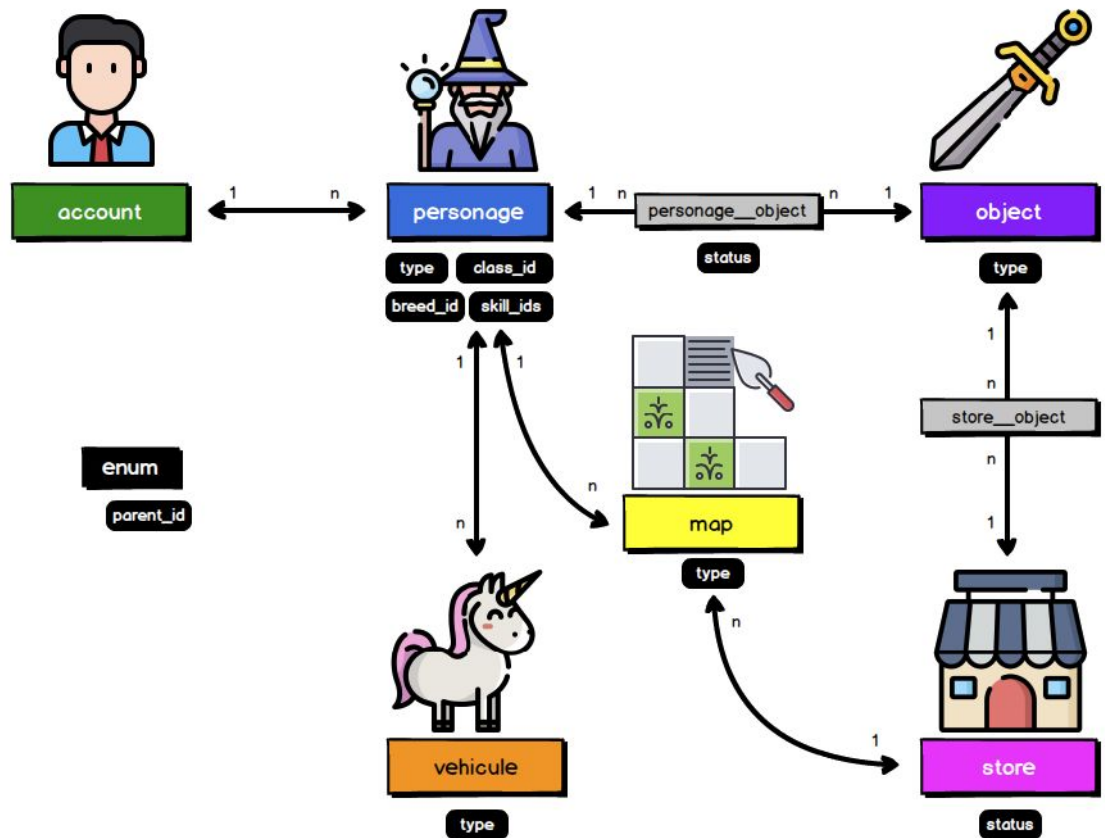
C'est parti !



# Cas pratique : Jeu RPG

Rappel des tables que nous avons décidé de créer

- account id INT, username VARCHAR(25), create\_date DATETIME, ip VARCHAR(15)
- enum id INT, parent\_id INT, slug TINYTEXT, label TINYTEXT
- map id INT, position\_x TINYINT, position\_y TINYINT, type INT, store\_id INT, enemy\_id INT
- object id INT, title TINYTEXT, type INT
- personnage id INT, account\_id INT, type INT, create\_date DATETIME, name VARCHAR(25), class\_id INT, breed\_id INT, skill\_ids TINYTEXT, health TINYINT, attack TINYINT, map\_id INT
- personnage\_\_object id INT, personnage\_id INT, object\_id INT, acquire\_date DATETIME, used\_date DATETIME, status INT
- store id INT, title TINYTEXT, status INT
- store\_\_object id INT, store\_id INT, object\_id INT, quantity SMALLINT
- vehicule id INT, personnage\_id INT, acquire\_date DATETIME, name VARCHAR(25), type INT, health TINYINT



### Requêtes à faire :

1. Récupérer l'ensemble des personnages pour un compte.
2. Récupérer l'ensemble des objets d'un personnage
3. Récupérer l'ensemble des véhicule d'un personnage qui sont des voiture.
4. Récupérer les tuiles de carte dans un secteur donnée (entre 0,0 et 10,10)
5. Avoir le nombre de personnage et d'objet par personnage, par compte.
6. Avoir le nombre de véhicule par type de personnage et par type de véhicule.
7. Récupérer la liste des tuile où se trouve un magasin ouvert vendant des objets que le personnage avaient mais n'a plus. (regrouper par objet, la liste des magasins)
- 8.

# Correction de l'exercice 1/3

- 1 . Récupérer l'ensemble des personnages pour un compte.

```
SELECT p.*  
FROM personnage p  
WHERE p.account_id = XX;
```

- 2 . Récupérer l'ensemble des objets d'un personnages

```
SELECT o.*  
FROM object o  
JOIN personnage__object po ON (po.personage_id = XX AND po.object_id = o.id);
```

- 3 . Récupérer l'ensemble des véhicule d'un personnage.

```
SELECT v.*  
FROM vehicule v  
JOIN enum vt ON (vt.id = v.type AND vt.slug = 'VEHICULE.TYPE.CAR')  
WHERE p.account_id = XX;
```

- 4 . Récupérer les tuiles de carte dans un secteur donnée (entre 0,0 et 10,10)

```
SELECT m.*  
FROM map m  
WHERE  
    position_x BETWEEN 0 AND 10  
    AND position_y BETWEEN 0 AND 10;
```

# Correction de l'exercice 2/3

5 . Avoir le nombre d'objet par personnage

```
SELECT
    p.id AS personnage_id,
    COUNT(*) AS count
FROM personnage p
RIGHT OUTER JOIN personnage__object po ON (po.personage_id = p.id)
RIGHT OUTER JOIN object o ON (o.id = po.object_id)
GROUP BY 1
```

6 . Avoir le nombre de véhicule par type de personnage et par type de véhicule.

```
SELECT
    pt.label AS pesonage_type,
    vt.label AS vehicule_type,
    COUNT(v.id) as count
FROM
    enum ppt
JOIN enum pvt ON (pvt.slug = 'VEHICULE_TYPE')
JOIN enum pt ON ( pt.parent_id = ppt.id)
JOIN enum vt ON (vt.parent_id = pvt.id)
LEFT OUTER JOIN personnage p ON (p.type = pt.id)
LEFT OUTER JOIN vehicule v ON (v.type = vt.id AND v.personage_id = p.id)
WHERE ppt.slug = 'PERSONAGE_TYPE'
GROUP BY 1, 2
```

# Correction de l'exercice 3/3

7 . Récupérer la liste des tuile où se trouve un magasins ouvert vendant des objets que l'utilisateur avaient mais n'a plus. (regrouper par objet, la liste des magasins)

```
SELECT
    o.id AS object_id,
    o.title AS object_title,
    GROUP_CONCAT(m.id, ',') AS map_ids
FROM map m
JOIN store s ON (s.id = m.store_id)
JOIN enum ss ON (ss.id = s.status AND slug = 'STORE_STATUS.OPEN')
JOIN store__object so ON (so.store_id = s.id AND quantity > 0)
JOIN object o ON (
    o.id = so.object_id AND o.id NOT IN (
        SELECT po.object_id
        FROM personage__object po
        JOIN enum pos ON (pos.id = po.status AND pos.slug = 'PERSONAGE__OBJECT_STATUS.STANDBY')
        WHERE po.personage_id = XX
    )
)
JOIN personage__object po ON (po.object_id = o.id AND po.personage_id = XX)
JOIN enum pos ON (pos.id = po.status AND pos.slug = 'PERSONAGE__OBJECT_STATUS.USED')
GROUP BY 1, 2
```