



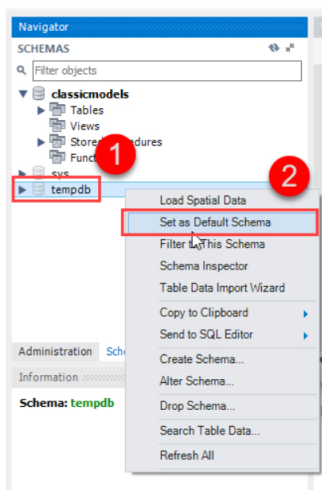
Utiliser une table et lire des données

Sélectionner une base de données

Pour sélectionner une base de données, il suffit de faire la commande suivante :

```
USE base_de_donnees;
```

Vous pouvez aussi la sélectionner dans `Workbench` en faisant :



Pour vérifier la base de données sur laquelle vous êtes :

```
select database();
```

Vous aurez par exemple `test`.

Une fois une base de données sélectionnée, vous n'avez plus besoin de la spécifier dans toutes les commandes `SQL` que vous allez faire.

Lire des données

Pour afficher toutes les données d'une table il suffit de faire :

```
SELECT * FROM table;
```

* signifie toutes les colonnes.

Pour n'afficher les données que de certaines colonnes, vous pouvez les spécifier en les séparant par des virgules.

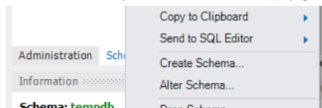
Par exemple :

```
SELECT firstname, lastname FROM user;
```

Filtrer les données récupérées

Si vous voulez récupérer des rangées respectant certaines conditions, il faut utiliser la commande `WHERE`.

Vous pouvez utiliser tous les opérateurs de comparaison suivants : `=` (égal), `!=` (différent), `>` (strictement



Pour vérifier la base de données sur laquelle vous êtes :

```
select database();
```

Vous aurez par exemple `test`.

Une fois une base de données sélectionnée, vous n'avez plus besoin de la spécifier dans toutes les commandes `SQL` que vous allez faire.

Lire des données

Pour afficher toutes les données d'une table il suffit de faire :

```
SELECT * FROM table;
```

Vous aurez par exemple `test`.

Une fois une base de données sélectionnée, vous n'avez plus besoin de la spécifier dans toutes les commandes `SQL` que vous allez faire.

Lire des données

Pour afficher toutes les données d'une table il suffit de faire :

```
SELECT * FROM table;
```

`*` signifie toutes les colonnes.

Pour n'afficher les données que de certaines colonnes, vous pouvez les spécifier en les séparant par des virgules.

Par exemple :

```
SELECT firstname, lastname FROM user;
```

Filtrer les données récupérées

```
SELECT * FROM user WHERE id_user = 2;
```

Vous pouvez combiner les conditions avec les opérateurs logiques : `AND` (ET) et `OR` (OU).

Par exemple :

```
SELECT * FROM user WHERE id_user = 2 AND password = 456;
```

Vous pouvez utiliser l'opérateur `BETWEEN` pour vérifier qu'une valeur est comprise entre deux valeurs incluses :

```
SELECT * FROM user WHERE id_user BETWEEN 1 AND 3;
```

Vous pouvez utiliser l'opérateur `BETWEEN` pour vérifier qu'une valeur est comprise entre deux valeurs incluses :

```
SELECT * FROM user WHERE id_user BETWEEN 1 AND 3;
```

Vous pouvez utiliser l'opérateur `IN` pour vérifier qu'une valeur est comprise parmi un ensemble de valeurs :

```
SELECT * FROM user WHERE firstname IN ('jean', 'pierre', 'paul', 'jacques');
```

Vous pouvez utiliser l'opérateur `NOT` pour inverser une condition :

```
SELECT * FROM user WHERE firstname NOT IN ('jean', 'pierre', 'paul', 'jacques');
```

Trier les données

Vous pouvez trier les données en utilisant `ORDER BY`.