

Correction du Devoir Maison - Base de données - Yu-Gi-Oh

1 La commande `COUNT` va permettre de réaliser ce calcul. Cette commande permet de compter le nombre d'enregistrements (c'est-à-dire de lignes de la table). La commande à utiliser est donc :

`SELECT COUNT(*) FROM cards`, le résultat obtenu est `5427`.

2 Il va falloir utiliser une condition pour obtenir ces informations à l'aide du mot clé `WHERE`

`SELECT Atk,Def FROM cards WHERE Name="Guardian Grarl"`, le résultat obtenu est une attaque de `2500` et une défense de `1000`.

3 `SELECT Name`

`FROM cards`

`WHERE (Atk BETWEEN 500 AND 550)`

`AND`

`(Def BETWEEN 150 AND 200)`

On trouve 3 monstres distincts : `Magical Plant Mandragola`, `Flip Flop Frog` et `Alien Ammonite`.

Sans l'utilisation de la commande `BETWEEN`, la commande est :

`SELECT Name`

`FROM cards`

`WHERE 500<=Atk AND Atk<=550`

`AND`

`150<=Def AND Def<=200`

4 On peut récupérer ces enregistrements en effectuant une sélection dont les attaques ne sont pas vides. `SELECT * FROM cards WHERE Atk<>""`

5 L'idée est de récupérer dans un premier temps l'ensemble de tous les enregistrements de la table `cards` et de retirer via la commande `EXCEPT` celles qui ont des valeurs d'attaque vide.

`SELECT * FROM cards EXCEPT SELECT * FROM cards WHERE Atk=""`

6 `SELECT COUNT(*) FROM (SELECT * FROM cards EXCEPT SELECT * FROM cards WHERE Atk="")`. On trouve alors effectivement `3192`.

7 Il suffit d'utiliser la commande `MAX(Atk)` appliquée à la sélection exempte des cartes « Piège » et « Magie ».

`SELECT MAX(Atk) FROM cards WHERE Atk<>""`

On trouve alors que le niveau d'attaque maximum est de `5000`.

8 `SELECT Name FROM cards WHERE Atk=(SELECT MAX(Atk) FROM cards WHERE Atk<> "")`. Les deux monstres sont `Dragon Master Knight` et `Five-Headed Dragon`.

9 `SELECT Name, Atk FROM cards WHERE Atk=(SELECT MAX(Atk) FROM cards WHERE Atk<>"" AND Level<=8) AND Level<=8`. On trouve alors un monstre :

`Beast Machine King Barbaros Ūr` qui a un niveau d'attaque de `3800`.

10 D'après la figure 3, le type d'une carte depuis la table `cards` est stockée dans la table `types`. On peut proposer :

`SELECT t.Name FROM cards as c JOIN types as t ON c.Type_ID=t.Type_ID WHERE c.Name="Vampire Genesis"`.

Le résultat retourné contient deux lignes : `Effect monster` et sa traduction française `Monstre à effet`.

11 On peut proposer :

```
SELECT me.Name FROM cards as c JOIN cards_x_monster_effects as cxme ON c.ID_Card=cxme.ID_Card JOIN
monster_effects as me ON me.Monster_Effect_ID=cxme.Monster_Effect_ID WHERE c.Name="Vampire Genesis".
```

Le résultat retourné contient deux lignes : Ignition effect et sa traduction française Effet de coût.

12 On propose :

```
SELECT AVG(c.AtK), mt.Name
FROM cards as c JOIN monster_types as mt
ON c.Monster_Type_ID=mt.Monster_Type_ID
WHERE mt.Language_ID=2
GROUP BY mt.Monster_Type_ID
```

13 À l'aide de la commande ORDER BY, on propose :

```
SELECT AVG(c.AtK) as moy, mt.Name
FROM cards as c JOIN monster_types as mt
ON c.Monster_Type_ID=mt.Monster_Type_ID
WHERE mt.Language_ID=2
GROUP BY mt.Monster_Type_ID
ORDER BY moy
```

14 La commande HAVING va nous permettre de réaliser la requête demandée :

```
SELECT AVG(c.AtK) as moy, mt.Name
FROM cards as c JOIN monster_types as mt
ON c.Monster_Type_ID=mt.Monster_Type_ID
WHERE mt.Language_ID=2
GROUP BY mt.Monster_Type_ID
HAVING COUNT(*)>=300
ORDER BY moy
```

15 On peut sélectionner les noms uniques avec un GROUP BY (comme ici) ou avec un DISTINCT c1.Name) SELECT

```
c1. Name FROM cards AS c1
JOIN cards AS c2
ON c1.Card_ID<>c2.Card_ID AND c1.id_Card = c2.id_Card
GROUP BY c1.Name
```

16 Il faut faire attention à grouper par c1.Card_ID pour avoir des groupes correspondant à une même carte. Le filtrage s'effectue sur COUNT(*)+1 car il y a dans chaque groupe un élément pour chaque c2.id_Card différent de c1.id_Card mais il manque justement c1.id_Card que l'on rajoute.

```
SELECT DISTINCT c1.Name
FROM cards AS c1
JOIN cards AS c2
ON c1.Card_ID<>c2.Card_ID AND c1.id_Card = c2.id_Card
GROUP BY c1.Card_ID
HAVING COUNT(*)+1>=10
```

17 On peut proposer :

```
SELECT Atk FROM
cards WHERE Atk<>""
ORDER BY Atk DESC
```

18 On peut proposer :

```
SELECT DISTINCT(Atk) FROM
cards WHERE Atk<>""
ORDER BY Atk DESC
```

19 On utilise la commande LIMIT

```
SELECT DISTINCT(Atk) FROM  
cards WHERE Atk<>"")  
ORDER BY Atk DESC  
LIMIT 10
```

20 On utilise la commande LIMIT associée avec OFFSET

```
SELECT DISTINCT(Atk) FROM  
(SELECT * FROM cards EXCEPT SELECT * FROM cards WHERE Atk="")  
ORDER BY Atk DESC  
LIMIT 5  
OFFSET 10
```