

Requêtes SQL :

Suite à la création de la base formula1, voici les requêtes SQL fonctionnelles avec les contraintes de commande demandés dans l'énoncé.

1. classement_drivers → Vue qui classe les pilotes par points

The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The query editor contains the following SQL code:

```
1 • Use formula1;
2
3 • CREATE VIEW classement_drivers AS
4     SELECT u.firstname, u.lastname, SUM(b.point) AS total_points
5     FROM users u
6     JOIN drivers d ON u.id_user = d.id_user
7     JOIN results r ON d.id_driver = r.id_driver
8     JOIN bareme b ON r.id_bareme = b.id_bareme
9     GROUP BY u.firstname, u.lastname
10    ORDER BY total_points DESC;
11
12
```

The results grid below the query editor displays the structure of the view:

	Field	Type	Null	Key	Default	Extra
▶	firstname	varchar(50)	NO		NULL	
▶	lastname	varchar(50)	NO		NULL	
	total_points	decimal(32,0)	YES		NULL	

Avec le Having :

```
12 • CREATE VIEW classement_drivers AS
13     SELECT u.firstname, u.lastname, SUM(b.point) AS total_points
14     FROM users u
15     JOIN drivers d ON u.id_user = d.id_user
16     JOIN results r ON d.id_driver = r.id_driver
17     JOIN bareme b ON r.id_bareme = b.id_bareme
18     GROUP BY u.firstname, u.lastname
19     HAVING SUM(b.point) > 50
20     ORDER BY total_points DESC;
21
```

2. Bloc suivant → Sélection d'utilisateurs selon critères.

```

16 •   SELECT DISTINCT u.firstname, u.lastname, u.email
17     FROM users u
18     WHERE u.firstname LIKE 'M%'
19     AND u.id_user IN (1, 2, 3, 4);
20

```

The screenshot shows the MySQL Workbench interface with a query editor containing the code above and a results grid below it. The results grid displays one row of data:

	firstname	lastname	email
▶	Max	Verstappen	max.verstappen@redbull.com

3. Bloc suivant → Mise à jour des points des équipes selon leur score. (Ajoute de points bonus suivant score)

```

22 •   UPDATE teams
23   SET points = CASE
24     WHEN points > 300 THEN points + 10
25     WHEN points BETWEEN 100 AND 300 THEN points + 5
26     ELSE points
27   END;

```

The screenshot shows the MySQL Workbench interface with a query editor containing the code above and a results grid below it. The results grid displays the structure of the 'teams' table:

	Field	Type	Null	Key	Default	Extra
▶	id_team	int	NO	PRI	NULL	auto_increment
	libelle	varchar(50)	NO		NULL	
	date_creation	date	YES		NULL	
	points	int	YES		0	

4. Bloc suivant → Suppression des résultats d'un pilote spécifique.

The screenshot shows the MySQL Workbench interface with a query editor containing the code above and an output log below it. The log shows the execution of a delete query:

```

31 •   DELETE FROM results
32   WHERE id_driver = 1;
33

```

Action Output

#	Time	Action	Message	Duration / Fets
1	5 13:54:17	DELETE FROM results WHERE id_driver = 1	0 row(s) affected	0.000 sec

5. Bloc suivant → Combine équipes et pilotes dans une seule liste.

```

33 •   SELECT libelle AS nom, 'Equipe' AS type FROM teams
34   UNION
35   SELECT firstname AS nom, 'Pilote' AS type FROM users
36   ORDER BY nom;
37

```

The screenshot shows the MySQL Workbench interface with a query editor containing the code above and a results grid below it. The results grid displays a combined list of team names and pilot names, categorized by type:

	nom	type
▶	Alexander	Pilote
	Alpine	Equipe
	Andrea Kimi	Pilote
	Aston Martin	Equipe
	Carlos	Pilote
	Charles	Pilote
	Esteban	Pilote
	Fernando	Pilote
	Ferrari	Equipe
	Franco	Pilote
	Gabriel	Pilote
	George	Pilote
	Haas F1 Team	Equipe
	Isack	Pilote
	Kick Sauber	Equipe
	Lance	Pilote
	Lando	Pilote
	Lewis	Pilote

Et avec la limite à 10 :

```
32
33 •  SELECT libelle AS nom, 'Equipe' AS type FROM teams
34 UNION
35 SELECT firstname AS nom, 'Pilote' AS type FROM users
36 ORDER BY nom
37 LIMIT 10;
~
```

Result Grid | Filter Rows: Export: Wrap Cell Content

nom	type
Alexander	Pilote
Alpine	Equipe
Andrea Kimi	Pilote
Aston Martin	Equipe
Carlos	Pilote
Charles	Pilote
Esteban	Pilote
Fernando	Pilote
Ferrari	Equipe
Franco	Pilote

6. Bloc suivant → Ajout d'une colonne « statut » dans « drivers ».

```
42
43 • ALTER TABLE drivers
44 ADD COLUMN statut VARCHAR(20);
45
46 • Explain drivers;
```

Result Grid | Filter Rows: Export: Wrap Cell Content

Field	Type	Null	Key	Default	Extra
id_driver	int	NO	PRI	NULL	auto_increment
points	int	YES		0	
id_user	int	NO	MUL	NULL	
statut	varchar(20)	YES		NULL	

Output x

Action Output

#	Time	Action	Message
10	13:56:41	SELECT libelle AS nom, 'Equipe' AS type FROM teams UNION SELECT firstname AS nom, 'Pilote' AS type FROM users	30 row(s) returned
11	13:59:15	ALTER TABLE drivers ADD COLUMN statut VARCHAR(20)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
12	14:00:44	Explain drivers	4 row(s) returned

7. Bloc suivant → Suppression d'une table temporaire (si elle existe, ici non).

```
50 • DROP TABLE IF EXISTS temp_stats;
51
52
~
```

Output

Action Output

#	Time	Action
11	13:59:15	ALTER TABLE drivers ADD COLUMN statut VARCHAR(20)
12	14:00:44	Explain drivers
13	14:01:33	DROP TABLE IF EXISTS temp_stats

8. Bloc suivant → Vidage de la table « barème ». Ici j'ai rencontré l'erreur 1701 et il m'a fallu donc aller chercher comment le résoudre →
<https://blog.juansorroche.com/mysql-desactiver-les-constraintes-referentiellestemporairement>

```

46 • SET FOREIGN_KEY_CHECKS=0;
47 • TRUNCATE TABLE bareme;
48 • SET FOREIGN_KEY_CHECKS=1;

Output:
# Time Action
# 13 14:01:33 DROP TABLE IF EXISTS temp_stats
# 14 14:03:11 TRUNCATE TABLE bareme
# 15 14:05:39 SET FOREIGN_KEY_CHECKS=1;

Message
0 row(s) affected, 1 warning(s); 1051 Unknown table formula1.temp_stats'
Error Code: 1701 Cannot truncate a table referenced in a foreign key constraint ('formula1`.`results', CONSTRAINT 'results_bfk_2')
0 row(s) affected

```

9. Bloc suivant → Analyse d'une requête sur les circuits.

49 • EXPLAIN SELECT * FROM circuits WHERE longueur > 5000;

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	circuits	NULL	ALL	NULL	NULL	NULL	NULL	24	33.33	Using where

10. Bloc suivant → Vue des meilleures équipes.

```

52 • CREATE OR REPLACE VIEW top_teams AS
53     SELECT t.libelle, SUM(b.point) AS total_points
54     FROM teams t
55     JOIN teams_users tu ON t.id_team = tu.id_team
56     JOIN users u ON tu.id_user = u.id_user
57     JOIN drivers d ON u.id_user = d.id_user
58     JOIN results r ON d.id_driver = r.id_driver
59     JOIN bareme b ON r.id_bareme = b.id_bareme
60     GROUP BY t.libelle
61     ORDER BY total_points DESC;
62
63 • Explain top_teams;

```

Field	Type	Null	Key	Default	Extra
libelle	varchar(50)	NO	NULL		
total_points	decimal(32,0)	YES	NULL		

11. Bloc suivant → Suppression d'une vue.

```
63 •  DROP VIEW IF EXISTS classement_drivers;
64
65 -- Explain classement_drivers;
```

Output

#	Time	Action	Message
18	14:09:54	Explain top_teams	2 row(s) returned
19	14:11:17	DROP VIEW IF EXISTS classement_drivers	0 row(s) affected
20	14:11:34	Explain classement_drivers	Error Code: 1146. Table 'formula1.classement_drivers' doesn't exist

J'ai rencontré l'erreur 1175 et l'erreur 1054 au tout début du script aussi, après recherche il fallait désactiver un paramètre dans Edit, voir →
<https://center.iobeya.com/fr/troubleshooting/while-runing-scripts-i-have-an-error-because-of-the-safe-update-mode/> et <https://sebhastian.com/mysql-error-1054-fix/>