



SportStats : Données de la Formule 1  
SAE 2.03 – Mettre en place une solution  
informatique pour l'entreprise



## Introduction

Dans le cadre de notre travail sur la SAE 2.03 qui se concentre sur l'exploration des solutions informatiques d'entreprise, nous avons été chargés de concevoir, peupler et exploiter une base de données SQL sur un sujet de notre choix. Inspirés par la dynamique et l'histoire riche de la Formule 1, nous avons choisi de créer une base de données exhaustive qui couvre l'évolution de ce sport de haute performance depuis sa naissance en 1950 jusqu'à nos jours.

Notre base de données offre un aperçu détaillé de l'univers de la Formule 1, englobant des informations sur les Grands Prix et les courses qui les composent, les pilotes qui ont marqué l'histoire, les circuits emblématiques, les châssis (véhicules) utilisés au fil des ans, ainsi que les écuries et leurs directeurs respectifs. Cet outil, nommé SportStats, permet alors d'exploiter tout le potentiel des données recueillies au fil des championnats disputés.

## Table des matières

Introduction.....	2
1. Description du projet .....	4
2. Diagramme de conception UML .....	5
3. Schéma relationnel.....	6
4. Questions en français et quelques de requêtes.....	7
5. Gestion de projet .....	23
6. Conclusion .....	24

# 1. Description du projet

L'outil SportStats permet d'exploiter des données portant sur la Formule 1 depuis sa création en 1950. Basé sur le système de gestion de base de données PostgreSQL installé sur le serveur OVH3 proposé par l'IUT, il doit permettre de lier les données des pilotes, des courses, des écuries ou encore des nationalités pour proposer des données finales pertinentes et exploitables pour analyser les tendances, les performances ou encore les statistiques des différentes entités de la Formule 1. Pour proposer un projet conséquent, nous avons créé et peuplé 9 tables liées entre elles.

Les données qui les composent sont les données officielles de la FIA (Fédération Internationale de l'Automobile) dont l'une des missions est d'encadrer les courses automobiles au niveau mondial. Ces données, consultables individuellement sur le site officiel de la FIA, ne sont pas téléchargeables à proprement parler : la plateforme propose un outil de recherche qui n'est pas adapté à notre usage. Après quelques recherches, nous avons donc choisi d'utiliser les données proposées par Aaron Frias, qui a compilé les données fournies par la FIA et les propose gratuitement au format CSV sur le site de [Kaggle](#). Nous avons donc dû modifier ces fichiers CSV pour qu'ils correspondent à notre base de données : après avoir modifié ces fichiers, nous avons pu directement les importer dans PostgreSQL.

Chaque pilote est enregistré avec son nom, prénom, son année de naissance et sa nationalité. Les pilotes peuvent participer à plusieurs championnats chaque année. Les championnats sont répertoriés par leur nom et leur date de création. Il est également nécessaire de répertorier les champions de chaque championnat annuel.

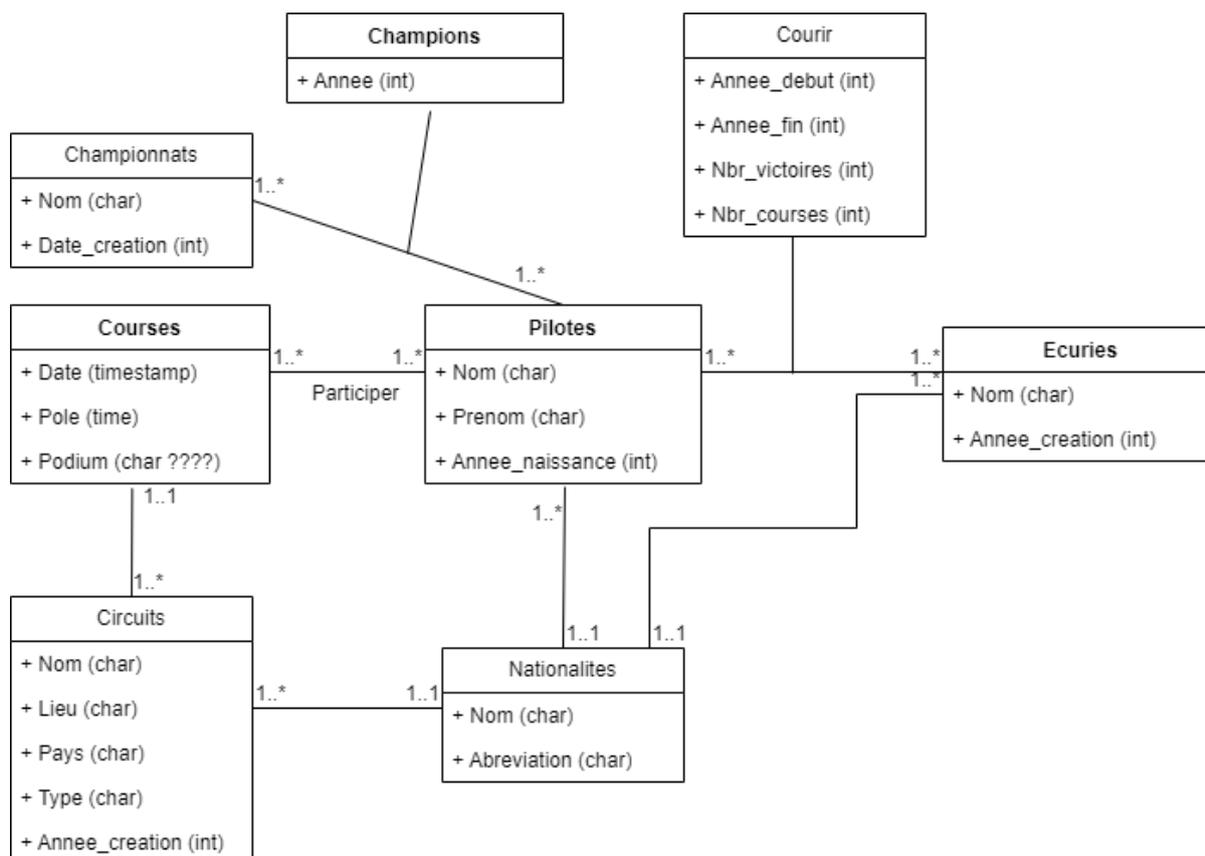
Un championnat est également défini par ses courses rattachées. Chaque championnat est constitué de plusieurs courses définies par leur nom, leur date, leur championnat rattaché, leur circuit rattaché (chaque course se dispute sur un seul circuit), le podium des trois premiers pilotes à passer la ligne d'arrivée, le poleman et enfin la pole. Chaque course est disputée une seule fois dans un championnat donné, mais une nouvelle édition du même championnat permet de disputer une course identique.

Les circuits sont définis par leur noms, leur lieu (ville), leur pays, leur type (asphalte dans le cas présent), ainsi que leur année de création.

Un pilote est obligatoirement lié à une écurie pour une saison (année) donnée. Cette écurie est enregistrée avec sa nationalité, son nom et son année de création. Cette écurie est gérée par un directeur, aussi appelé « Team Principal ». Celui-ci a comme attributs son nom, son prénom, sa nationalité, et l'année de prise et de fin de poste.

## 2. Diagramme de conception UML

Afin de pouvoir développer notre projet, il a d'abord fallu définir quelles tables pourraient être utiles pour proposer un outil complet, et choisir comment formater les données recueillies précédemment. Voici donc le diagramme de conception UML qui représente notre base de données :



### 3. Schéma relationnel

Nous avons également réalisé un diagramme relationnel à partir du diagramme de conception UML précédent. Les clés primaires sont soulignées, et les clés étrangères sont en italique et précédées d'un dièse :

Nationalites (id, nom, abreviation)

Championnats (id, nom, date\_creation)

Pilotes (id, nom, prenom, année\_naissance, *#nationalite*)

Champions (*#id\_championnat*, annee, *#id\_pilote*)

Circuits (id, nom, lieu, *#pays*, type, annee\_creation)

Courses (id, date, gagnant, poleman, pole, podium, *#id\_championnat*, *#id\_circuit*)

Participer (id, *#id\_pilote*, *#id\_course*)

Ecuries (id, nom, annee\_creation, *#nationalite*)

Chassis (id, nom, annee, *#ecurie*)

Team\_principal (id, nom, prenom, annee\_prise\_poste, annee\_fin\_poste, *#nationalite*)

Courir (id, annee\_debut, annee\_fin, nbr\_victoires, nbr\_courses, *#id\_pilote*, *#id\_ecurie*)

Diriger (id, *#id\_ecurie*, *#id\_tp*)

## 4. Questions en français et quelques de requêtes

Pour répondre pleinement à la problématique de ce projet et pour avoir une base données interactive, nous avons défini des questions, d'abord en français puis après nous les avons traduites en langage SQL.

### 1. Combien de temps un pilote a-t-il été dans une écurie ?

L'objectif de cette question est de savoir la durée totale qu'a passé un pilote dans les différentes écuries pour lesquelles il a couru. Pour cela, nous avons besoin de la table pilotes, courir ainsi qu'écurie.

```
SELECT pilotes.prenom, pilotes.nom, ecuries.nom as nom_ecurie,  
(courir.annee_fin - courir.annee_debut+1) AS duree  
FROM pilotes  
JOIN courir ON courir.id_pilote = pilotes.id  
JOIN ecuries ON ecuries.id = courir.id_ecurie WHERE  
pilotes.nom='Nom_Pilote' and pilotes.prenom= 'Prenom_Pilote'
```

On voit à travers ce code SQL que nous importons la table pilotes et que, par la suite, pour pouvoir obtenir les informations de l'écurie, nous devons passer par la table de jointure courir qui permet grâce aux id de la table pilotes et écuries de les relier en rajoutant des informations comme indiqué dans les différents diagrammes présentés ci-dessus. Puis nous faisons une jonction entre la table courir et écuries pour obtenir les informations souhaitées. Enfin, nous utilisons la fonction WHERE pour imposer nos conditions et afficher uniquement les données souhaitées. Cette requête permet, une fois mise en forme grâce a du code PHP et HTML, d'obtenir le résultat suivant si nous recherchons le pilote Lewis Hamilton :

### Combien de temps un pilote a-t-il été dans une écurie ?

Prénom :  Nom :

- Lewis Hamilton a été dans l'écurie McLaren pendant 6 années.
- Lewis Hamilton a été dans l'écurie Mercedes pendant 11 années.



3. Quelle est l'écurie la plus ancienne présente dans un championnat spécifié ainsi que les pilotes qui courrait pour elles durant ce championnat ?

```
SELECT
ecuries.id as id_ecurie,
ecuries.nom as nom_ecurie,
ecuries.annee_creation,
nat_ecurie.nom as nationalite_ecurie,
( SELECT STRING_AGG(pilotes.nom || ' ' || pilotes.prenom, ', ') as pilotes FROM pilotes
JOIN courir on pilotes.id = courir.id_pilote WHERE courir.id_ecurie = ecuries.id AND
courir.annee_debut <= 'Annee_Championnat' AND courir.annee_fin >= 'Annee_Championnat' GROUP BY
courir.id_ecurie) as pilotes
FROM ecuries
JOIN nationalites as nat_ecurie on ecuries.nationalite = nat_ecurie.id
JOIN courir on ecuries.id = courir.id_ecurie
JOIN courses on courses.id_championnat = 'Annee_Championnat'
WHERE courir.annee_debut <= 'Annee_Championnat' AND courir.annee_fin >= 'Annee_Championnat'
ORDER BY ecuries.annee_creation ASC LIMIT 1;
```

Cette requête SQL récupère les informations des deux écuries les plus anciennes qui ont participé à un championnat spécifique, en se basant sur l'année de création de l'écurie. Elle joint plusieurs tables, notamment écuries, nationalités, courir et courses pour obtenir des détails tels que l'ID de l'écurie, le nom, l'année de création et la nationalité de l'écurie. De plus, une sous-requête est utilisée pour obtenir la liste des pilotes associés à chaque écurie, en regroupant les résultats par ID de l'écurie et en concaténant les noms des pilotes en une seule chaîne. Les résultats sont ensuite triés par année de création de l'écurie en ordre croissant et limités aux deux premières lignes.

**3. Quelle est l'écurie la plus ancienne présente dans un championnat spécifié ainsi que les pilotes qui courrait pour elles durant ce championnat ?**

Annee Championnat:    
Écurie : Ferrari  
Année de création : 1929  
Nationalité : Italian  
Pilotes : Schumacher Michael, Irvine Eddie

```

ebusson=> SELECT
  ecuries.id as id_ecurie,
  ecuries.nom as nom_ecurie,
  ecuries.annee_creation,
  nat_ecurie.nom as nationalite_ecurie,
  (
    SELECT STRING_AGG(pilotes.nom || ' ' || pilotes.prenom, ', ') as pilotes
    FROM pilotes
    JOIN courir on pilotes.id = courir.id_pilote
    WHERE courir.id_ecurie = ecuries.id
    AND courir.annee_debut <= 1998 AND courir.annee_fin >= 1998
    GROUP BY courir.id_ecurie
  ) as pilotes
FROM ecuries
JOIN nationalites as nat_ecurie on ecuries.nationalite = nat_ecurie.id
JOIN courir on ecuries.id = courir.id_ecurie
JOIN courses on courses.id_championnat = 1998
WHERE courir.annee_debut <= 1998 AND courir.annee_fin >= 1998
ORDER BY ecuries.annee_creation ASC LIMIT 1;
 id_ecurie | nom_ecurie | annee_creation | nationalite_ecurie |          pilotes
-----+-----+-----+-----+-----
        6 | Ferrari   |          1929 | Italian            | Schumacher Michael, Irvine Eddie
(1 row)

```

4. Quelles sont les écuries dont les pilotes ont le plus gagné de courses dans chaque championnat ?

L'objectif de cette question est de connaître le nombre de victoire et le nom des écuries aillant gagné durant un championnat donnée au préalable.

```

SELECT ch.nom AS championnat_nom, e.nom AS ecurie_nom, COUNT(co.id) AS victoires

FROM courses co

JOIN championnats ch ON co.id_championnat = ch.id

JOIN pilotes p ON CAST(co.gagnant AS INTEGER) = p.id

JOIN courir c ON p.id = c.id_pilote

        AND TO_DATE(co.date, 'YYYY-MM-DD') >= TO_DATE(c.annee_debut::text
|| '-01-01', 'YYYY-MM-DD')

        AND TO_DATE(co.date, 'YYYY-MM-DD') <= TO_DATE(c.annee_fin::text || '-
12-31', 'YYYY-MM-DD')

JOIN ecuries e ON c.id_ecurie = e.id

WHERE ch.date_creation = 'Annee_Championnat'

GROUP BY ch.nom, e.nom

ORDER BY victoires DESC ;

```

Cette requête SQL récupère le nombre de victoires par écurie pour un championnat spécifique. Elle joint plusieurs tables, filtre les résultats pour un championnat donné, et compte le nombre de victoires pour chaque écurie. Les résultats sont ensuite triés par le nombre de victoires en ordre décroissant. Chaque ligne du résultat contient le nom du championnat, le nom de l'écurie et le nombre de victoires de cette écurie dans ce championnat. Voici le résultat sur le site et dans la console :

#### Quelles sont les écuries dont les pilotes ont le plus gagné de courses dans chaque championnat ?

Annee Championnat

Championnat: F1\_1997  
Écurie: Williams  
Victoires: 8

Championnat: F1\_1997  
Écurie: Ferrari  
Victoires: 5

Championnat: F1\_1997  
Écurie: McLaren  
Victoires: 3

Championnat: F1\_1997  
Écurie: Benetton  
Victoires: 1

Championnat: F1\_1997  
Écurie: Sauber  
Victoires: 1

```

ebusson=> SELECT ch.nom AS championnat_nom, e.nom AS ecurie_nom, COUNT(co.id) AS victoires
FROM courses co
JOIN championnats ch ON co.id_championnat = ch.id
JOIN pilotes p ON CAST(co.gagnant AS INTEGER) = p.id
JOIN courir c ON p.id = c.id_pilote
AND TO_DATE(co.date, 'YYYY-MM-DD') >= TO_DATE(c.annee_debut::text || '-01-01', 'YYYY-MM-DD')
AND TO_DATE(co.date, 'YYYY-MM-DD') <= TO_DATE(c.annee_fin::text || '-12-31', 'YYYY-MM-DD')
JOIN ecuries e ON c.id_ecurie = e.id
WHERE ch.date_creation = 1997
GROUP BY ch.nom, e.nom
ORDER BY victoires DESC ;
championnat_nom | ecurie_nom | victoires
-----+-----+-----
F1_1997          | Williams   |          8
F1_1997          | Ferrari    |          5
F1_1997          | McLaren    |          3
F1_1997          | Benetton   |          1
F1_1997          | Sauber     |          1
(5 rows)

```

5. Combien de championnats a gagné un pilote ?

```

SELECT COUNT(*) AS count FROM champions, pilotes
WHERE pilotes.nom = 'Nom_pilote' AND pilotes.prenom = 'Prenom_Pilote' AND
pilotes.id = champions.id_pilote;

```

## Nombre de championnats gagnés par un pilote

Prénom :  Nom :

Le pilote Lewis Hamilton a gagné 7 championnats.

```

ebusson=> SELECT COUNT(*) AS count FROM champions, pilotes
WHERE pilotes.nom = 'Hamilton' AND pilotes.prenom = 'Lewis' AND pilotes.id = champions.id_pilote;
count
-----
7
(1 row)

```

6. Combien de pilotes courent sous une nationalité donnée ?

```

SELECT COUNT(*) AS count FROM pilotes JOIN nationalites ON
nationalites.id=pilotes.nationalite WHERE nationalites.nom='Nom_Nationalite';

```

## Combien de pilotes courent sous une nationalité donnée ?

ID de la nationalité :

Il y a 165 pilotes sous la nationalité avec l'ID British.

```
ebusson=> SELECT COUNT(*) AS count FROM pilotes JOIN nationalites ON nationalites.id=pilotes.nationalite WHERE nationalites.nom='British';
count
-----
165
(1 row)
```

## 7. Quel est le champion du monde ?

Cette question permet d'obtenir le nom, prénom du pilote ainsi que sa nationalité, mais aussi le nom de l'écurie ainsi que sa nationalité.

```
SELECT pilotes.nom AS nom_pilote, pilotes.prenom AS prenom_pilote, nat_pilote.nom AS nationalite_pilote, ecuries.nom, nat_ecurie.nom AS nationalite_ecurie
FROM pilotes JOIN nationalites AS nat_pilote ON pilotes.nationalite = nat_pilote.id
JOIN courir ON pilotes.id = courir.id_pilote
JOIN ecuries ON ecuries.id = courir.id_ecurie
JOIN nationalites AS nat_ecurie ON ecuries.nationalite = nat_ecurie.id JOIN champions
ON champions.id_pilote = pilotes.id
JOIN championnats ON championnats.id = champions.id_championnat WHERE
championnats.id =1998 AND championnats.id BETWEEN courir.annee_debut AND
courir.annee_fin ;
```

Cette requête SQL récupère les informations sur les pilotes et les écuries qui ont participé à un championnat spécifique. Elle renvoie le nom et le prénom du pilote, la nationalité du pilote, le nom de l'écurie et la nationalité de l'écurie. La requête joint plusieurs tables, notamment pilotes, nationalités, courir, écuries, champions et championnats en utilisant les clés appropriées. Elle filtre ensuite les résultats pour inclure uniquement les pilotes et les écuries qui ont participé au championnat spécifié pendant les années où le pilote était actif dans l'écurie. Voici les résultats sur le site et dans la console :

## Details d'un champion du monde

Nom du pilote : Häkkinen

Prénom du pilote : Mika

Nationalité du pilote : Finnish

Nom de l'écurie : McLaren

Nationalité de l'écurie : British

```

ebusson=> SELECT pilotes.nom AS nom_pilote, pilotes.prenom AS prenom_pilote, nat_pilote.nom AS nationalite_pilote, ecuries.nom, nat_ecurie.nom
AS nationalite_ecurie
FROM pilotes JOIN nationalites AS nat_pilote ON pilotes.nationalite = nat_pilote.id
JOIN courir ON pilotes.id = courir.id_pilote
JOIN ecuries ON ecuries.id = courir.id_ecurie
JOIN nationalites AS nat_ecurie ON ecuries.nationalite = nat_ecurie.id JOIN champions ON champions.id_pilote = pilotes.id
JOIN championnats ON championnats.id = champions.id_championnat WHERE championnats.id =1998 AND championnats.id BETWEEN courir.annee_debut AND
courir.annee_fin ;
nom_pilote | prenom_pilote | nationalite_pilote | nom | nationalite_ecurie
-----+-----+-----+-----+-----
Häkkinen | Mika | Finnish | McLaren | British
(1 row)

```

8. Combien d'écuries a fait un pilote ?

```

SELECT pilotes.prenom, pilotes.nom, COUNT(DISTINCT ecuries.id) AS nombre_ecurie,
SUM(courir.annee_fin - courir.annee_debut + 1) AS duree_totale FROM pilotes

```

```

JOIN courir ON courir.id_pilote = pilotes.id

```

```

JOIN ecuries ON ecuries.id = courir.id_ecurie WHERE pilotes.nom = 'Nom_Pilote' AND
pilotes.prenom='Prenom_Pilote' GROUP BY pilotes.prenom, pilotes.nom;

```

## Combien d'écuries a fait un pilote ?

Nom du pilote :  Prenom du pilote :

Le pilote Charles Leclerc a été dans 2 écuries pendant 6 années.

```

ebusson=> SELECT pilotes.prenom, pilotes.nom, COUNT(DISTINCT ecuries.id) AS nombre_ecurie, SUM(courir.annee_fin - courir.annee_debut + 1) AS d
uree_totale FROM pilotes
JOIN courir ON courir.id_pilote = pilotes.id
JOIN ecuries ON ecuries.id = courir.id_ecurie WHERE pilotes.nom = 'Leclerc' AND pilotes.prenom='Charles' GROUP BY pilotes.prenom, pilotes.nom;
prenom | nom | nombre_ecurie | duree_totale
-----+-----+-----+-----
Charles | Leclerc | 2 | 6
(1 row)

```

9. Quel est le podium avec le plus de titres de champions du monde ?

```
SELECT c.nom AS Nom_Course, p1g.nom AS Premier, p2d.nom AS Deuxieme, p3t.nom
AS Troisieme, (p1.titres + p2.titres + p3.titres) AS Total_Titres FROM courses c
JOIN Pilotes p1g ON CAST(c.gagnant AS INTEGER) = p1g.id
JOIN Pilotes p2d ON CAST(c.deuxieme AS INTEGER) = p2d.id JOIN Pilotes p3t ON
CAST(c.troisieme AS INTEGER) = p3t.id
JOIN (
SELECT id_pilote, COUNT(*) AS titres FROM Champions GROUP BY id_pilote)
p1 ON CAST(c.gagnant AS INTEGER) = p1.id_pilote
JOIN (
SELECT id_pilote, COUNT(*) AS titres FROM Champions GROUP BY id_pilote)
p2 ON CAST(c.deuxieme AS INTEGER) = p2.id_pilote
JOIN (
SELECT id_pilote, COUNT(*) AS titres FROM Champions GROUP BY id_pilote)
p3 ON CAST(c.troisieme AS INTEGER) = p3.id_pilote ORDER BY Total_Titres DESC LIMIT
1;
```

Cette requête SQL récupère les informations sur les courses, y compris le nom de la course et les noms des pilotes qui ont terminé en première, deuxième et troisième position. Elle utilise plusieurs jointures pour associer les pilotes aux positions qu'ils ont obtenues dans chaque course. En outre, elle utilise des sous-requêtes pour compter le nombre total de titres remportés par chaque pilote (en tant que champion). Ces sous-requêtes sont jointes à la table principale courses sur les champs correspondants pour les gagnants, les deuxièmes et les troisièmes. Enfin, elle calcule le total des titres pour le podium de chaque course (la somme des titres des trois pilotes) et trie les résultats par ce total en ordre décroissant. Voici les résultats sur le site et dans la console :

## Quel est le podium avec le plus de titres de champions du monde ?

Rechercher

Podium avec le plus de titres :  
Course : Singapore Grand Prix  
Premier : Hamilton  
Deuxième : Verstappen  
Troisième : Vettel  
Total des titres : 14

```
ebusson=> SELECT c.nom AS Nom_Course, p1g.nom AS Premier, p2d.nom AS Deuxieme, p3t.nom AS Troisieme, (p1.titres + p2.titres + p3.titres) AS Total_Titres FROM courses c
JOIN Pilotes p1g ON CAST(c.gagnant AS INTEGER) = p1g.id
JOIN Pilotes p2d ON CAST(c.deuxieme AS INTEGER) = p2d.id JOIN Pilotes p3t ON CAST(c.troisieme AS INTEGER) = p3t.id
JOIN (
SELECT id_pilote, COUNT(*) AS titres FROM Champions GROUP BY id_pilote)
p1 ON CAST(c.gagnant AS INTEGER) = p1.id_pilote
JOIN (
SELECT id_pilote, COUNT(*) AS titres FROM Champions GROUP BY id_pilote)
p2 ON CAST(c.deuxieme AS INTEGER) = p2.id_pilote
JOIN (
SELECT id_pilote, COUNT(*) AS titres FROM Champions GROUP BY id_pilote)
p3 ON CAST(c.troisieme AS INTEGER) = p3.id_pilote ORDER BY Total_Titres DESC LIMIT 1;
nom_course | premier | deuxieme | troisieme | total_titres
-----|-----|-----|-----|-----
Spanish Grand Prix | Prost | Senna | Schumacher | 14
(1 row)
```

10. Sur combien de circuits différents un championnat s'est-il disputé ?

```
SELECT c.id_championnat, COUNT(DISTINCT c.id_circuit) AS nombre_circuits FROM
courses c WHERE c.id_championnat = 'Annee_Championnat' GROUP BY
c.id_championnat;
```

## Sur combien de circuits différents un championnat s'est-il disputé ?

Année du championnat :  Rechercher  
Le championnat de 2020 s'est disputé sur 14 circuits différents.

```
ebusson=> SELECT c.id_championnat, COUNT(DISTINCT c.id_circuit) AS nombre_circuits FROM courses c WHERE c.id_championnat =2020 GROUP BY c.id_c
hampionnat;
id_championnat | nombre_circuits
-----|-----
2020 | 14
(1 row)
```

11. Combien de pilotes différents une écurie a-t-elle eu ?

```
SELECT ecuries.nom, count(distinct courir.id_pilote) as nombre_pilotes from ecuries
join courir on courir.id_ecurie=ecuries.id where ecuries.nom= 'Nom_ecurie' group
by ecuries.nom;
```

## Combien de pilotes différent une écurie a-t-elle eu ?

Nom de l'ecurie :

- Ferrari : 97 pilotes

```
ebusson=> SELECT ecuries.nom, count(distinct courir.id_pilote) as nombre_pilotes from ecuries join courir on courir.id_ecurie=ecuries.id where
ecuries.nom= 'Ferrari' group by ecuries.nom;
  nom | nombre_pilotes
-----|-----
Ferrari |          97
(1 row)
```

12. Les années ou une nationalité a été la plus présente ?

```
SELECT ch.nom AS championnat, n.nom AS nationalite, COUNT(DISTINCT p.id) AS
nombre_pilotes
```

```
FROM pilotes p
```

```
JOIN nationalites n ON p.nationalite = n.id
```

```
JOIN participer ON participer.id_pilote = p.id
```

```
join courses on courses.id=participer.id_course
```

```
JOIN championnats ch ON ch.id = courses.id_championnat
```

```
WHERE n.nom = 'Nationalite'
```

```
GROUP BY ch.nom, n.nom ORDER BY COUNT(DISTINCT p.id) DESC LIMIT 10;
```

## Les années ou une nationalité a été la plus présente ?

Nationalite

- Championnat: F1\_1959, Nombre de pilotes: 25
- Championnat: F1\_1960, Nombre de pilotes: 22
- Championnat: F1\_1952, Nombre de pilotes: 21
- Championnat: F1\_1961, Nombre de pilotes: 21
- Championnat: F1\_1958, Nombre de pilotes: 20
- Championnat: F1\_1954, Nombre de pilotes: 19
- Championnat: F1\_1953, Nombre de pilotes: 18
- Championnat: F1\_1957, Nombre de pilotes: 17
- Championnat: F1\_1956, Nombre de pilotes: 15
- Championnat: F1\_1962, Nombre de pilotes: 15

```

ebusson=> SELECT ch.nom AS championnat, n.nom AS nationalite, COUNT(DISTINCT p.id) AS nombre_pilotes
FROM pilotes p
JOIN nationalites n ON p.nationalite = n.id
JOIN participer ON participer.id_pilote = p.id
join courses on courses.id=participer.id_course
JOIN championnats ch ON ch.id = courses.id_championnat
WHERE n.nom = 'British'
GROUP BY ch.nom, n.nom ORDER BY COUNT(DISTINCT p.id) DESC LIMIT 10;
championnat | nationalite | nombre_pilotes
-----
F1_1959      | British     | 25
F1_1960      | British     | 22
F1_1952      | British     | 21
F1_1961      | British     | 21
F1_1958      | British     | 20
F1_1954      | British     | 19
F1_1953      | British     | 18
F1_1957      | British     | 17
F1_1956      | British     | 15
F1_1962      | British     | 15
(10 rows)

```

13. Le pilote qui a fait le plus de pole ?

```

SELECT pilotes.nom, COUNT(pilotes.nom) AS nombre_occurrences FROM pilotes, courses
WHERE CAST(courses.poleman AS INTEGER) = pilotes.id GROUP BY pilotes.nom ORDER BY
nombre_occurrences DESC';

```

### Quel pilote a fait le plus de pole ?

Rechercher

- Hamilton : 107 pole positions
- Vettel : 57 pole positions
- Schumacher : 41 pole positions
- Rosberg : 30 pole positions
- Verstappen : 28 pole positions
- Alonso : 23 pole positions
- Räikkönen : 19 pole positions
- Bottas : 19 pole positions
- Leclerc : 18 pole positions
- Massa : 16 pole positions
- Hill : 13 pole positions
- Webber : 12 pole positions
- Häkkinen : 10 pole positions

```

ebusson=> SELECT pilotes.nom, COUNT(pilotes.nom) AS nombre_occurrences FROM pilotes, courses WHERE CAST(courses.poleman AS INTEGER) = pilotes.
id GROUP BY pilotes.nom ORDER BY nombre_occurrences DESC LIMIT 10;
nom | nombre_occurrences
-----
Hamilton | 107
Vettel | 57
Schumacher | 41
Rosberg | 30
Verstappen | 28
Alonso | 23
Räikkönen | 19
Bottas | 19
Leclerc | 18
Massa | 16
(10 rows)

```

14. Quel est le nombre de pilotes gagnant pour un championnat ?

```
SELECT pilotes.nom, COUNT(courses.gagnant) AS nbr_victoires
FROM pilotes
JOIN courses ON CAST(pilotes.id AS VARCHAR) = courses.gagnant
JOIN championnats ON courses.id_championnat = championnats.id
WHERE championnats.id = 'Annee_Championnat'
GROUP BY pilotes.nom ;
```

### Quel est le nombre de pilotes gagnant pour un championnat ?

Championnat

Le pilote Rosberg a remporté 5 victoire(s) dans le championnat 2014.

Le pilote Ricciardo a remporté 3 victoire(s) dans le championnat 2014.

Le pilote Hamilton a remporté 11 victoire(s) dans le championnat 2014.

```
ebusson=> SELECT pilotes.nom, COUNT(courses.gagnant) AS nbr_victoires
FROM pilotes
JOIN courses ON CAST(pilotes.id AS VARCHAR) = courses.gagnant
JOIN championnats ON courses.id_championnat = championnats.id
WHERE championnats.id = 2014
GROUP BY pilotes.nom ;
nom      | nbr_victoires
-----+-----
Rosberg  |          5
Ricciardo |          3
Hamilton |         11
(3 rows)
```

15. Quel pilote a débuté en même temps qu'un autre ?

```
SELECT p1.nom AS pilote1, p2.nom AS pilote2

FROM (

    SELECT pilotes.id AS id, pilotes.nom AS nom, MIN(courir.annee_debut) AS
annee_debut

    FROM pilotes

    JOIN courir ON pilotes.id = courir.id_pilote

    WHERE pilotes.nom = 'Nom_Pilote'

    GROUP BY pilotes.id, pilotes.nom

) p1

JOIN (

    SELECT pilotes.id AS id, pilotes.nom AS nom, MIN(courir.annee_debut) AS
annee_debut

    FROM pilotes

    JOIN courir ON pilotes.id = courir.id_pilote

    GROUP BY pilotes.id, pilotes.nom

) p2 ON p1.annee_debut = p2.annee_debut AND p1.id < p2.id;
```

### Quel pilote a debute en meme temps qu'un autre ?

Nom du pilote :

Le pilote Hamilton et le pilote Vettel ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Miller ont commencé à courir la même année.  
Le pilote Hamilton et le pilote James ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Fotheringham-Parker ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Richardson ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Fischer ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Jover ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Kovalainen ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Gordini ont commencé à courir la même année.  
Le pilote Hamilton et le pilote James ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Abecassis ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Winkelhock ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Forberg ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Hirt ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Ball ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Nakajima ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Mackey ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Sutil ont commencé à courir la même année.  
Le pilote Hamilton et le pilote Grignard ont commencé à courir la même année.

```

ebusson=> SELECT p1.nom AS pilote1, p2.nom AS pilote2
FROM (
  SELECT pilotes.id AS id, pilotes.nom AS nom, MIN(courir.annee_debut) AS annee_debut
  FROM pilotes
  JOIN courir ON pilotes.id = courir.id_pilote
  WHERE pilotes.nom = 'Hamilton'
  GROUP BY pilotes.id, pilotes.nom
) p1
JOIN (
  SELECT pilotes.id AS id, pilotes.nom AS nom, MIN(courir.annee_debut) AS annee_debut
  FROM pilotes
  JOIN courir ON pilotes.id = courir.id_pilote
  GROUP BY pilotes.id, pilotes.nom
) p2 ON p1.annee_debut = p2.annee_debut AND p1.id < p2.id;
pilote1 |      pilote2
-----+-----
Hamilton | Vettel
Hamilton | Miller
Hamilton | James
Hamilton | Fotheringham-Parker
Hamilton | Richardson
Hamilton | Fischer
Hamilton | Jover
Hamilton | Kovalainen
Hamilton | Gordini
Hamilton | James
Hamilton | Abecassis
Hamilton | Winkelhock
Hamilton | Forberg
Hamilton | Hirt
Hamilton | Ball
Hamilton | Nakajima
Hamilton | Mackey
Hamilton | Sutil
Hamilton | Grignard
(19 rows)

```

16. Quel est La moyenne de courses effectuer par un pilote dans une écurie donnée ?

```

SELECT courir.id_ecurie, AVG(courir.nbr_courses) as Moyenne_courses

FROM courir

JOIN pilotes ON courir.id_pilote = pilotes.id

join ecuries on ecuries.id=courir.id_ecurie where ecuries.nom= 'Nom_Ecurie'

GROUP BY courir.id_ecurie;

```

### La moyenne de courses effecuter par un pilote dans une ecuries donnees

ID de l'écurie    
 L'écurie Ferrari a une moyenne de 42 course(s) par pilote.

```

ebusson=> SELECT courir.id_ecurie, AVG(courir.nbr_courses) as Moyenne_courses
FROM courir
JOIN pilotes ON courir.id_pilote = pilotes.id
join ecuries on ecuries.id=courir.id_ecurie where ecuries.nom= 'Ferrari'
GROUP BY courir.id_ecurie;
id_ecurie |  moyenne_courses
-----+-----
6 | 41.7113402061855670
(1 row)

```

17. Quels sont les noms des pilotes qui ont participé à toutes les courses d'un championnat spécifique ?

```
SELECT pilotes.nom
FROM pilotes
JOIN participer ON pilotes.id = participer.id_pilote
JOIN courses ON participer.id_course = courses.id
WHERE courses.id_championnat = 'Annee_Championnat'
GROUP BY pilotes.nom
HAVING COUNT(DISTINCT courses.id) = (SELECT COUNT(*) FROM courses WHERE
id_championnat = 'Annee_Championnat');
```

Quels sont les noms des pilotes qui ont participé à toutes les courses d'un championnat spécifique ?

Annee Du championnat

Berger  
Coulthard  
Diniz  
Frentzen  
Häkkinen  
Larini  
Nakano  
Panis  
Schumacher  
Trulli

```
ebusson=> SELECT pilotes.nom
ebusson->         FROM pilotes
ebusson->         JOIN participer ON pilotes.id = participer.id_pilote
ebusson->         JOIN courses ON participer.id_course = courses.id
ebusson->         WHERE courses.id_championnat = 1997
ebusson->         GROUP BY pilotes.nom
ebusson->         HAVING COUNT(DISTINCT courses.id) = (SELECT COUNT(*) FROM courses WHERE id_championnat = 1997);
nom
-----
Berger
Coulthard
Diniz
Frentzen
Häkkinen
Larini
Nakano
Panis
Schumacher
Trulli
(10 rows)
```

## 5. Gestion de projet

Tâche / Semaine	14	15	16	17	18	19	20	21	22
Cahier des charges									
Schéma conceptuel									
Schéma relationnel									
Recherche et récupération des datas									
Création des tables									
Peuplement des tables									
Tests unitaires									
Création des requêtes finales									
Rédaction du rapport									
Préparation de la soutenance									

Nous avons comme objectif de finir le projet à la fin de la semaine 21, et préparer une soutenance orale pour la semaine 22. Nous avons pu nous répartir les tâches équitablement en fonction des compétences de chacun. Les schémas conceptuel et relationnel ont été légèrement modifiés durant le projet pour correspondre aux données en notre possession, mais les liens entre les tables étaient corrects.

Nous avons rencontré quelques difficultés lors de la récupération des données, puisque ces dernières se font rares en accès libre sur Internet. Nous avons également dû les reformater afin de les faire correspondre à notre conception de tables, ce qui nous a fait prendre un léger retard sur le peuplement des tables. Nous avons cependant pu nous rattraper sur la phase de peuplement des tables, avec des fichiers au format CSV. Nous pouvons donc dire que notre emploi du temps prévisionnel a été globalement bien respecté.

## 6. Conclusion

En conclusion, le projet SportStats a permis de créer une base de données complète et détaillée sur l'univers de la Formule 1. Grâce à un travail rigoureux, nous avons réussi à compiler et organiser des données allant de 1950 à aujourd'hui.

Ce projet a non seulement renforcé notre compréhension des solutions informatiques d'entreprise, mais a également souligné l'importance de la gestion efficace des données dans le monde du sport. En effet, il est important de proposer une source de données sûre et fiable pour pouvoir créer des statistiques dans un monde en constante évolution.

Le projet SportStats est une expérience enrichissante qui offre une perspective sur les opportunités que l'on peut rencontrer dans le monde professionnel, en particulier dans le domaine de la gestion de données et de l'informatique.