

# **Rapport Laboratoire 3**

Sahitaj Edison & Nicolet Victor, Pilet Théo

Décembre 2024

## Table des matières

<i>Introduction .....</i>	<b>3</b>
<i>Partie 1 – Requêtes SQL.....</i>	<b>3</b>
<i>Partie 2 – Script SQL .....</i>	<b>10</b>
<i>Tests .....</i>	<b>10</b>
<i>Conclusion .....</i>	<b>11</b>

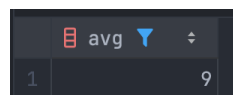
# Introduction

Ce rapport présente les solutions des différentes requêtes SQL demandées ainsi que l'implémentation d'une contrainte d'intégrité.

## Partie 1 – Requêtes SQL

- Les frais moyens d'inscription des événements organisés par la fédération 'BRITISH GYMNASTICS'.

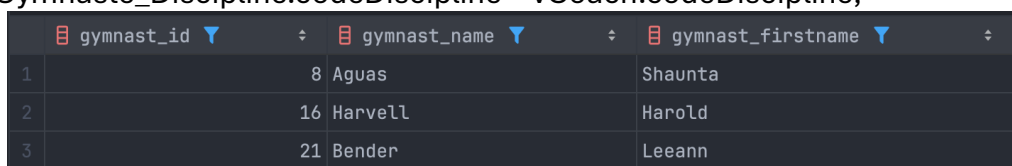
```
SELECT AVG(Evenement.fraisinscription)
FROM Evenement INNER JOIN Federation on Evenement.idfederation = Federation.id
WHERE Federation.nom = 'BRITISH GYMNASTICS';
```



	avg
1	9

- Les gymnastes qui pratiquent la discipline dont leur entraineur est le spécialiste et qui sont originaires du même pays.

```
SELECT DISTINCT
  vGymnaste.id AS gymnast_id,
  vGymnaste.nom AS gymnast_name,
  vGymnaste.prenom AS gymnast_firstname
FROM vGymnaste
  INNER JOIN Gymnaste_Discipline ON vGymnaste.id =
  Gymnaste_Discipline.idGymnaste
  INNER JOIN vCoach ON vGymnaste.idCoach = vCoach.id
WHERE vGymnaste.idPays = vCoach.idPays
AND Gymnaste_Discipline.codeDiscipline = vCoach.codeDiscipline;
```



	gymnast_id	gymnast_name	gymnast_firstname
1	8	Aguas	Shaunta
2	16	Harvell	Harold
3	21	Bender	Leeann

- Les coaches nés entre 1981 et 1996 (années comprises) qui entraînent au moins un gymnaste plus âgé qu'eux.

```
SELECT DISTINCT
  vCoach.id AS coach_id,
  vCoach.nom AS coach_name,
  vCoach.prenom AS coach_firstname
FROM vCoach
  INNER JOIN vGymnaste ON vGymnaste.idCoach = vCoach.id
WHERE EXTRACT(YEAR FROM vCoach.dateNaissance) BETWEEN 1981 AND 1996
AND vGymnaste.dateNaissance < vCoach.dateNaissance;
```

	coach_id	coach_name	coach_firstname
1	1	Hernandez	Luis
2	5	Traore	Aicha

Nous avons eu besoin de EXTRACT pour permettre d'extraire les années dans une date et ensuite de les comparer

- Les coaches qui entraînent plus de 2 gymnastes qui se sont inscrits à au moins 1 événement.

```
SELECT
  vCoach.id AS coach_id,
  vCoach.nom AS coach_name,
  vCoach.prenom AS coach_firstname,
  COUNT(DISTINCT Inscription.idGymnaste) AS gymnasts_count
FROM Inscription
  INNER JOIN vGymnaste ON Inscription.idGymnaste = vGymnaste.id
  INNER JOIN vCoach ON vGymnaste.idCoach = vCoach.id
GROUP BY vCoach.id, vCoach.nom, vCoach.prenom
HAVING COUNT(DISTINCT Inscription.idGymnaste) > 2;
```

	coach_id	coach_name	coach_firstname	gymnasts_count
1	1	Hernandez	Luis	3
2	5	Traore	Aicha	3

L'utilisation de COUNT avec HAVING est importante pour identifier les coaches qui entraînent plus de deux gymnastes.

- Pour les gymnastes ayant participé à au moins 2 événements, indiquer l'écart entre les frais d'inscription les moins élevés et les plus élevés payés. Classer les résultats par ordre alphabétique croissant du nom, puis du prénom des gymnastes.

```
SELECT
  vGymnaste.id AS gymnast_id,
  vGymnaste.nom AS gymnast_name,
  vGymnaste.prenom AS gymnast_firstname,
  MAX(Evenement.fraisInscription) - MIN(Evenement.fraisInscription) AS fee_difference
FROM Inscription
  INNER JOIN vGymnaste ON Inscription.idGymnaste = vGymnaste.id
  INNER JOIN Evenement ON Inscription.idEvenement = Evenement.id
GROUP BY vGymnaste.id, vGymnaste.nom, vGymnaste.prenom
HAVING COUNT(DISTINCT Inscription.idEvenement) >= 2
ORDER BY vGymnaste.nom ASC, vGymnaste.prenom ASC;
```

	gymnast_id	gymnast_name	gymnast_firstname	fee_difference
1	8	Aguas	Shaunta	72
2	21	Bender	Leeann	130
3	13	Burgdorf	Providencia	40
4	11	Deeann	Hibbert	0
5	12	Defazio	Annice	0
6	19	Halls	Krysta	20
7	20	Hinrichs	ShieLa	0
8	9	Lubinsky	Al	60
9	7	Plaisance	Isabella	10
10	14	Weinberger	Ozie	40

L'écart entre les frais d'inscription minimum et maximum pour chaque gymnaste est calculé avec MAX et MIN

- Indiquer (gymnaste + événement avec sa date d'ouverture + nombre de jours de retard) toutes les inscriptions faites après la date d'ouverture d'un événement.

```
SELECT
  vGymnaste.id AS gymnast_id,
  vGymnaste.nom AS gymnast_name,
  vGymnaste.prenom AS gymnast_firstname,
  Evenement.nom AS event_name,
  TO_CHAR(Evenement.dateOuverture, 'DD.MM.YYYY') AS event_opening_date,
  DATE_PART('day', Inscription.date - Evenement.dateOuverture) AS delay_days
FROM Inscription
  INNER JOIN vGymnaste ON Inscription.idGymnaste = vGymnaste.id
  INNER JOIN Evenement ON Inscription.idEvenement = Evenement.id
WHERE Inscription.date > Evenement.dateOuverture;
```

	gymnas...	gymnast_name	gymnast_firstname	event_name	event_...	delay_days
1	21	Bender	Leeann	FIG World Cup 2021	09.03.2021	1
2	13	Burgdorf	Providencia	TRA RGI World Cup 2022	12.03.2022	33

La différence en jours entre la date d'inscription et d'ouverture s'est fait via DATE\_PART et le format de la date (« DD.MM.YYYY ») via TO\_CHAR

- Par événement, lister tous les gymnastes inscrits en les classant par date d'inscription à l'événement (de la plus ancienne à la plus récente).

```
SELECT
  Evenement.nom AS event_name,
  TO_CHAR(Inscription.date, 'DD.MM.YYYY') AS registration_date,
  vGymnaste.id AS gymnast_id,
  vGymnaste.nom AS gymnast_name,
  vGymnaste.prenom AS gymnast_firstname
FROM Inscription
  INNER JOIN vGymnaste ON Inscription.idGymnaste = vGymnaste.id
  INNER JOIN Evenement ON Inscription.idEvenement = Evenement.id
ORDER BY Evenement.nom ASC, Inscription.date ASC;
```

	event_name	registration_date	gymnast_id	gymnast_name	gymnast_firstname
1	FIG World Cup 2021	10.03.2021	21	Bender	Leeann
2	FIG World Cup 2022	11.05.2022	8	Aguas	Shaunta
3	FIG World Cup 2022	11.05.2022	7	Plaisance	Isabella
4	FIG World Cup 2022	12.05.2022	9	Lubinsky	Al
5	FIG World Cup 2022	13.05.2022	10	Vaillancourt	Laurence
6	FIG World Cup 2022	14.05.2022	11	Deeann	Hibbert
7	FIG World Cup 2022	15.05.2022	12	Defazio	Annice
8	FIG World Cup 2022	16.05.2022	13	Burgdorf	Providencia
9	FIG World Cup 2022	17.05.2022	14	Weinberger	Ozie
10	FIG World Cup 2022	18.05.2022	15	Nygaard	Preston
11	FIG World Cup 2022	20.05.2022	17	Hutson	Dwight
12	FIG World Cup 2022	22.05.2022	19	Halls	Krysta
13	FIG World Cup 2022	23.05.2022	20	Hinrichs	Shiela
14	FIG World Cup 2022	24.05.2022	21	Bender	Leeann
15	FIG World Cup 2022	24.05.2022	18	LoveLace	Gena
16	FIG World Cup 2023 - AER	24.12.2022	8	Aguas	Shaunta
17	FIG World Cup 2023 - GFA - TRA	24.10.2021	9	Lubinsky	Al
18	FIG World Cup 2023 - GFA - TRA	24.01.2022	19	Halls	Krysta
19	FIG World Cup 2023 - GFA - TRA	22.03.2022	21	Bender	Leeann
20	TRA RGI World Cup 2022	24.10.2021	8	Aguas	Shaunta
21	TRA RGI World Cup 2022	01.12.2021	9	Lubinsky	Al
22	TRA RGI World Cup 2022	12.03.2022	14	Weinberger	Ozie
23	TRA RGI World Cup 2022	14.04.2022	13	Burgdorf	Providencia
24	VII Copa Interacional Catalunya	24.10.2023	7	Plaisance	Isabella
25	VIII Copa Interacional Catalunya	21.10.2023	7	Plaisance	Isabella
26	VIII Copa Interacional Catalunya	24.10.2023	11	Deeann	Hibbert
27	VIII Copa Interacional Catalunya	01.11.2023	12	Defazio	Annice
28	VIII Copa Interacional Catalunya	11.11.2023	20	Hinrichs	Shiela

- En partant du principe que les coaches payent les frais d'inscription aux événements de leurs gymnastes qui y participent en tant que juniors (moins de 16 ans au début de l'événement), classer les coaches par le montant total des frais d'inscriptions qu'ils ont payé (ordre décroissant) puis par ordre alphabétique de leur nom.

```

SELECT
  vCoach.id AS coach_id,
  vCoach.nom AS coach_name,
  vCoach.prenom AS coach_firstname,
  SUM(Evenement.fraisInscription) AS total_fees_paid
FROM vCoach
  INNER JOIN vGymnaste ON vCoach.id = vGymnaste.idCoach
  INNER JOIN Inscription ON vGymnaste.id = Inscription.idGymnaste
  INNER JOIN Evenement ON Inscription.idEvenement = Evenement.id
WHERE AGE(Evenement.dateOuverture, vGymnaste.dateNaissance) < INTERVAL '16
years'
GROUP BY vCoach.id, vCoach.nom, vCoach.prenom
ORDER BY total_fees_paid DESC, vCoach.nom ASC, vCoach.prenom ASC;

```

	coach_id	coach_name	coach_firstname	total_fees_paid
1	6	Hunt	Finley	150
2	1	Hernandez	Luis	80

- Par fédération qui a au moins 1 membre, indiquer le nombre de membres qui sont originaires de son pays et le nombre qui sont originaires d'un autre pays.

```

SELECT
    Federation.nom AS federation_name,
    COUNT(CASE WHEN vGymnaste.idPays = Federation.idPays THEN 1 END) AS
members_from_same_country,
    COUNT(CASE WHEN vGymnaste.idPays != Federation.idPays THEN 1 END) AS
members_from_other_countries
FROM Federation
    INNER JOIN vGymnaste ON Federation.id = vGymnaste.idFederation
GROUP BY Federation.nom
HAVING COUNT(vGymnaste.id) > 0;

```

	federation_name	members_from_same_country	members_from_other_countries
1	REAL FEDERACION ESPANOLA DE GIMNASIA	2	1
2	BRITISH GYMNASTICS	5	0
3	CANADIAN GYMNASTICS FEDERATION	1	2
4	FEDERATION SUISSE DE GYMNASTIQUE	3	1

Nous avons eu recours au conditions CASE pour compter de manière unique chaque membre originaire et non originaires du pays de la fédération

- Pour chaque gymnaste, indiquer le nombre d'événements auxquels il est inscrit et qui n'ont pas encore commencé.

```

SELECT
    vGymnaste.id AS gymnast_id,
    vGymnaste.nom AS gymnast_name,
    vGymnaste.prenom AS gymnast_firstname,
    COUNT(Evenement.id) AS upcoming_event_count
FROM Inscription
    INNER JOIN vGymnaste ON Inscription.idGymnaste = vGymnaste.id
    INNER JOIN Evenement ON Inscription.idEvenement = Evenement.id
WHERE Evenement.dateOuverture > CURRENT_DATE
GROUP BY vGymnaste.id, vGymnaste.nom, vGymnaste.prenom;

```

	gymnast_id	gymnast_name	gymnast_firstname	upcoming_event_count
1	7	Plaisance	Isabella	2
2	20	Hinrichs	Shiela	1
3	11	Deeann	Hibbert	1
4	12	Defazio	Annice	1

- Pour l'événement 'TRA RGI World Cup 2022', lister toutes les inscriptions (gymnastes + frais réels). Pour calculer les frais réels, on prend les frais d'inscription en tenant compte de 2 règles : On les double pour toute inscription hors délai (après l'ouverture de l'événement). Les gymnastes membres de la fédération qui organise l'événement ont un rabais de 20%.

```

SELECT
  vGymnaste.id AS gymnast_id,
  vGymnaste.nom AS gymnast_name,
  vGymnaste.prenom AS gymnast_firstname,
  TO_CHAR(Inscription.date, 'DD.MM.YYYY') AS registration_date,
  CASE
    WHEN Inscription.date > Evenement.dateOuverture THEN
      CASE
        WHEN vGymnaste.idFederation = Evenement.idFederation THEN 0.8 * 2 *
Evenement.fraisInscription
        ELSE 2 * Evenement.fraisInscription
      END
    ELSE
      CASE
        WHEN vGymnaste.idFederation = Evenement.idFederation THEN 0.8 *
Evenement.fraisInscription
        ELSE Evenement.fraisInscription
      END
    END AS actual_fee
FROM Inscription
  INNER JOIN vGymnaste ON Inscription.idGymnaste = vGymnaste.id
  INNER JOIN Evenement ON Inscription.idEvenement = Evenement.id
WHERE Evenement.nom = 'TRA RGI World Cup 2022';

```

	gymnast_id	gymnast_name	gymnast_firstname	registration_date	actual_fee
1	13	Burgdorf	Providencia	14.04.2022	128
2	14	Weinberger	Ozie	12.03.2022	64
3	8	Aguas	Shaunta	24.10.2021	80
4	9	Lubinsky	Al	01.12.2021	80

- Les fédérations (triées par leur nom) avec leurs événements ainsi que l'écart entre les frais d'inscription à l'événement et les frais d'inscription moyens aux événements de la fédération.

```

SELECT
  Federation.nom AS federation_name,
  Evenement.nom AS event_name,
  Evenement.fraisInscription AS event_fee,
  ROUND(AVG(Evenement.fraisInscription) OVER (PARTITION BY Federation.id), 2) AS
avg_event_fee,
  ROUND(Evenement.fraisInscription - AVG(Evenement.fraisInscription) OVER
(PARTITION BY Federation.id), 2) AS fee_difference
FROM Federation
  INNER JOIN Evenement ON Federation.id = Evenement.idFederation;

```

	federation_name	event_name	event_fee	avg_event_fee	fee_difference
1	FEDERATION SUISSE DE GYMNASTIQUE	TRA RGI World Cup 2022	80	78	2
2	FEDERATION SUISSE DE GYMNASTIQUE	FIG World Cup 2023 - GFA - TRA	20	78	-58
3	FEDERATION SUISSE DE GYMNASTIQUE	FIG World Cup 2022	40	78	-38
4	FEDERATION SUISSE DE GYMNASTIQUE	Match Junior SUI-FRA-GER 2022	100	78	22
5	FEDERATION SUISSE DE GYMNASTIQUE	FIG World Cup 2021	150	78	72
6	BRITISH GYMNASTICS	FIG World Cup 2023 - RGI	10	9	1
7	BRITISH GYMNASTICS	FIG World Cup 2023 - AER	8	9	-1
8	REAL FEDERACION ESPANOLA DE GIMNASIA	VIII Copa Interacional Catalunya	40	45	-5
9	REAL FEDERACION ESPANOLA DE GIMNASIA	VII Copa Interacional Catalunya	50	45	5



La fonction AVG OVER (PARTITION BY) nous a permis de calculer des moyennes dynamiques pour chaque fédération sans avoir besoin de créer des sous-requêtes

- Les gymnastes qui ne se sont inscrits qu'à des événements organisés par leur fédération

```
SELECT DISTINCT
  vGymnaste.id AS gymnast_id,
  vGymnaste.nom AS gymnast_name,
  vGymnaste.prenom AS gymnast_firstname
FROM Inscription
  INNER JOIN vGymnaste ON Inscription.idGymnaste = vGymnaste.id
  INNER JOIN Evenement ON Inscription.idEvenement = Evenement.id
WHERE NOT EXISTS (
  SELECT 1
  FROM Evenement AS Other_Event
  WHERE Other_Event.id = Inscription.idEvenement
  AND Other_Event.idFederation != vGymnaste.idFederation
);
```

	gymnast_id	gymnast_name	gymnast_firstname
1	7	Plaisance	Isabella
2	12	Defazio	Annice
3	13	Burgdorf	Providencia
4	14	Weinberger	Ozie
5	20	Hinrichs	Shiela
6	21	Bender	Leeann

## Partie 2 – Script SQL

```
CREATE OR REPLACE FUNCTION verify_coach_age()
  RETURNS TRIGGER AS $$
DECLARE
  coach_birth_date DATE;
BEGIN
  -- Check if the coach exists
  IF EXISTS (
    SELECT 1 FROM Coach WHERE idPersonne = NEW.idPersonne
  ) THEN
    -- Get the coach's birth date
    SELECT dateNaissance INTO coach_birth_date
    FROM Personne
    WHERE id = NEW.idPersonne;

    -- Check if the coach is at least 18 years old
    IF AGE(CURRENT_DATE, coach_birth_date) < INTERVAL '18 years' THEN
      RAISE EXCEPTION 'Coach must be at least 18 years old';
    END IF;
  END IF;

  RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER enforce_coach_age
  BEFORE INSERT OR UPDATE ON Coach
  FOR EACH ROW
  EXECUTE FUNCTION verify_coach_age();
```

### Tests

- Insertion d'un nouveau coach avec âge > 18

```
INSERT INTO Personne (id, idPays, nom, prenom, dateNaissance)
VALUES (22, 1, 'Doe', 'John', '1990-01-01');
```

```
INSERT INTO Coach (idPersonne, codeDiscipline)
VALUES (22, 'RGI');
```

- Insertion d'un nouveau coach avec âge < 18

```
INSERT INTO Personne (id, idPays, nom, prenom, dateNaissance)
VALUES (23, 1, 'Smith', 'Jane', '2010-05-15');
```

```
INSERT INTO Coach (idPersonne, codeDiscipline)
VALUES (23, 'RGI');
```

```
[2024-12-04 20:18:47] Connected
postgres> INSERT INTO Personne (id, idPays, nom, prenom, dateNaissance)
          VALUES (23, 1, 'Smith', 'Jane', '2010-05-15')
[2024-12-04 20:18:47] 1 row affected in 8 ms
postgres> INSERT INTO Coach (idPersonne, codeDiscipline)
          VALUES (23, 'RGI')
[2024-12-04 20:18:47] [P0001] ERROR: Coach must be at least 18 years old
[2024-12-04 20:18:47] 0ù : PL/pgSQL function verify_coach_age() line 12 at RAISE
```

## Conclusion

Ce labo nous a permis d'approfondir nos connaissances sur les requêtes SQL en implémentant des requêtes plus au moins intéressantes. De plus nous avons pu apprendre à implémenter une contrainte d'intégrité via un trigger.