

Projet Football Rapport

Calin Popa : 20158726

Joseph El-Sayegh : 20110482

Justin Bazinet : 20188987

Changmin Lee : 20169924

Passage du modèle E/A au modèle relationnel :

Explications :

1. Noter toutes les entités en tant que relations et écrire leurs attributs:

CoupeDeMonde(edition)

Equipe(pays)

Joueur(joueur_numero, prenom, nom, position, numero_chandail)

Staff(nom, prenom, staff_numero, fonction)

Match(lieu, date, rang, score_equipe1, score_equipe2)

Arbitre(arbitre_numero, prenom, nom)

2. Trouver les liens 0:1 et 1:1, dès qu'une association entre deux entités possèdent au moins l'un d'entre eux (pas besoin d'être les deux), on inscrit les attributs de l'association dans la relation de l'entité du côté du lien 0:1 ou 1:1 et met comme clé étrangère l'identifiant de l'autre entité (celle n'ayant pas "reçu" les attributs de l'association)

Equipe(pays, #edition, numéro_entraineur)

Match(lieu, date, rang, score_equipe1, score_equipe2, #edition, #pays1, #pays2, numéro_arbitre)

3. Pour les associations n'ayant pas de lien 0:1 ou 1:1, il faut les transformer en relation, avec leurs attributs ainsi que comme clés étrangères les clés primaires des entités qu'elle relie. Aussi, si l'association a un identificateur, ça va devenir la clé primaire. Sinon, ce seront les clés étrangères qui vont être la clé primaire.

EquipeJoueur(#pays, #edition, #joueur_numero)

Collaborateur(#pays, edition, #staff_numero)

ArbitreAssistant(#arbitre_numero, #pays1, #pays2, #edition, #date)

Participation(nb_passes, nb_buts, nb_tirs, nb_assistances, #joueur_numero, #date, #edition, #pays1, #pays2)

Sanction(couleur, heure, #joueur_numero, #date, #edition, #pays1, #pays2, #numero_arbitre)

4. Pour les attributs multivariés (donc quand un attribut peut être répété plusieurs fois), il faut les convertir en relation, mettre leurs attributs (si c'est le cas) et ajouter comme clé étrangère la clé primaire de la relation représentant l'entité qui avait cet attribut multivarié dans le modèle original. La clé primaire devient l'ensemble des attributs, si je comprends bien

Pas d'attributs multivariés, alors on ne fait rien à cette étape-ci.

Résultats finaux :

CoupeDeMonde(edition)

Joueur(joueur_numero, prenom, nom, position, numero_chandail)

Staff(nom, prenom, staff_numero, fonction)

Arbitre(arbitre_numero, prenom, nom)

Equipe(pays, #edition, numéro_entraineur)

Match(lieu, date, rang, score_equipe1, score_equipe2, #edition, #pays1, #pays2,
numéro_arbitre)

EquipeJoueur(#pays, #edition, #joueur_numero)

Collaborateur(#pays, edition, #staff_numero)

ArbitreAssistant(#arbitre_numero, #pays1, #pays2, #edition, #date)

Participation(nb_passes, nb_buts, nb_tirs, nb_assistances, #joueur_numero, #date, #edition, #pays1, #pays2)

Sanction(couleur, heure, #joueur_numero, #date, #edition, #pays1, #pays2, #numero_arbitre)

Normalisation :

CoupeDeMonde :

DF : aucune (1 seul attribut)

Résultat : **CoupeDeMonde(edition)** avec $\mathcal{F} = \emptyset$, ce qui est en BCNF, car aucune DF signifie aucune DF ne respectant pas les critères de cette forme normale.

Equipe :

DF : edition, pays \rightarrow entraineur_numéro (il faut connaître le pays de l'équipe ainsi que l'année en question pour savoir qui est l'entraîneur à ce moment-là).

Résultat : **Equipe(#edition, pays, #entraîneur_numéro)** avec $\mathcal{F} = \{ \text{edition, pays} \rightarrow \text{entraîneur_numéro} \}$. La relation est en BCNF, car la seule dépendance fonctionnelle $X \rightarrow A$ en est une avec X étant une super-clé (edition, pays).

Joueur :

DF : joueur_numéro \rightarrow prénom, nom, numero_chandail (joueur_numéro est justement un identifiant que nous avons choisi pour faciliter l'identification des joueurs, les joueurs étant représentés par leur prénom, leur nom ainsi que le numéro de leur chandail et leur position.)

Résultat :

Joueur(joueur_numéro, prénom, nom, position, numero_chandail) avec $\mathcal{F} = \{ \text{joueur_numéro} \rightarrow \text{prénom, nom, position, numero_chandail} \}$. La relation est en 3NF et BCNF (équivalents dans ce cas-ci).

La relation est donc déjà en BCNF.

Staff :

DF : $\text{staff_numéro} \rightarrow \text{prénom, nom, fonction}$ (chaque numéro de staff est associé à un seul employé dans l'ensemble de la base de données, donc c'est suffisant pour obtenir les autres informations. Le numéro permet de régler le cas de si deux membres de staff possèdent le même nom et le même prénom.)

Résultat : **Staff(staff_numéro, prénom, nom, fonction)** avec $\mathcal{F} = \{ \text{staff_numéro} \rightarrow \text{prénom, nom, fonction} \}$. Cette relation est en BCNF.

Match :

DF : $\text{date} \rightarrow \text{rang}$ (la date du match indique c'est rendu à quel rang, car les premiers jours sont pour la qualification et les derniers, pour la finale, par exemple).

$\text{edition, date, pays1} \rightarrow \text{pays1_score_final}$ (Pour déterminer c'est quoi le score de l'équipe 1, il faut savoir ce qu'est l'équipe 1 (ex : 2-1, mais on veut savoir le score de l'Argentine, est-ce que c'est la deuxième ou la première équipe? Cette dépendance fonctionnelle nous permet de répondre à cette question) ainsi que le match en question (pas besoin de connaître la deuxième équipe, car une équipe ne fait qu'un match max par journée, alors ça devrait être suffisant pour l'identifier).

$\text{edition, date, pays2} \rightarrow \text{pays2_score_final}$ (Même chose que dépendance fonctionnelle précédente).

$\text{edition, pays1, pays2, date} \rightarrow \text{lieu, arbitre_principal_numero}$ (En reconnaissant le match exactement, on peut savoir c'est où qu'il s'est déroulé et qui l'a arbitré. La raison qu'on a besoin de edition et de date est qu'on suppose que date n'est qu'un jour et un mois et non une année.)

Résultat : **Match(#edition, pays1, pays2, date, lieu, rang, pays1_score_final, pays2_score_final, #arbitre_principal_numero)**

La relation n'est pas 2NF (parce que certains attributs sont seulement déterminés par une partie de clé).

On sépare en

Niveau(date, rang) avec $\mathcal{F} = \{ \text{date} \rightarrow \text{rang} \}$. Cette relation est en BCNF.

Score1(#edition, date, #pays1, pays1_score_final) avec $\mathcal{F} = \{ \text{edition, date, pays1} \rightarrow \text{pays1_score_final} \}$. Cette relation est en BCNF.

Score2(#edition, date, #pays2, pays2_score_final) avec $\mathcal{F} = \{ \text{edition, date, pays2} \rightarrow \text{pays2_score_final} \}$. Cette relation est en BCNF.

InfosMatch(#edition, #pays1, #pays2, date, lieu, arbitre_principal_numero) avec $\mathcal{F} = \{ \text{edition, pays1, pays2, date} \rightarrow \text{lieu, arbitre_principal_numero} \}$. Cette relation est en BCNF.

Arbitre :

DF : arbitre_numéro \rightarrow prénom, nom (le numéro de l'arbitre sert d'identifiant unique, ce qui permet de trouver son prénom et son nom.)

Résultat : **Arbitre(arbitre_numéro, prénom, nom)** avec $\mathcal{F} = \{ \text{arbitre_numéro} \rightarrow \text{prénom, nom} \}$, ce qui est en BCNF.

ArbitreAssistant :

DF : Aucune DF (toutes les clés sont importantes afin d'identifier l'arbitre assistant, on ne peut pas juste dire qu'avoir la clé du match permet d'identifier l'arbitre assistant, car il y a 3 arbitres assistants, donc ça signifierait que l'attribut arbitre_numero ne serait pas unique à une certaine clé).

Résultat : **ArbitreAssistant(#arbitre_numero, #pays1, #pays2, #edition, #date)** avec $\mathcal{F} = \emptyset$, ce qui est en BCNF.

Sanction :

DF : edition, pays1, pays2, joueur_numero, arbitre_numero, date, heure \rightarrow couleur
(pour connaître une sanction donnée, il faut connaître le match (edition, pays1, pays2, date) ainsi que le joueur l'ayant reçu (joueur_numero) ainsi que l'arbitre l'ayant donnée (arbitre_numero) plus l'heure (au cas où que deux cartons jaunes soient donnés dans le même match (ne concerne pas les cartons rouges, car 1 max par match étant donné que ça cause l'expulsion du joueur))

Résultat : **Sanction(#edition, #pays1, #pays2, #joueur numéro, #arbitre numero, #date, heure, couleur)** avec $\mathcal{F} = \{ \text{edition, pays1, pays2, joueur_numero, arbitre_numero, date, heure} \rightarrow \text{couleur} \}$, ce qui est en BCNF.

Participation :

DF : edition, date, pays1, pays2, joueur_numéro \rightarrow nb_passes, nb_buts, nb_tirs, nb_assistances (edition, date, pays1 et pays2 permettent d'identifier le match et joueur_numéro permet d'identifier le joueur. Les deux combinés permettent de savoir la performance de quel joueur à quel match et d'ensuite aller chercher les informations pertinentes).

Résultat : **Participation(#edition, #date, #pays1, #pays2, #joueur numéro, nb_passes, nb_buts, nb_tirs, nb_assistances)** avec $\mathcal{F} = \{ \text{edition, date, pays1, pays2, joueur_numéro} \rightarrow \text{nb_passes, nb_buts, nb_tirs, nb_assistances} \}$, ce qui est en BCNF, car la seule dépendance en est une avec comme attributs du côté gauche la clé de la relation.

EquipeJoueur :

DF : joueur_numero \rightarrow pays (un joueur va toujours être dans l'équipe du même pays. C'est vraiment juste l'édition qui ne peut pas être déterminé par un autre attribut.)

Résultat : **EquipeJoueur(#edition, #joueur numero, #pays)** avec $\mathcal{F} = \{ \text{joueur_numero} \rightarrow$

pays}, ce qui n'est pas en 2NF, car pays ne dépend que d'une partie de clé. Il suffit de séparer en deux relations :

PaysJoueur(#joueur_numero, pays) avec $\mathcal{F} = \{ \text{joueur_numero} \rightarrow \text{pays} \}$, ce qui est en BCNF.

AnneesJoueur(#edition, #joueur_numero) avec $\mathcal{F} = \emptyset$, car il faut garder une relation avec la clé de la relation originale. Cette relation est en BCNF aussi.

Collaborateur :

DF : $\text{staff_numero} \rightarrow \text{pays}$ (un membre du staff va toujours servir l'équipe du même pays (à ma connaissance). C'est vraiment juste l'edition qui ne peut pas être déterminé par un autre attribut.)

Résultat : **Collaborateur(#edition, #joueur_numero, #pays)** avec $\mathcal{F} = \{ \text{staff_numero} \rightarrow \text{pays} \}$, ce qui n'est pas en 2NF, car pays ne dépend que d'une partie de clé. Il suffit de séparer en deux relations :

PaysStaff(#staff_numero, pays) avec $\mathcal{F} = \{ \text{staff_numero} \rightarrow \text{pays} \}$, ce qui est en BCNF.

AnneesStaff(#edition, #staff_numero) avec $\mathcal{F} = \emptyset$, car il faut garder une relation avec la clé de la relation originale. Cette relation est en BCNF aussi.

Schéma final de la base de données :

CoupeDeMonde(edition)

Equipe(#edition, pays, #entraîneur_numéro)

Joueur(joueur_numero, prénom, nom, position, numero_chandail)

Staff(staff_numero, prénom, nom, fonction)

Niveau(date, rang)

Score1(#edition, date, #pays1, pays1_score_final)

Score2(#edition, date, #pays2, pays2_score_final)

InfosMatch(#edition, #pays1, #pays2, date, lieu, #arbitre_principal_numero)

Arbitre(arbitre_numéro, prénom, nom)

ArbitreAssistant(#arbitre_numero, #pays1, #pays2, #edition, #date)

Sanction(#edition, #pays1, #pays2, #joueur numéro, #arbitre_numero, #date, heure, couleur)

Participation(#edition, #date, #pays1, #pays2, #joueur numéro, nb_passes, nb_buts, nb_tirs, nb_assistances)

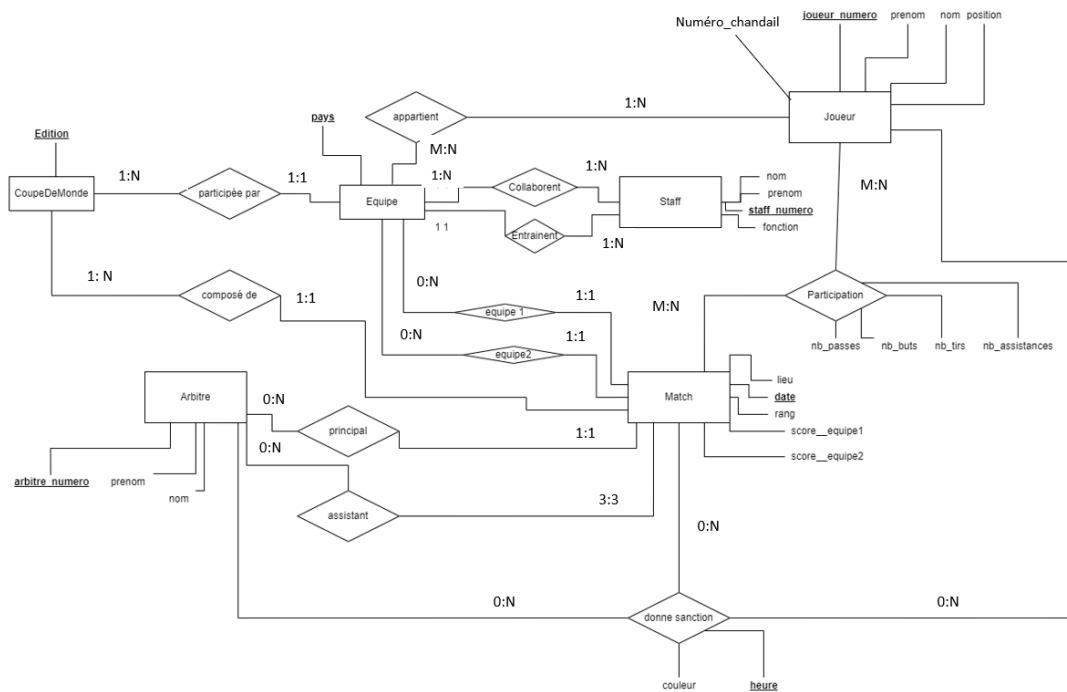
PaysJoueur(#joueur_numero, pays)

AnneesJoueur(#edition, #joueur_numero)

PaysStaff(#staff_numero, pays)

AnneesStaff(#edition, #staff_numero)

Model E/A



Un point important à mentionner est que ce schéma considère chaque équipe comme étant unique, donc l'équipe du Brésil lors de la première édition n'est pas la même que l'équipe du Brésil lors de la deuxième édition. C'est pour cette raison que chaque Equipe participe à 1 CoupeDeMonde.

Quant aux choix des entités, ce sont celles qui étaient mentionnées dans la description de la mise en situation. Il y a CoupeDeMonde, les Équipes, les Joueurs, le Staff technique, les Matches ainsi que les Arbitres. Pays n'a pas été inclus en tant qu'entité, car le mettre en attribut d'Équipe est suffisant. Aussi, Sanction a été relégué au rang d'association entre Match, Arbitre et Joueur.

Pour les associations, « participée par » indique les équipes participant à une certaine édition d'un tournoi, donc les pays ayant pu se rendre à la coupe, en quelque sorte. Une équipe ne participe qu'à une seule coupe, car on différencie les équipes selon les années aussi. « composé de » réfère au fait que plusieurs matchs se déroulent lors d'une coupe, mais que chaque match est associé à 1 seule coupe. Pour « appartient », un joueur appartient à plusieurs équipes, car il peut revenir pour différentes coupes du monde, et 1 équipe possède plusieurs joueurs (donc M :N). Pour « Collaborent », ce sont les différents membres du staff technique (excluant l'entraîneur) qui servent une équipe (chaque membre du staff est associé à 1 équipe ou plusieurs équipes, car ils

peuvent revenir d'une année à l'autre). « Entraîne » est la même chose, mais pour l'entraîneur de l'équipe. Vu que l'entraîneur peut revenir d'une année à l'autre, il y a aussi un lien 1 :N de Equipe à Staff pour cette association. Cependant, vu que chaque équipe n'a qu'un entraîneur, il y a un lien 1 :1 de ce côté. Match possède plusieurs associations. « Equipe 1 » et « Equipe 2 » sont séparés en deux associations au lieu d'une afin de pouvoir identifier quelle équipe a fait quel nombre de points (aussi la raison que score final a été séparé en deux). On ne voulait pas faire « Équipe Gagnante » et « Équipe perdante », car ça ne tiendrait pas compte des égalités. La raison qu'une équipe peut être 0 :N « equipe1 » ou « equipe2 » est qu'il est possible qu'une certaine équipe soit toujours assignée en tant que l'équipe 1, par exemple, donc ça signifierait qu'elle ne serait jamais l'équipe 2. Pour « Participation », un joueur joue plusieurs matchs et un match est joué par plusieurs joueurs, jamais un seul, sinon ça serait assez plate à regarder, disons. L'association « principal » est pour l'arbitre principal d'un match et il n'y en a qu'un seul, tandis que « assistant » est pour les arbitres assistants, qui sont au nombre de 3. La raison que c'est 0 :N est que ça se peut qu'un arbitre soit seulement un arbitre principal ou seulement un arbitre assistant s'il vient de commencer. Pour la dernière association, « donne sanction » correspond à un carton ayant été donné par un arbitre à un joueur lors d'un certain match. Autant un arbitre peut donner aucune ou plusieurs sanctions qu'un joueur peut en recevoir et qu'un certain nombre peut être donné lors d'un match, alors ce sont tous des liens 0 :N.

Pour les attributs des entités :

Il y a la CoupeDeMonde (qui correspond au tournoi en question). On peut l'identifier avec son édition, car il n'y a qu'une 3^e coupe du monde, par exemple.

Equipe a comme attribut important son pays.

Un Joueur est identifié par son numéro dans son équipe et il possède un prénom, un nom ainsi qu'une position dans son équipe. En tant que tel, vu que le numéro du joueur l'identifiant n'est pas spécifié, on va supposer que ce n'est pas son numéro de chandail et qu'il y a un autre numéro l'identifiant bel et bien de manière unique dans la base de données (car sinon, il y aurait des problèmes à chaque fois qu'un certain joueur réutilise le même numéro de chandail).

Un membre du Staff possède aussi un nom, un prénom, une fonction (collaborateur ou entraîneur) ainsi qu'un numéro de staff aidant à l'identifier.

Même chose pour Arbitre, mais sans la fonction, car on sait déjà que c'est un arbitre.

Pour les attributs de Match, il y a la date, le lieu, le rang (l'étape de la compétition, en quelque sorte), le score de l'équipe 1 et le score de l'équipe 2. On a décidé de mettre ces deux scores-là, car c'est associé au score final, qui est un attribut du match (au lieu de les mettre avec les associations equipe1 et equipe2).

Pour les attributs des entités :

Il n'y a que « Participation » et « donne sanction » qui sont des associations avec des attributs. Pour « Participation », ça correspond à la performance d'un joueur lors d'un match afin de récupérer des statistiques justement. Ce sont les attributs nb_passes, nb_buts, nb_tirs, nb_assistances.

Pour « donne sanction », la couleur du carton est importante ainsi que l'heure où ça a été donnée.

Projet Football #5, question 1

Question : Comment se nomme l'arbitre semblant avoir une dent contre David Garcia et lui a donné le plus grand nombre de cartons de pénalité (à travers toutes les éditions de la coupe du monde)?

Choses à faire :

-Trouver le numéro de David Garcia ainsi que son équipe (sélection dans Joueur).

-Faire une jointure naturelle entre ce tableau-là et Sanction (sur joueur_numero).

-Grouper les tuples selon arbitre_numero et faire un compte

-Sélection le numéro d'arbitre avec le plus haut compte.

-Faire une jointure naturelle entre ce tuple et le tableau Arbitre.

-Retourner le prénom et le nom de l'arbitre correspondant.

Algèbre relationnelle :

$r_1 \leftarrow \sigma_{(\text{prenom}='David' \text{ AND } \text{nom}='Garcia')}(Joueur)$

$r_2 \leftarrow \Pi_{\text{joueur_numero}} (r_1)$

$r_3 \leftarrow \text{Sanction} \bowtie r_2$

$r_4 \leftarrow \rho_{(\text{arbitre_numero}, \text{nbSanctions})}(\text{arbitre_numero} A_{\text{count}(*)}(r_3))$

$r_5 \leftarrow \rho_{(\text{nbSanctionsMax})}(A_{\text{max}(\text{nbSanctions})}(r_4))$

$r_6 \leftarrow r_4 \bowtie_{\text{nbSanctions}=\text{nbSanctionsMax}} r_5$

$r_7 \leftarrow \text{Arbitre} \bowtie r_6$

$\Pi_{\text{prenom}, \text{nom}}(r_7)$

SQL :

DROP TABLE IF EXISTS t1 CASCADE;

DROP TABLE IF EXISTS t2 CASCADE;

DROP TABLE IF EXISTS t3 CASCADE;

CREATE TABLE t1 as (SELECT * from Sanction where (joueur_numero =(Select joueur_numero from Joueur where (prenom = 'David' AND nom = 'Garcia'))));

CREATE TABLE t2 as (SELECT arbitre_numero, COUNT(*) AS nbSanctions from t1 group by arbitre_numero);

CREATE TABLE t3 as (SELECT arbitre_numero from t2 where (nbSanctions = (Select MAX(nbSanctions) from t2)));

select prenom, nom from Arbitre natural join t3;

Ça donne comme réponse Maxence Gagnon.

Optimisation :

La table t1 est faite en commençant par une sélection, puis une projection suivie d'une autre sélection. C'est plus rapide que de faire une jointure naturelle avec une table ayant le numéro de David Garcia et avec la table Sanction. Maintenant qu'on a les

informations nécessaires, on peut utiliser les fonctions d'agrégats (faire ça avant aurait compté les sanctions qui ont été données à quelqu'un d'autre que David Garcia). Encore là, les commandes ont été optimisées afin de ne pas faire de jointure naturelle et seulement des sélections et projections, sauf à la toute fin, car t3 est une assez longue requête pour qu'on ne soit pas sûr qu'on puisse l'entrer dans la dernière requête afin de servir de sélection. Sinon, ça reste une jointure naturelle avec une seule valeur dans t3 (sauf s'il y a des égalités entre quelques arbitres).

Projet Football #5, question 2

Question : Trouvez le nom du gardien avec le plus de buts dans tous les matchs.

Algèbre relationnelle :

$t_1 \leftarrow \sigma_{(position="Gardien")} \pi_{joueur_numero, prenom, nom} (Joueur)$

$t_2 \leftarrow \pi_{joueur_numero, pays} (PaysJoueur)$

$t_3 \leftarrow \pi_{joueur_numero, edition} (AnneesJoueur)$

$t_4 \leftarrow \pi_{joueur_numero, nb_buts, pays1} (Participation)$

$t_5 \leftarrow t_1 \bowtie t_2$

$t_6 \leftarrow t_5 \bowtie t_3$

$t_7 \leftarrow \pi_{prenom, nom, nb_buts} (t_6 \bowtie t_4)$

$t_8 \leftarrow \pi_{\max(nb_buts)} (t_7)$

$c \leftarrow t_8.max(nb_buts)$

$\pi_{prenom, nom} (\sigma_{(t_7.nb_buts = c)} (t_7))$

SQL :

DROP TABLE IF EXISTS t1 CASCADE;

DROP TABLE IF EXISTS t2 CASCADE;

DROP TABLE IF EXISTS t3 CASCADE;

DROP TABLE IF EXISTS t4 CASCADE;

DROP TABLE IF EXISTS t5 CASCADE;

DROP TABLE IF EXISTS t6 CASCADE;

DROP TABLE IF EXISTS t7 CASCADE;

DROP TABLE IF EXISTS t8 CASCADE;

```

CREATE TABLE t1 as (SELECT joueur_numero, prenom, nom FROM Joueur WHERE
position='Gardien');
CREATE TABLE t2 as (SELECT joueur_numero, pays FROM PaysJoueur);
CREATE TABLE t3 as (SELECT joueur_numero, edition FROM anneesJoueur);
CREATE TABLE t4 as (SELECT joueur_numero, nb_buts, pays1 as pays FROM Participation);
CREATE TABLE t5 as (SELECT * FROM t1 inner join t2 using(joueur_numero));
CREATE TABLE t6 as (SELECT * FROM t5 inner join t3 using(joueur_numero));
CREATE TABLE t7 as (SELECT prenom, nom, nb_buts FROM t6 inner join t4 on (t6.joueur_numero =
t4.joueur_numero AND t6.pays=t4.pays));
CREATE TABLE t8 as (SELECT max(nb_buts) as res FROM t7);
SELECT prenom, nom FROM t7, t8 WHERE (t7.nb_buts = t8.res) GROUP BY prenom, nom;

```

Optimisation de question de 5.2

Voici ma stratégie d'optimisation :

Exécuter les opérations de table dans l'ordre suivant : SELECT, PROJECTION, JOIN.

SELECT et PROJECTION réduisent la taille de la table, et JOIN augmente considérablement la taille de la table. J'ai réduit les tailles de table possibles en effectuant SELECT et PROJECTION sur les quatre premières tables (t1, t2, t3, t4). Ensuite, les opérations JOIN sont effectuées avec la table réduite possible (t5, t6, t7).

Projet Football #5, question 3

Requête : Trouvez tous les joueurs avec leur prenom, leur nom ainsi que le nombre de passes pour chacun d'eux.

Étapes :

- 1- Pour avoir les identifiants des joueurs (numero de joueur) on fait une selection dans Joueur.
- 2- On fait inner Join avec Participations pour avoir nb de passes.
- 3- On trouve le nom, prenom du joueur ainsi que le compte du nombre de passes.

Algebre relationnelle :

$$r_1 \leftarrow \Pi_{\text{joueur_numero, prenom, nom}}(\text{Joueur})$$

$r_2 \leftarrow r_1 \bowtie \pi_{\text{joueur_numero}, \text{nb_passes}}(\text{Participation})$

$\pi_{\text{prenom}, \text{nom}, \text{count}(\text{nb_passes})}(r_2)$

requête SQL :

WITH

r1 AS (SELECT joueur_numero, prenom, nom FROM Joueur),

r2 AS (SELECT joueur_numero, nb_passes FROM Participation)

SELECT r1.prenom, r1.nom, COUNT(r2.nb_passes) as nb_passes

FROM r1 INNER JOIN r2 ON r1.joueur_numero=r2.joueur_numero

Group by r1.prenom, r1.nom ;

Projet Football #5, question 4

Question : L'équipe de quel pays a gagné la première édition de la coupe du monde et avec combien de buts au total?

Choses à faire :

- Trouver le match de la finale de la première édition.
- Trouver le score de la première équipe.
- Trouver le score de la 2^e équipe.
- Comparer les deux scores et prendre le plus grand
- Faire la différence entre les deux
- Retourner le pays de l'équipe gagnante et la différence du score.

Algèbre relationnelle :

$r_1 \leftarrow \pi_{\text{edition, pays1, pays2, date}}(\sigma_{\text{edition} = 1}(\text{InfosMatch}))$

$r_2 \leftarrow \pi_{\text{date}}(\sigma_{\text{rang} = \text{'Finale'}}(\text{Niveau}))$

$r_3 \leftarrow r_1 \bowtie r_2$

$r_4 \leftarrow \pi_{\text{edition, pays1, date}}(r_3)$

$r_5 \leftarrow \pi_{\text{edition, pays2, date}}(r_3)$

$r_6 \leftarrow \text{Score1} \bowtie r_4$

$r_7 \leftarrow \text{Score2} \bowtie r_5$

$r_8 \leftarrow \pi_{\text{pays, score}}(\rho_{(\text{edition, date, pays, score})}(r_6 \cup r_7))$

$r_9 \leftarrow \rho_{(\text{ScoreGagnant})}(A_{\max(\text{score})}(r_8))$

$r_{10} \leftarrow r_8 \bowtie_{\text{score} = \text{ScoreGagnant}} r_9$

$\pi_{\text{pays, score}}(r_{10})$

SQL :

DROP TABLE IF EXISTS t1 CASCADE;

DROP TABLE IF EXISTS t2 CASCADE;

DROP TABLE IF EXISTS t3 CASCADE;

DROP TABLE IF EXISTS t4 CASCADE;

DROP TABLE IF EXISTS t5 CASCADE;

DROP TABLE IF EXISTS t6 CASCADE;

DROP TABLE IF EXISTS t7 CASCADE;

DROP TABLE IF EXISTS t8 CASCADE;

CREATE TABLE t1 as (SELECT edition, pays1, pays2, date from InfosMatch where (edition = 1));

CREATE TABLE t2 as (SELECT date from Niveau where rang='Finale');

CREATE TABLE t3 as (SELECT * from t1 natural join t2);

CREATE TABLE t4 as (Select edition, date, pays1 from t3);


```
CREATE TABLE t5 as (Select edition, date, pays2 from t3);  
CREATE TABLE t6 as (SELECT * from Score1 natural join t4);  
CREATE TABLE t7 as (SELECT * from Score2 natural join t5);  
CREATE TABLE t8 as (Select pays1_score_final as score, pays1 as pays from t6)  
UNION (Select pays2_score_final as score, pays2 as pays from t7);  
SELECT pays, score from t8 where (score= (Select MAX(score) from t8));
```

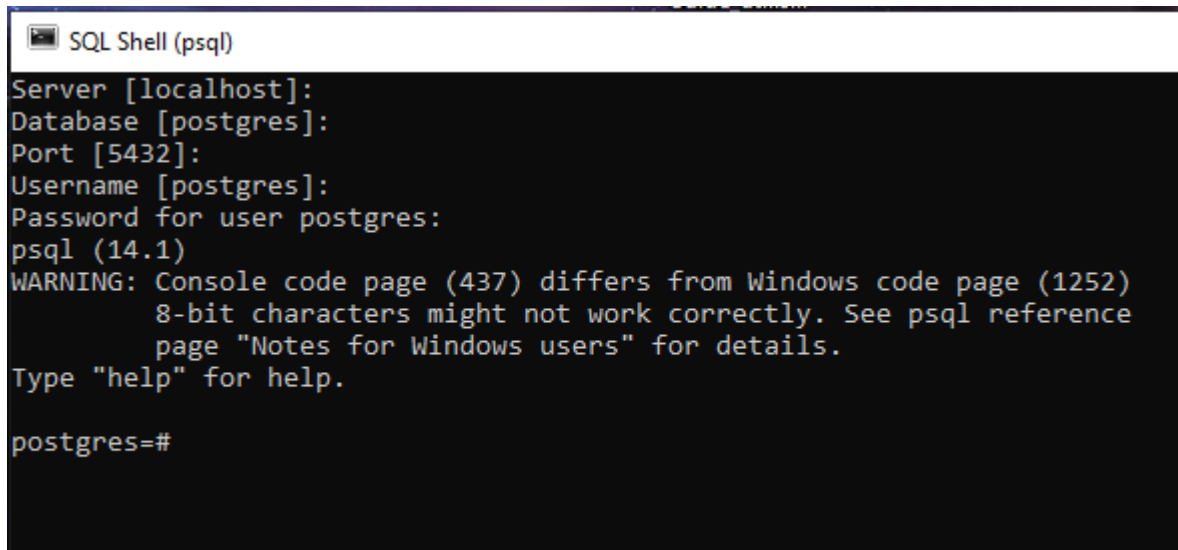
Ça donne comme réponse Canada avec 2 points.

Optimisation :

On commence par faire les sélections pertinentes dans InfosMatch et Niveau avant de faire une jointure naturelle. Ensuite, on fait les tables t4 et t5 afin d'avoir les scores des deux équipes. Cela se fait avec une jointure naturelle entre Score1 et t4 et une autre entre Score2 et t5. Le but de ces deux jointures naturelles séparées était de pouvoir avoir deux tuples différents afin de pouvoir trouver le maximum d'une colonne que l'on nomme « Score ». On trouve à quelle équipe ce score appartient, puis on retourne le pays et le score correspondants.

Guide Utilisateur pour l'application

- 1) Ouvrir le shell de psql et se connecter comme le montre l'image ci-dessous :



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (14.1)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=#
```

- 2) Une fois connecté, on doit créer une base de données qu'on va nommer projet par exemple :

```
postgres=# CREATE DATABASE projet;
```

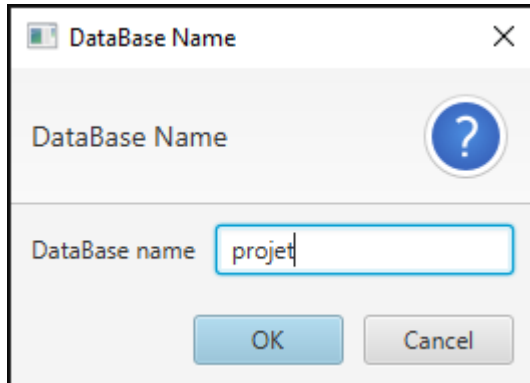
- 3) Une fois la BD créée, on veut importer les commandes SQL (fichier disponible dans le dossier de remise) pour créer nos tables. Il faut écrire : « \i » suivi du « chemin pour se rendre au fichier .txt » voulu, et en finissant par un « / » et le nom du fichier.txt. Une note importante, c'est que le chemin sur Windows est par défaut écrit avec « \ », il sera important de les changer pour des « / » afin que ça fonctionne (exemple : ceci\est\comment\Windows\fonctionne POUR ceci/est/comment/Windows/fonctionne). Voici un exemple :

```
postgres=# \i 'YOUR/FILE/PATH/code_sql.txt'
```

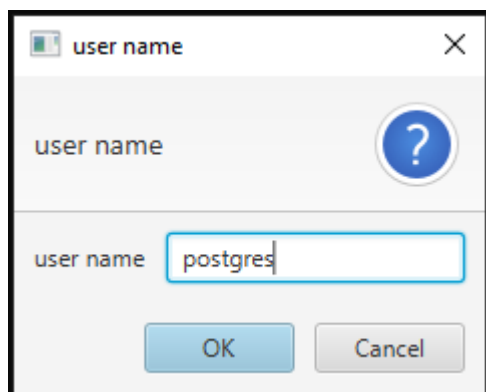
- 4) Après avoir fini les étapes 1 à 3, on peut fermer la fenêtre du Shell.
- 5) Il faut se diriger vers l'emplacement du fichier JAR pour pouvoir l'exécuter. Une fois dans le bon dossier contenant le JAR, on ouvre la ligne de commande (CMD) et on roule la commande suivante :

```
java -jar JavaBD.jar
```

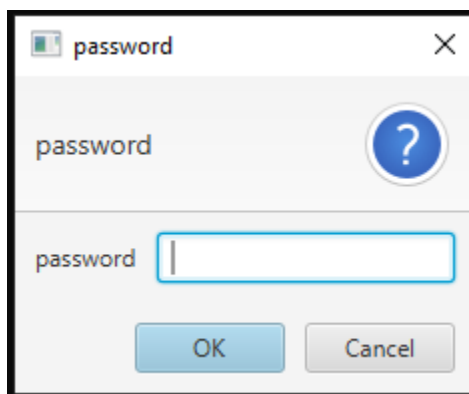
- 6) À ce moment, l'application va démarrer et nous allons avoir la première fenêtre qui s'affiche comme l'exemple ci-dessous. Il sera important d'indiquer dans cette fenêtre le nom de notre BD que nous avons initialisé à l'étape 2. Dans notre cas, c'est projet. On poursuit avec le bouton « ok »



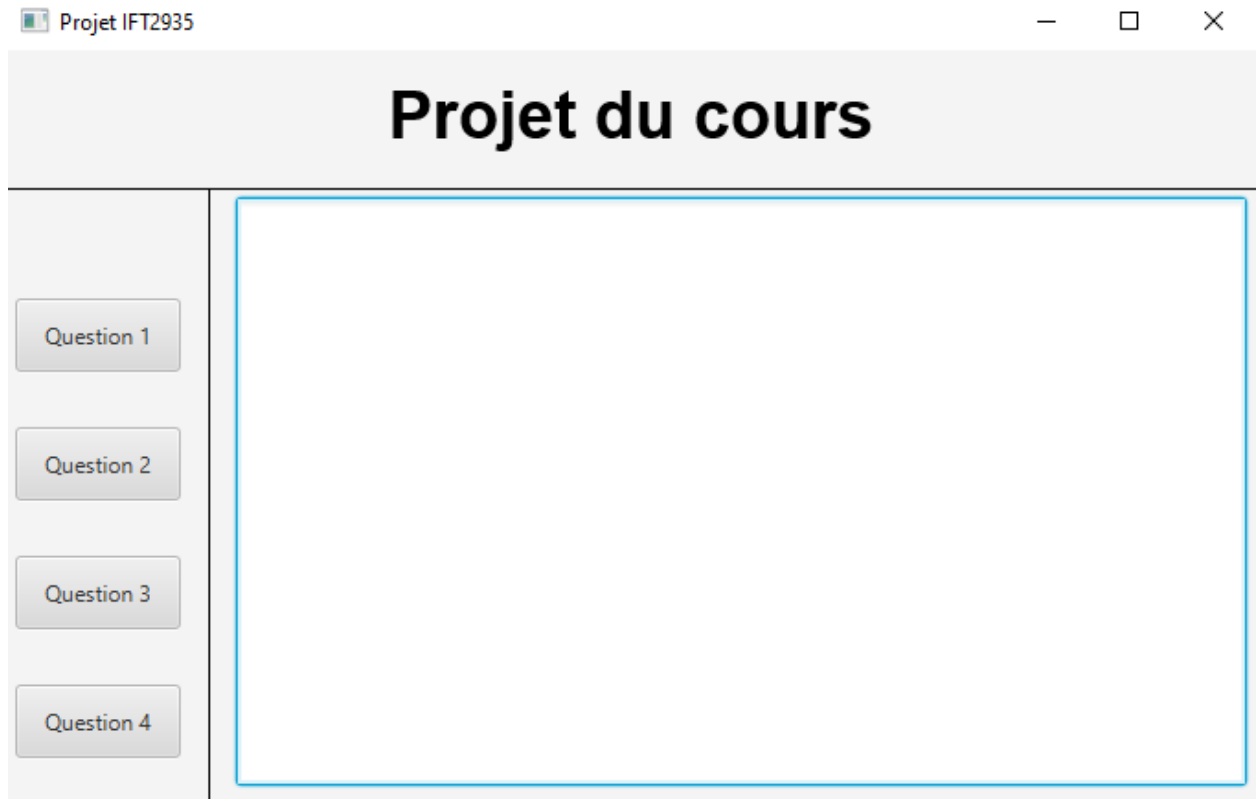
- 7) La deuxième fenêtre nous demande d'indiquer le nom de username de notre BD. Comme on peut le voir dans la capture d'écran de la question 1, le username est postgres par défaut. Voici la suite de l'exécution :



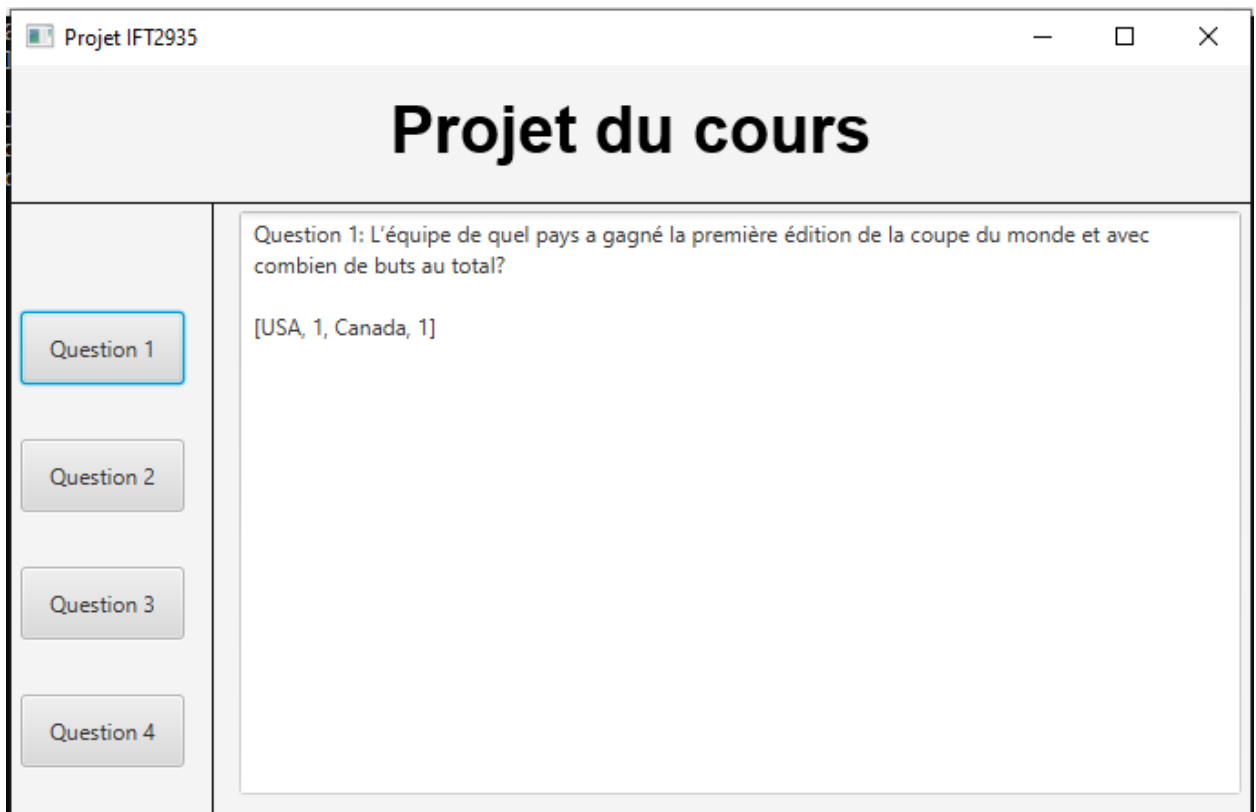
- 8) La troisième fenêtre nous demande d'écrire le mot de passe de notre nom utilisateur de la BD, le même mot de passe qu'on a utilisé pour ouvrir une session dans le Shell à la question 1. Voici la prochaine fenêtre :



- 9) Finalement, notre application va démarrer et aura la forme suivante comme état initial :



- 10) Lorsqu'on va cliquer sur les différents boutons, nous allons voir apparaitre dans le grand rectangle blanc nos différentes requêtes. Voici les différents résultats que nous obtenons pour chaque bouton :



Projet IFT2935

Projet du cours

Question 1

Question 2

Question 3

Question 4

Question 2: Trouvez le nom du gardien avec le plus de buts dans tous les matchs

[James, Smith]

Projet IFT2935

Projet du cours

Question 1

Question 2

Question 3

Question 4

Question 3: Trouvez tous les joueurs avec leur prenom, leur nom ainsi que le nombre de passes pour chacun d'eux

[Joseph, Davis, 1, Jerry, Boulet, 1, Gabriel, Lebeau, 3, James, Smith, 4, William, Jones, 1, David, Garcia, 2]

Projet du cours

Question 1

Question 2

Question 3

Question 4

Question 4: Comment se nomme l'arbitre semblant avoir une dent contre David Garcia et lui a donné le plus grand nombre de cartons de pénalité (à travers toutes les éditions de la coupe du monde)?

[Maxence, Gagnon]