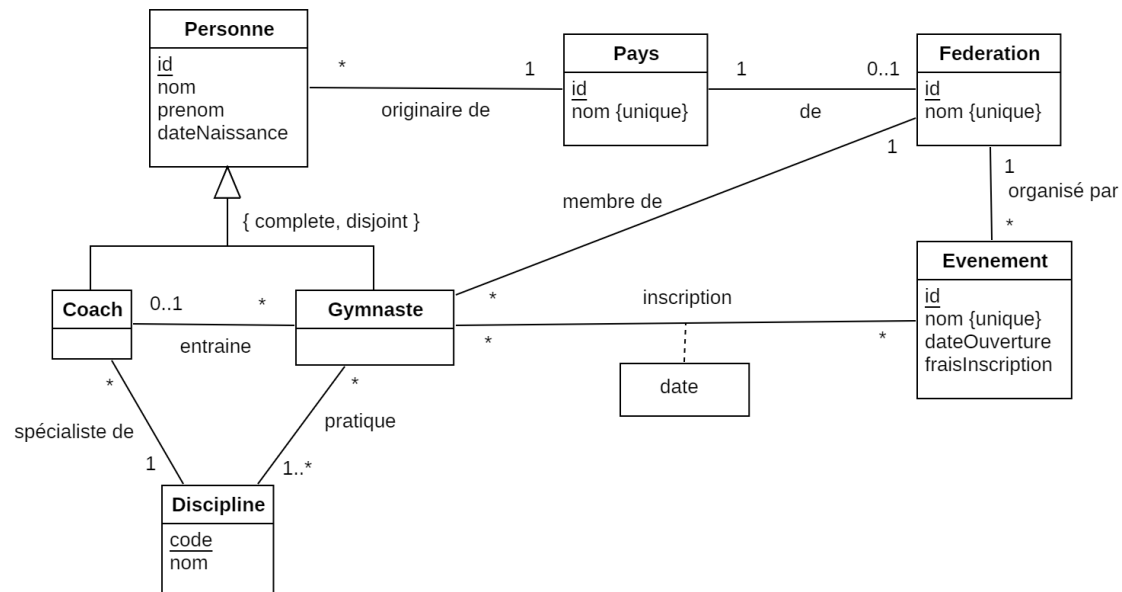


Laboratoire no 3

Donnée

Soit une base de données permettant de gérer principalement des gymnastes et les événements auxquels ils peuvent s'inscrire.

Schéma EA :



Relations/Tables :

- Discipline(code, nom)
- Pays(id, nom)
- Federation(id, idPays, nom)
- Personne(id, idPays, nom, prenom, dateNaissance)
- Coach(idPersonne, codeDiscipline)
- Gymnaste(idPersonne, idCoach, idFederation)
- Gymnaste_Discipline(idGymnaste, codeDiscipline)
- Evenement(id, idFederation, nom, dateOuverture, fraisInscription)
- Inscription(idGymnaste, idEvenement, date)

Vues :

- vCoach(id, nom, prenom, dateNaissance, idPays, codeDiscipline)
- vGymnaste(id, nom, prenom, dateNaissance, idPays, idFederation, idCoach)

Le script SQL permettant de construire et peupler la base de données décrite ci-dessus est disponible sur Cyberlearn.

Partie 1

Pour chaque point ci-dessous, écrire une requête SQL permettant d'obtenir les informations demandées de la manière la plus simple et efficace possible

1. Les frais moyens d'inscription des événements organisés par la fédération 'BRITISH GYMNASTICS'.
2. Les gymnastes qui pratiquent la discipline dont leur entraîneur est le spécialiste et qui sont originaires du même pays.
3. Les coaches nés entre 1981 et 1996 (années comprises) qui entraînent au moins un gymnaste plus âgé qu'eux.
4. Les coaches qui entraînent plus de 2 gymnastes qui se sont inscrits à au moins 1 événement.
5. 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.
6. 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.
7. 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).
8. 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.
9. 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.
10. Pour chaque gymnaste, indiquer le nombre d'événements auxquels il est inscrit et qui n'ont pas encore commencé.
11. 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%.
12. 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.
13. Les gymnastes qui ne se sont inscrits qu'à des événements organisés par leur fédération

Partie 2

Ecrire un script SQL qui permet de rajouter la vérification (CI) suivante :

Tout coach doit avoir au moins 18 ans.

Directives/indications

- Préfixer le nom de chaque attribut par celui de la table dès que plus d'une table est impliquée
- Nommer de manière claire les colonnes des résultats (utiliser des alias si le nom n'est pas suffisamment parlant)
- Afficher les dates au format "jj.mm.aaaa"
- Ne pas faire les jointures dans le WHERE
- Eviter, sauf nécessité absolue, d'utiliser des FROM (SELECT ...)
- Sauf indication contraire, quand on demande les informations sur les données d'un type d'entité, cela signifie les attributs suivants :
 - Gymnaste : id, nom, prenom (attributs issus de Personne/vGymnaste)
 - Coach : id, nom, prenom (attributs issus de Personne/vCoach)
 - Discipline : code, nom
 - Evenement : nom
 - Pays : nom
 - Federation : nom

Points à réaliser/rendre

- Un rapport contenant
 - Partie 1, pour chaque résultat demandé
 - Le numéro ainsi que l'énoncé de la requête
 - Le code SQL de la requête (au format texte)
 - Les résultats obtenus sous forme de tableau (par exemple une capture d'écran du résultat dans DBeaver)
 - Vos éventuelles remarques
 - Partie 2
 - Le script SQL qui permet d'ajouter la vérification de la CI
 - Les tests effectués pour valider le bon fonctionnement de cette vérification

A réaliser

Par groupe de 3

Travail à rendre le 05.12.2024 à 16h35

A remettre

1. Le rapport imprimé
2. Le rapport au format PDF sur CyberLearn