

M1104 – TP1 – Compléments

EXERCICE 1 :

Soit le modèle relationnel suivant :

ELEVE			CLASSE		PROFESSEUR		SERVICE		
numE	nomE	nomC	nomC		numP	nomP	numP	nomC	nbh
10	FRICOTIN	CM1	CM1		P1	DUPONT	P1	CM1	12
11	FILOCHARD	CM2	CM2		P2	DUBOIS	P1	CM2	25
12	CROQUIGNOL	CE1	CE1		P3	DUGENOU	P2	CE1	18
13	RIBOUDINGUE	CE2	CE2		P4	DURACUIRE	P3	CE2	30
14	TARZAN	CM1			P5	DUBALAI	P3	CM2	28
15	TINTIN	CM2			P6	DUVAL	P4	CE1	15
16	MILOU	CE1					P4	CE2	30
17	SPIROU	CE2					P5	CE1	32
18	FANTASIO	CM1					P5	CM1	18
19	MICKEY	CM2					P6	CM1	25
20	DONALD	CE1					P6	CE2	28
22	BOULE	CE2							
23	BILL	CM1							

1. Comprendre la base de données.
2. Définir les clés primaires et étrangères. Ajouter des contraintes :
 - Tous les champs doivent avoir une valeur.
 - Le nombre d'heure d'un professeur donné dans une classe donnée ne doit pas excéder 50h.
 - La longueur du nom de la classe ne doit pas excéder 3 caractères.
3. Créer les tables sous postgresql.
4. Visualiser le contenu de chaque table.
5. Tenter d'insérer des nouvelles valeurs « incorrectes » : par exemple
 - 2 tuples avec la même clé primaire
 - 1 professeur sans nom
 - 1 nouveau service de 60h.
 - 1 nouveau service pour le professeur P7.
6. Tenter de supprimer le professeur P1.

EXERCICE 2 :

Soit le modèle relationnel suivant :

```

joueur(nom, prenom, age, nationalite)
rencontre(nomgagnant, nomperdant, nomtournoi, annee, score )
gain(nomjoueur, nomtournoi, annee, rang , prime )
sponsor(nom, nomtournoi, annee , montant )
tournoi(nom , pays)
  
```

Il ne peut pas y avoir deux joueurs possédant le même nom de famille. Pour simplifier, le score sera stocké comme une chaîne de caractères (par exemple '6/3-6/4'), et le rang également ('gagnant', 'finaliste', etc.).

1. Comprendre la base de données.
2. Définir les clés primaires et étrangères. Ajouter des contraintes d'intégrité.