

1- Identifier les clés étrangères sur le schéma de la BD
Effet (code, nomEffet, texte)

Deck(id, nomDeck, dateCreation, #nomClasse, #email)

Carte(nomCarte, mana, prix, attaque, vie, #nomClasse)

Classe(nomClasse)

Joueur(email, nomJoueur, mdp, enregistrement, pays)

RapourEffet(#nomCarte, #code)

Rappartient(#id, #nomCarte, nbFois)

Rpossede(#nomCarte, #email, nbOccurence)

RgagneAvec(#nomClasse, #email, nbVictoire)

Rami(#email1, # email2)

2- Lister tous les attributs de la BD et leur donner un type (faire un tableau).

nom	Type
code	Char(6) not null
nomeffet	Varchar(20)
texte	Varchar(50)
id	Char(3) not null
nomdeck	Varchar(20)
Datecreation	Date
nomclasse	Varchar(20) not null
Email	Varchar(50) not null
Nomcarte	Varchar(20) not null
Mana	Number(2)
prix	Number (2)
attaque	Number(2)
vie	Number(2)
nomjoueur	Varchar(20)
mdp	Varchar(20)
enregistrement	Date
pays	Varchar(20)
Nbfois	Number(2)
nboccurence	Number(2)
nbvictoire	Number(2)

3- Créer les tables avec les clés primaires et les clés étrangères, nommer les contraintes de clés étrangères.

```
CREATE TABLE Effet
```

```
(  
  code char(6) NOT NULL,  
  nomEffet VARCHAR(20),  
  texte VARCHAR(50),  
  CONSTRAINT cp_table_effet_code PRIMARY KEY (code)  
);
```

```
CREATE TABLE Joueur
```

```
(  
  email VARCHAR(50) NOT NULL,  
  nomJoueur VARCHAR(20),  
  mdp VARCHAR(20),  
  enregistrement DATE,  
  pays VARCHAR(20),  
  CONSTRAINT cp_table_joueur_email PRIMARY KEY (email)  
);
```

```
CREATE TABLE Classe
```

```
(  
  nomClasse VARCHAR(20) NOT NULL,  
  CONSTRAINT cp_table_classe_nomclasse PRIMARY KEY (nomClasse)  
);
```

```
CREATE TABLE Deck
```

```
(  
  id char(3) NOT NULL,  
  nomDeck VARCHAR(20),  
  dateCreation DATE,  
  nomClasse VARCHAR(20) NOT NULL,  
  email VARCHAR(50) NOT NULL,  
  CONSTRAINT cp_table_Deck_id PRIMARY KEY (id),  
  CONSTRAINT ce_table_Deck_email FOREIGN KEY (email) REFERENCES Joueur(email),  
  CONSTRAINT ce_table_Deck_nomclasse FOREIGN KEY (nomClasse) REFERENCES  
  Classe(nomClasse)  
);
```

```
CREATE TABLE Carte
```

```
(  
  nomCarte VARCHAR(20) NOT NULL,  
  mana NUMBER(2),  
  prix NUMBER(2),  
  attaque NUMBER(2),  
  vie NUMBER(2),  
  nomClasse VARCHAR(20) NOT NULL,  
  CONSTRAINT cp_table_Carte_nomCarte PRIMARY KEY (nomCarte),
```

```
        CONSTRAINT ce_table_Carte_nomClasse FOREIGN KEY (nomClasse) REFERENCES
            Classe(nomClasse)
    );
CREATE TABLE RapourEffet
(
    nomCarte VARCHAR(20) NOT NULL,
    code CHAR(6) NOT NULL,
    CONSTRAINT cp_rapoureffet_nomcartecode PRIMARY KEY (nomCarte,code),
    CONSTRAINT ce_table_rapoureffet_nomcarte FOREIGN KEY (nomCarte) REFERENCES
        Carte(nomCarte),
    CONSTRAINT ce_table_rapoureffet_code FOREIGN KEY (code) REFERENCES Effet(code)
);
CREATE TABLE Rappartient
(
    id CHAR(3) NOT NULL,
    nomCarte VARCHAR(20) NOT NULL,
    nfois NUMBER(2),
    CONSTRAINT cp_rappartient_idnomcarte PRIMARY KEY (id,nomCarte),
    CONSTRAINT ce_table_rappartient_id FOREIGN KEY (id) REFERENCES Deck(id),
    CONSTRAINT ce_table_rappartient_nomcarte FOREIGN KEY (nomCarte) REFERENCES
        Carte(nomCarte)
);
CREATE TABLE Rpossede
(
    nomCarte VARCHAR(20) NOT NULL,
    email VARCHAR(50) NOT NULL,
    nbOccurence NUMBER(2),
    CONSTRAINT cp_rpossede_nomcarteemail PRIMARY KEY (nomCarte,email),
    CONSTRAINT ce_table_rpossede_nomcarte FOREIGN KEY (nomCarte) REFERENCES
        Carte(nomCarte),
    CONSTRAINT ce_table_rpossede_email FOREIGN KEY (email) REFERENCES Joueur(email)
);
CREATE TABLE Rgagneavec
(
    nomclasse VARCHAR(20) NOT NULL,
    email VARCHAR(50) NOT NULL,
    nbvictoire NUMBER(2),
    CONSTRAINT cp_rgagneavec_nomclasseemail PRIMARY KEY (nomclasse,email),
    CONSTRAINT ce_table_rgagneavec_nomclasse FOREIGN KEY (nomclasse) REFERENCES
        Classe(nomClasse),
    CONSTRAINT ce_table_rgagneavec_email FOREIGN KEY (email) REFERENCES Joueur(email)
);
CREATE TABLE Rami
(
    email1 VARCHAR(50) NOT NULL,
```

```
email2 VARCHAR(50) NOT NULL,  
CONSTRAINT cp_rami_email1email2 PRIMARY KEY (email1,email2),  
CONSTRAINT ce_table_rami_email1 FOREIGN KEY (email1) REFERENCES Joueur(email),  
CONSTRAINT ce_table_rami_email2 FOREIGN KEY (email2) REFERENCES Joueur(email)
```

);

4- En complétant les données fournies dans la partie 2 (mathématiques) , concevoir un jeu d'essai **cohérent** respectant les contraintes suivantes :

- a. Le coût en mana d'une carte varie de 0 à 10
- b. Un deck comporte 8 cartes
- c. Nombre de tuple par table :

Effet	5
Deck	fournis
Carte	20
Classe	fournis
Joueur	fournis
Rappartient	40
RapourEffet	10
Rpossède	40
RgagneAvec	12 (compléter les tuples fournis)
Rami	5

Pour construire votre jeu d'essai vous devez compléter le fichier excel disponible sur le cours en ligne. Chaque onglet concerne une table.

L'onglet "Effet" contient toutes les informations permettant de construire la requête SQL "INSERT" à copier-coller dans votre environnement SQL pour exécution. Les tuples sont fournis pour les tables Deck, Classe, Joueur. La table RgagneAvec est à compléter. Les tables Effet, Carte, Rappartient, RapourEffet, Rami sont à compléter intégralement. Toutes les cellules vertes vides du fichier sont donc à compléter.

Reproduire cette démarche pour toutes les autres tables :

1. Compléter les cases bleues et orange (noms des attributs et leurs types)
2. Compléter toutes les cases vertes (valeurs des attributs), bien les choisir pour respecter la cohérence des données et les contraintes de clés primaires et étrangères.
3. Ecrire la formule de concaténation pour une ligne. Pour la table Effet, la cellule F3 contient **=CONCATENER(A3;B3;" ";"C3;" ";"D3;E3)**

4. Reproduire la formule sur les lignes suivantes par copie incrémentale

Remarque : La qualité du jeu d'essai sera évaluée.