# Les courses de chevaux

# Récolte des besoins

Un parieur assidu des champs de courses et des bases de données, voudrait mémoriser, dans une base de données

relationnelle, les courses de chevaux, les paris qu'il a faits et les résultats.

Plus précisément, il veut enregistrer les informations suivantes pour chaque course :

- Le nom et la date (ex : Prix d'Amérique, 21-07-92)

- Le numéro, le nom chevaux partants

- Ses paris, avec pour chacun :

o Le type de pari (couplé, tiercé, quarté, quinté...).

▪ Couplé = Pari sur 2 chevaux (1er et 2nd)

▪ Tiercé = Pari sur 3 chevaux (1er, 2ème et 3ème)

▪ Etc...

o La somme jouée et les numéros de chevaux dans l'ordre du pari.

Une fois la course jouée, on enregistre aussi :

- Le résultat : l'ordre d'arrivée des chevaux.

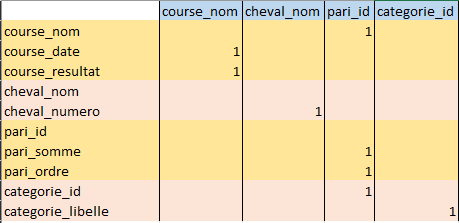
- Le gain total du parieur pour la course (couplé = somme jouée x 2, tiercé = somme jouée \* 3, etc...).

Les noms de chevaux sont uniques et les noms de courses sont uniques (à un instant donné)

# Dictionnaire des données

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entité | Mnémonique | Signification | Type(longueur) | Contraintes |
| Course | course\_nom | nom de la course | VARCHAR(50) | identifiant |
| course\_date | date de la course | DATE | obligatoire |
|  | course\_resultat | résultat de la course | VARCHAR(50) | obligatoire |
| Cheval | cheval\_nom | nom du cheval | VARCHAR(50) | identifiant |
|  | cheval\_numero | numéro du cheval | TINYINT | obligatoire |
| Pari | pari\_id | identifiant du pari | INT(11) | obligatoire, ai |
|  | pari\_somme | somme jouée pour le pari | DECIMAL(10,2) | obligatoire |
|  | pari\_ordre | ordre d’arrivée des chevaux du pari | VARCHAR(50) | obligatoire |
| Categorie | categorie\_id | identifiant de la catégorie de pari | INT(11) | identifiant |
| categorie\_libelle | libellé de la catégorie de pari (couplé,tiercé…) | VARCHAR(50) | obligatoire |

# Matrice



# Règles de gestion

A 1 course participent 1 ou plusieurs chevaux

1 cheval participe à 1 ou plusieurs courses

1 pari est fait pour 1 seule course

Pour 1 course on fait 1 ou plusieurs pari

A un pari correspond 1 seule categorie de pari  
A une categorie de pari correspondent 0 ou plusieurs paris

# Dépendances fonctionnelles

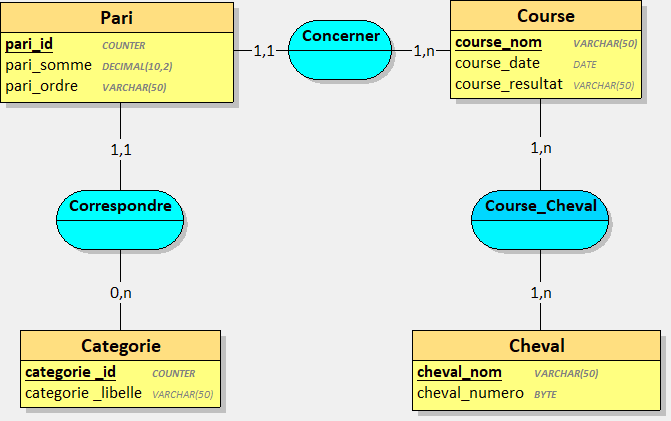
course\_nom -> course\_date, course\_resultat

cheval\_nom -> cheval\_numero

pari\_id -> pari\_somme, #course\_nom, #type\_id

categorie \_id -> categorie \_libelle

# Modèle conceptuel des données



# Modèle relationnel des données

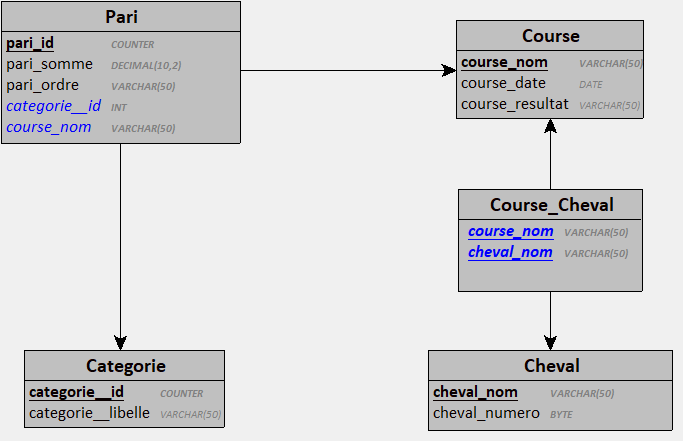
Categorie(categorie \_id, categorie \_libelle)

Course(course\_nom, course\_date, course\_resultat)

Pari(pari\_id, pari\_somme, pari\_ordre, # categorie \_id, #course\_nom)

Cheval(cheval\_nom, cheval\_numero)

Course\_Cheval(#course\_nom, #cheval\_nom)



# Création de la base de données (DDL) SQL server

drop table if exists Course\_Cheval;

drop table if exists Cheval;

drop table if exists Pari;

drop table if exists Course;

drop table if exists Categorie;

create table Categorie

(

categorie\_id int identity(1,1)

,categorie\_libelle varchar(50) not null

,primary key (categorie\_id)

)

;

create table Course

(

course\_nom varchar(50) not null

,course\_date date not null

,course\_resultat varchar(50) not null

,primary key (course\_nom)

)

;

-- Est-ce acceptable d'intégrer directement les foreign key au creat table ?

create table Pari

(

pari\_id int identity(1,1)

,pari\_somme decimal(10,2) not null

,pari\_ordre varchar(50) not null

,categorie\_id int

,course\_nom varchar(50)

,primary key (pari\_id)

,foreign key (categorie\_id) references Categorie(categorie\_id)

,foreign key (course\_nom) references Course(course\_nom)

)

;

create table Cheval

(

cheval\_nom varchar(50) not null

,cheval\_numero tinyint not null

,primary key (cheval\_nom)

)

;

create table Course\_Cheval

(

course\_nom varchar(50)

,cheval\_nom varchar(50)

,primary key (course\_nom, cheval\_nom)

,foreign key (course\_nom) references Course(course\_nom)

,foreign key (cheval\_nom) references Cheval(cheval\_nom)

)

;