

---

## TD-3 BIS : CREATION BD ET JOINTURES

**Objectif** : 1. Révisons des jointures. SQL.

**Prérequis** : slides du cours.

---

### EXERCICE 1 : Création DB

**1.1** Créez un schéma relationnel pour stocker les données relatives au tour de France. La BD doit pouvoir contenir les données personnelles de chaque cycliste : nom, prénom, date de naissance, numéro de dossard et équipe. L'équipe a un nom et un directeur sportif (caractérisé par nom et prénom et date de naissance). Chaque équipe inclut plusieurs cyclistes. Le tour est composé par plusieurs étapes. Pour chaque étape il faudrait pouvoir stocker le lieu de départ, le lieu d'arrivée, le dénivelé, la date, les participants, leur classement et leur temps. La BD doit pouvoir stocker les données de plusieurs éditions du tour de France.

```
personne[idPers,nom, prenom, dateNaissance] ;
equipe[idEquipe,nomEquipe,@directeurSportif] ;
cycliste[@idCycliste, dossard,@equipe] ;
etape[idEtape,lieuDepart,lieuArrivee,denivele,date, annee]
classementEtape[@idEtape,@idCycliste,classement,temps] ;
```

```
cycliste.idCycliste -> personne.idPers
cycliste.equipe -> equipe.idEquipe
equipe.directeurSportif -> personne.idPers
classementEtape.idEtape -> etape.idEtape ;
classementEtape -> cycliste.idCycliste ;
```

### 1.2

```
personne.idPers = int
personne.nom = varchar
personne.prenom=varchar
personne.dateNaissance = date
```

```
equipe.idEquipe = int
equipe.nomEquipe = varchar
equipe.directeurSportif = int
```

```
cycliste.idCycliste = int
cycliste.dossard = int
cycliste.equipe = int
```

```
etape.idEtape = int
etape.lieuDepart = varchar
etape.lieuArrivee = varchar
etape.denivele = int
etape.date = date
etape.annee = int
```

classementEtape.idEtape = int  
classementEtape.idCycliste = int  
classementEtape.classement = int  
classementEtape.temps = time

---

### EXERCICE 2 : Algèbre relationnelle

**2.1** A l'aide du schéma défini dans l'exercice 1, écrire les requêtes relationnelles permettant de répondre aux questions suivantes :

1) Quels sont les noms des cyclistes ayant participé à au moins une étape du tour de France 2022 ?

$\pi_{\text{nom}} (\sigma_{\text{annee}=2022} (\text{cycliste} \bowtie_{\text{idCycliste}=\text{idPersonne}} \text{personne}))$ ;

2) Listez les années où un Tour de France a eu lieu.

$\pi_{\text{annee}} (\text{etape})$

3) Listez les noms et prénoms de cyclistes ayant participé à au moins une étape du tour de France (peu importe l'édition)

$\pi_{\text{nom}, \text{prenom}} (\sigma_{\text{annee}=2022} (\text{cycliste} \bowtie_{\text{idCycliste}=\text{idPersonne}} \text{personne} \bowtie \text{classementEtape}))$ ;

4) Listez les noms et prénoms de cyclistes ayant gagné au moins une étape du tour de France 2022

$\pi_{\text{nom}, \text{prenom}} (\sigma_{\text{annee}=2022 \text{ AND } \text{classement}=1} (\text{cycliste} \bowtie_{\text{idCycliste}=\text{idPersonne}} \text{personne} \bowtie \text{classementEtape}))$ ;

5) Listez les noms des équipes ayant gagné au moins une étape du Tour de France 2022.

$\pi_{\text{nom},} (\sigma_{\text{annee}=2022 \text{ AND } \text{classement}=1} (\text{cycliste} \bowtie_{\text{equipe}=\text{idEquipe}} \text{equipe} \bowtie \text{classementEtape}))$ ;

### **2.2** Traduire en SQL les requêtes

1) *select nom*

*from cycliste inner join personne on idCycliste=idPersonne natural join classementEtape  
where tour.annee=2022 ;*

2) *Select distinct annee*

*from tour;*

3) *select nom, prenom*

*from cycliste inner join personne on idCycliste=idPersonne natural join classementEtape ;*

4) *select nom, prenom*

*from cycliste inner join personne on idCycliste=idPersonne natural join classementEtape  
where annee=2022 and classement=1;*

5) *select nom*

*from cycliste inner join equipe on equipe=idEquipe natural join classementEtape  
where annee=2022 and classement=1;*