**Bases de données**

**TP N°2**

1. 1.

*Note : on n’a pas besoin de mettre* **NOT** **NULL** *sur une primary key.*

*Note 2 : rien n’est indiqué sur la nécessité d’avoir des champs nuls ou non.*

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `etudiant` **(**

`numetu` **int(**11**)** **NOT** **NULL,**

`nom` **varchar(**50**),**

`prenom` **varchar(**50**),**

`datenaiss` **date,**

`rue` **varchar(**100**),**

`cp` **int(**11**),**

`ville` **varchar(**50**),**

**PRIMARY** **KEY** **(**`numetu`**)**

**)** ENGINE**=**InnoDB **DEFAULT** CHARSET**=**latin1**;**

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `matiere` **(**

`codemat` **varchar(**3**)** **NOT** **NULL,**

`libelle` **varchar(**50**),**

`coef` **float,**

**PRIMARY** **KEY** **(**`codemat`**)**

**)** ENGINE**=**InnoDB **DEFAULT** CHARSET**=**latin1**;**

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `epreuve` **(**

`numepreuve` **int(**11**)** **NOT** **NULL,**

`datepreuve` **date,**

`lieu` **varchar(**50**),**

`codemat` **varchar(**3**)** **NOT** **NULL,**

**PRIMARY** **KEY** **(**`numepreuve`**),**

**FOREIGN** **KEY** **(**`codemat`**) REFERENCES** matiere(codemat)**,**

**)** ENGINE**=**InnoDB **DEFAULT** CHARSET**=**latin1**;**

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `notation` **(**

`numetu` **int(**11**)** **NOT** **NULL,**

`numepreuve` **int(**11**)** **NOT** **NULL,**

`note` **float,**

**PRIMARY** **KEY** **(**`numetu`**,**`numepreuve`**),**

**FOREIGN** **KEY** **(**`numetu`**)** **REFERENCES** etudiant**(**numetu**),**

**FOREIGN** **KEY** **(**`numepreuve`**)** **REFERENCES** epreuve**(**numepreuve**)**

**)** ENGINE**=**InnoDB **DEFAULT** CHARSET**=**latin1**;**

2. & 3. Quelques insertions…

**INSERT** **INTO** `etudiant` **(**`numetu`**,** `nom`**,** `prenom`**,** `datenaiss`**,** `rue`**,** `cp`**,** `ville`**)** **VALUES** **(**'110'**,** 'Dupont'**,** 'Albert'**,** '1980-06-01'**,** 'Rue de Crimée'**,** '69001'**,** 'Lyon'**);**

**INSERT** **INTO** `matiere` **(**`codemat`**,** `libelle`**,** `coef`**)** **VALUES** **(**'STA'**,** 'Statistique'**,** '0.4'**),** **(**'INF'**,** 'Informatique'**,** '0.4'**);**

**INSERT** **INTO** `epreuve` **(**`numepreuve`**,** `datepreuve`**,** `lieu`**,** `codemat`**)** **VALUES** **(**'11031'**,** '2003-12-15'**,** 'Salle 191L'**,** 'STA'**),** **(**'11032'**,** '2004-04-01'**,** 'Amphi G'**,** 'STA'**);**

**INSERT** **INTO** `notation` **(**`numetu`**,** `numepreuve`**,** `note`**)** **VALUES** **(**'300'**,** '11031'**,** '14'**),** **(**'421'**,** '21032'**,** **NULL);**

1. 1. & 2. Déjà fait dans la création de table page 1

3. On peut toujours rentrer cette requête, mais MySQL ne la prendra pas en compte.

**ALTER** **TABLE** `notation` **ADD** **CONSTRAINT** checkNote

**CHECK** **(**note **>=** 0 **and** note **<=** 20**);**

4.

**INSERT** **INTO** `etudiant` **(**`numetu`**,** `nom`**,** `prenom`**,** `datenaiss`**,** `rue`**,** `cp`**,** `ville`**)** **VALUES** **(**'1100'**,** 'TOTO'**,** 'TAlbert'**,** '1981-07-01'**,** 'Rue de Crimée'**,** '75019'**,** 'Paris'**);**

On s’aperçoit qu’on ne peut pas rentrer l’épreuve 31031 car elle n’est pas dans la base de données « epreuve » :

**INSERT** **INTO** `notation` **(**`numetu`**,** `numepreuve`**,** `note`**)** **VALUES** **(**'1100'**,** '31031'**,** '13'**);**

MySQL a répondu: Documentation

#1452 - Cannot add or update a child row: a foreign key constraint fails (`notation`, CONSTRAINT `notation\_ibfk\_2` FOREIGN KEY (`numepreuve`) REFERENCES `epreuve` (`numepreuve`))

1. 1.

**SELECT** **\*** **FROM** etudiant **ORDER** **BY** `etudiant`**.**`nom` **DESC**

2.

**SELECT** `libelle`**,** **SUBSTRING(**`coef` **\*** 100**,** 1**,** 2**)** **FROM** `matiere`

*Note : on utilise* **SUBSTRING** *pour tronquer le résultat à deux chiffres.*

3.

**SELECT** `nom`**,** `prenom` **FROM** `etudiant` **WHERE** `ville` **=** 'Lyon'

4.

**SELECT** `note` **FROM** `notation` **WHERE** `note` **>=** 10

5.

**SELECT** **\*** **FROM** `epreuve`

**WHERE** `datepreuve` **BETWEEN** '2004-01-01' **AND** '2004-06-30'

6.

**SELECT** **COUNT(\*)** **FROM** `epreuve`

7.

**SELECT** **COUNT(\*)** **FROM** `notation` **WHERE** `note` **IS** **NULL**

8.

**SELECT** `note`**,** `nom`**,** `prenom` **FROM** `notation`**,** `etudiant`

**WHERE** `notation`**.**`numetu` **=** `etudiant`**.**`numetu`

9.

**SELECT** **AVG(**`note`**)** **AS** `moyenne`**,** `nom`**,** `prenom`

**FROM** `notation`**,** `etudiant`

**WHERE** `notation`**.**`numetu` **=** `etudiant`**.**`numetu`

**GROUP** **BY** `notation`**.**`numetu`

**ORDER** **BY** `moyenne` **DESC**

10.

**SELECT** **AVG(**`note`**)** **AS** `moyenne`**,** `matiere`**.**`libelle`

**FROM** `matiere`**,** `notation`**,** `epreuve`

**WHERE** `matiere`**.**`codemat`**=**`epreuve`**.**`codemat`

**AND** `epreuve`**.**`numepreuve`**=**`notation`**.**`numepreuve`

**AND** `matiere`**.**`codemat` **IN**

**(SELECT** `codemat` **FROM** `epreuve`

**GROUP** **BY** `codemat`

**HAVING** **count(**`codemat`**)** **>** 1**)**

**GROUP** **BY** `matiere`**.**`codemat`

11.

**SELECT** **AVG(**`note`**)** **AS** `moyenne`**,** `epreuve`**.**`numepreuve`

**FROM** `notation`**,** `epreuve`

**WHERE** `epreuve`**.**`numepreuve`**=**`notation`**.**`numepreuve`

**AND** `epreuve`**.**`numepreuve` **IN**

**(SELECT** `numepreuve` **FROM** `notation`

**WHERE** note **IS** **NOT** **NULL**

**GROUP** **BY** `numepreuve`

**HAVING** **count(**`numetu`**)** **<** 6**)**

**GROUP** **BY** `epreuve`**.**`numepreuve`