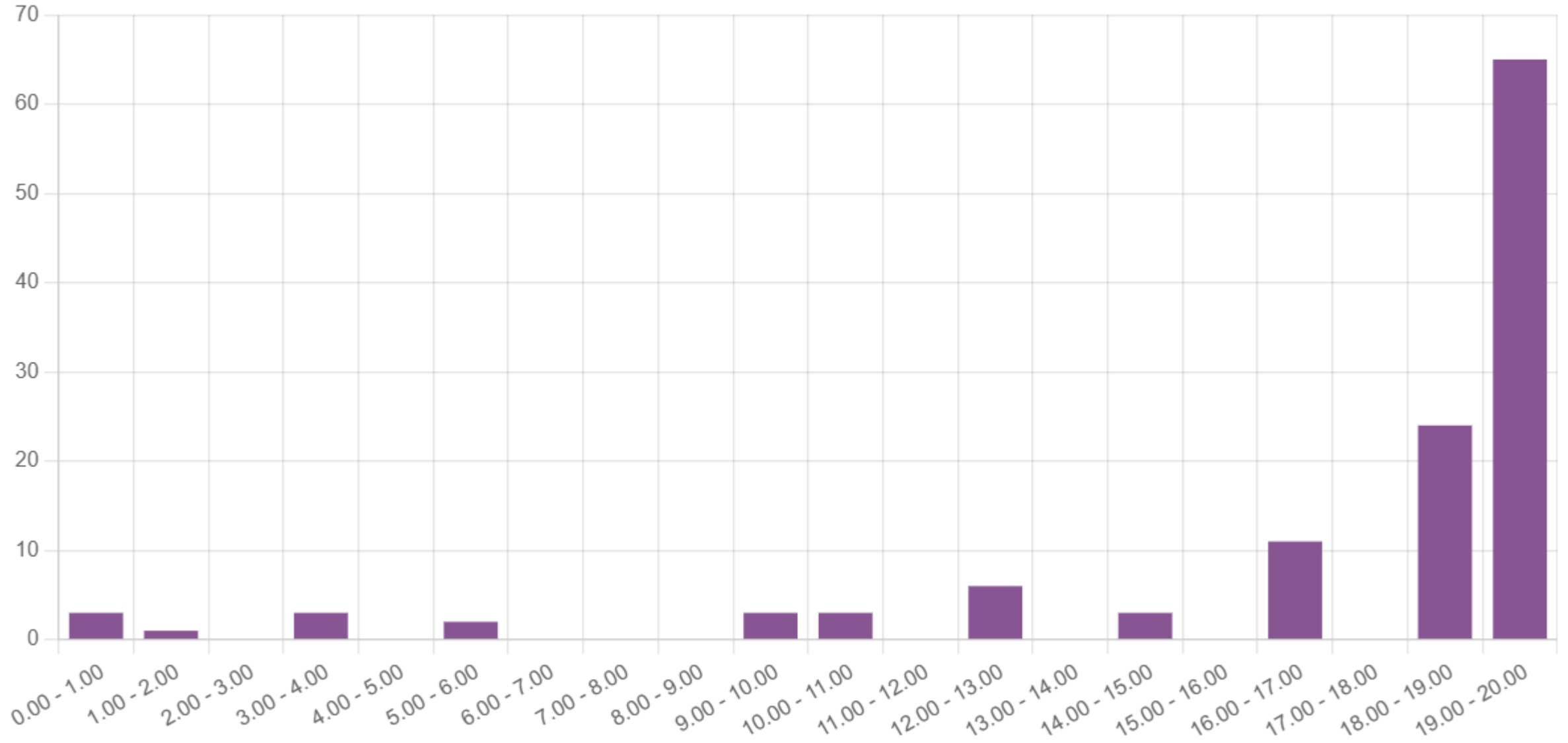


TP Biblio - Quelques retours sur
des « mauvaises pratiques »

Participants



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

On souhaite récupérer le titre et l'année d'édition des livres du catalogue.

Schéma attendu : (titre, anEd)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 SELECT titre, anEd FROM leCatalogue
```

Possible solution:

```
SELECT titre, anEd  
FROM LeCatalogue;
```

Question 2

Correct

Mark 1.00 out of 1.00

Flag question

On souhaite récupérer le numéro, le nom, le prénom et l'adresse des adhérents nés en 1960 ou après.
Schéma attendu : (noAdh, nom, prenom, adresse)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 SELECT noAdh, nom, prenom, adresse
2 FROM LesAdherents
3 WHERE anNais >= '1960';
```

On souhaite récupérer le numéro, le nom, le prénom et l'adresse des adhérents nés en 1960 ou après.
Schéma attendu : (noAdh, nom, prenom, adresse)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 SELECT noAdh, nom, prenom, adresse FROM LesAdherents WHERE anNais >= 1960;
```

Possible solution:

```
SELECT noAdh, nom, prenom, adresse
FROM LesAdherents
WHERE anNais >= 1960;
```

Question 3

Correct

Mark 1.00 out of 1.00

Flag question

On souhaite récupérer le nom et le prénom des auteurs dont la bibliothèque possède au moins un exemplaire.

Schéma attendu : (nom, prenom)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adr

```
1 SELECT DISTINCT nom, prenom
2 FROM LeCatalogue
3 WHERE nom IN (
4     SELECT nom
5     FROM LeCatalogue
6     WHERE nom = nom AND prenom = prenom
7     GROUP BY nom, prenom
8     HAVING COUNT (*) >= 1);
9
```

Possible solution:

```
SELECT DISTINCT nom, prenom
FROM LeCatalogue;
```

On souhaite récupérer le nom et le prénom des auteurs dont la bibliothèque possède au moins un exemplaire.

Schéma attendu : (nom, prenom)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 SELECT DISTINCT nom, prenom
2 FROM LeCatalogue JOIN LeFonds using(titre);
```

Question 4

Correct

Mark 1.00 out of 1.00

🚩 Flag question

On souhaite récupérer la cote et le titre des exemplaires de livres édités en 1980 ou après.

Schéma attendu : (cote, titre)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 select cote,titre from LeFonds where titre in (select titre from LeCatalogue where anEd>=1980);
```

Possible solution:

```
SELECT cote, titre  
FROM LeFonds JOIN LeCatalogue USING (titre)  
WHERE anEd >= 1980;
```

Question 5

Correct

Mark 1.00 out of 1.00

Flag question

On souhaite récupérer les numéros des adhérents qui n'ont aucun emprunt.**Schéma attendu : (noAdh)**

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)LeFonds (cote, titre)LesEmprunts (cote, noAdh, datEmp)LesAdherents (noAdh, nom, prenom, adresse,

```
1 SELECT LesAdherents.noAdh
2 FROM LesAdherents
3 MINUS
4 SELECT LesEmprunts.noAdh
5 FROM Lesemprunts
```

Possible solution:

```
SELECT noAdh
FROM LesAdherents
MINUS
SELECT noAdh
FROM LesEmprunts;
```

```
--Variant
FROM LesAdherents
WHERE noAdh NOT IN
  (SELECT noAdh
   FROM LesEmprunts);
```

On souhaite récupérer les numéros des adhérents qui n'ont aucun emprunt.**Schéma attendu : (noAdh)**

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)LeFonds (cote, titre)LesEmprunts (cote, noAdh, datEmp)LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 SELECT A.noAdh
2 FROM LesAdherents A
3 LEFT JOIN LesEmprunts E ON A.noAdh = E.noAdh
4 WHERE E.noAdh IS NULL;
```

Question 6

Correct

Mark 1.00 out of 1.00

Flag question

On souhaite récupérer les numéros des adhérents qui empruntent au moins un des exemplaires dont la cote est 2, 3, ou 8.

Schéma attendu : (noAdh)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 SELECT DISTINCT LesAdherents.noAdh
2 FROM LesAdherents
3 WHERE LesAdherents.noAdh IN (SELECT LesEmprunts.noAdh FROM LesEmprunts WHERE LesEmprunts.cote = 2 OR LesEmprunts.cote = 3 OR LesEmprunts.cote = 8 );
```

Possible solution:

```
SELECT DISTINCT noAdh
FROM LesEmprunts
WHERE cote = 2 OR cote = 3 OR cote = 8;
```


On souhaite récupérer le nom et le prénom des adhérents qui sont auteurs d'au moins un livre détenu par la bibliothèque.

Schéma attendu : (nom, prenom)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 SELECT DISTINCT LeCatalogue.nom, LeCatalogue.prenom FROM LeCatalogue
2 INTERSECT
3 SELECT LesAdherents.nom, LesAdherents.prenom FROM LesAdherents;
```

Possible solution:

```
SELECT nom, prenom
FROM LesAdherents
INTERSECT
SELECT nom, prenom
FROM LeCatalogue;
```

```
--Variant
SELECT DISTINCT nom, prenom
FROM LesAdherents JOIN LeCatalogue USING (nom, prenom);
```

Question 8

Incorrect

Mark 0.00 out of 1.00

Remove flag

On souhaite récupérer les numéros, noms et prénoms des adhérents qui n'empruntent aucun des exemplaires dont la cote est 2, 3, ou ~8.

Schéma attendu : (noAdh, nom, prenom)

Nous considérons le schéma suivant :

LeCatalogue (titre, nom, prenom, anEd)

LeFonds (cote, titre)

LesEmprunts (cote, noAdh, datEmp)

LesAdherents (noAdh, nom, prenom, adresse, anNais, datAdh)

```
1 SELECT DISTINCT noAdh,nom,prenom
2 FROM LesAdherents JOIN LesEmprunts USING(noAdh)
3 WHERE cote <> 2 OR cote <> 3 OR cote <> 8;
```

Contre-exemple

cote	noAdh
2	1
3	1
8	1
5	1

noAdh 1 dans
le résultat de la
requête... **FAUX**

Possible solution:

```
SELECT noAdh, nom, prenom
FROM LesEmprunts JOIN LesAdherents USING (noAdh)
MINUS
SELECT noAdh, nom, prenom
FROM LesEmprunts JOIN LesAdherents USING (noAdh)
WHERE cote = 2 OR cote = 3 OR cote = 8;
```

Conclusion

OPTIMISER

Eviter DISTINCT, JOINS, GROUP BY etc si on a pas besoin -> cela demande des ressources au système.

SIMPLIFIER

Eviter d'utiliser des constructions trop complexes si on peut faire plus simple

STYLISER

Attention au « style » des requêtes

- Homogène
- Sauts de ligne
- Simplicité