



Votre numéro d'anonymat :

--	--	--

**Bases de données – 2I009**

**Examen du 9 Mai 2016**

**Durée : 2 heures**

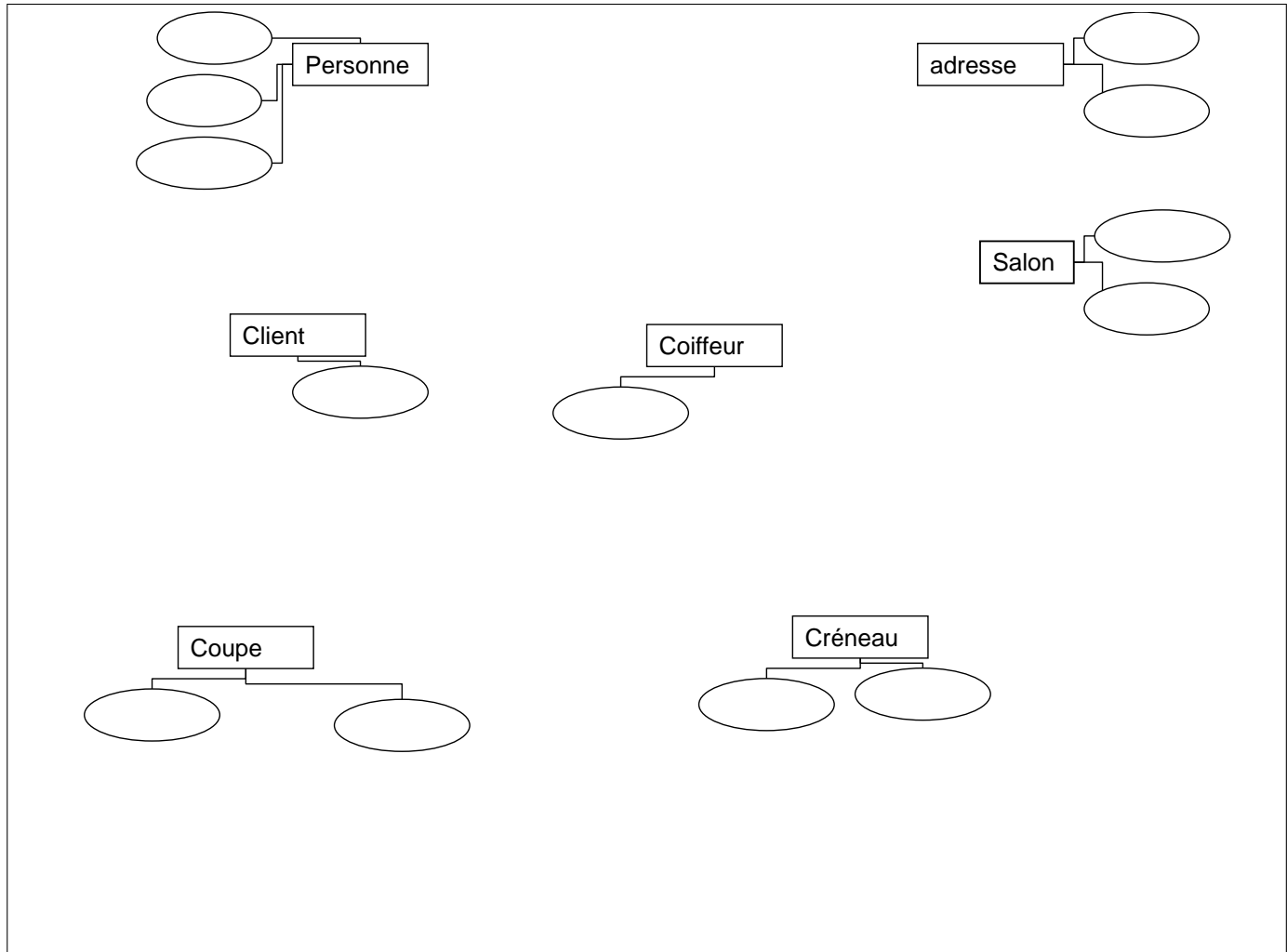
**Documents autorisés**

*Les téléphones mobiles doivent être éteints et rangés dans les sacs. Le barème sur 60 points (16 questions) n'a qu'une valeur indicative.*

**Page laissée intentionnellement vide**

# 1 Schéma Entité-Association (13 pts)

On considère une base de données qui permet de gérer des salons de coiffure avec leurs clients et qui est modélisée à l'aide d'un schéma entité-association dont toutes les entités apparaissent sur la figure suivante :



Une *Adresse* est composée d'une rue et d'une ville. Un *Créneau* correspond à un jour et une heure donnés.

Chaque *Personne* est identifiée par un idpers. On connaît son nom, son prénom et son adresse. Il y a deux types de personnes dans la base : Clients et Coiffeurs. Pour chaque *Client* on connaît la couleur de cheveux. On connaît aussi le niveau professionnel de chaque coiffeur.

Une *Coupe* est identifiée par son nom et est d'un certain type (long, court, moyen, ...).

Un *Salon* de coiffure est d'une marque (Tchip, Saint Algue, JL David, ...) et a une surface donnée. Il peut y avoir plusieurs salons d'une même marque, mais un seul salon d'une marque donnée à une adresse donnée.

## Question 1 (6 points)

Complétez le schéma Entité-Association précédent. Ne pas oublier de préciser les attributs de chaque entité ainsi que les identifiants des entités. Ajoutez les associations correspondant aux contraintes ci-dessous, en précisant leur **cardinalités** et leur éventuels **attributs** entre les entités. Rappel : les liens de généralisation/spécialisation (héritage) sont représentés par des flèches, les entités faibles par un double rectangle, leur association identifiante par un double losange.

- Une personne habite à une certaine adresse.
- Un salon est localisé à une certaine adresse.
- Un coiffeur travaille dans un seul salon.
- On stocke dans la base toutes les coupes réalisées (quel coiffeur a fait quelle coupe à quel client et quand)
- On stocke dans la base les rendez-vous pris par un client avec un coiffeur, à un certain créneau.
- Chaque client a une coupe favorite, celle qu'il préfère.

**Question 2** (2 points)

Quelle modification doit-on apporter au schéma si un coiffeur peut travailler dans plusieurs salons. Quel problème cela pose-t-il en ce qui concerne les rendez-vous ?

**Réponse :**

**Question 3** (5 points)

Compléter le cadre suivant pour traduire le schéma Entité/Association en relationnel. Pour chaque table, soulignez les clés primaires et ajoutez une '\*' aux attributs qui sont des clés étrangères.

**NB :** ne **pas** donner les commandes de création de table et les types des attributs !

**Réponse :** Personne(idpers, nom, prénom)

## 2 Requêtes (30 pts)

On considère le schéma relationnel suivant où les attributs de la clé primaire de chaque table sont soulignés et les attributs des clés étrangères sont indiqués par '\*' et portent le même nom que les attributs des clés primaires référencées :

<b>Serie</b> ( <u>titre</u> , genre, agemin, prix, distributeur, pays)	<b>Episode</b> ( <u>titre*</u> , <u>année</u> , <u>numéro</u> , resume)
<b>Acteur</b> ( <u>noact</u> , nom, prénom, naissact, pays)	<b>Rôle</b> ( <u>titre*</u> , <u>année*</u> , <u>noact*</u> , role)
<b>Client</b> ( <u>nocli</u> , nom, prénom, naisscli, pays)	<b>Abo</b> ( <u>nocli*</u> , <u>titre*</u> , datedebut, nbecrans)

La table **Serie** contient des informations sur des séries télévisées avec le titre de chaque série, son genre (comédie, action, animation, ...), l'âge minimum des spectateurs, le prix d'abonnement mensuel, le distributeur et le pays. Les séries sont diffusées pendant une ou plusieurs années (saisons) et chaque saison contient des épisodes numérotés (table **Episode**) avec un résumé par épisode. La base contient également des informations sur les acteurs (**Acteur**) et les clients (**Client**) identifiés par un numéro. Pour chaque acteur et client on connaît le nom (tout en majuscule), le prénom, la date de naissance et le pays. La table **Rôle** contient pour chaque saison le rôle joué par chaque acteur. Chaque abonnement dans la table **Abo** indique le client, la série, la date de début avec le nombre d'écrans autorisés. Un abonnement est valide pour toutes les saisons d'une série à partir de l'année de début de l'abonnement. Par exemple, un abonnement qui démarre le 1er novembre 2015 pour la série "House of Cards" donne un accès à tous les épisodes de la saison 2015 et ceux des saisons suivantes.

**Exprimez les requêtes ci-dessous dans les langages indiqués.**

**Pour SQL, ne pas utiliser les mots-clé inner join, natural join ou outer join ni d'imbrication dans le select ou le from.**

### Question 1 (4 points)

Les titres des séries distribuées par Webflux dans lesquelles l'acteur Kevin SPACEY joue un rôle pendant au moins une saison (année).

**Réponse : Calcul relationnel**

**SQL**

**Serie** (titre, genre, agemin, prix, distributeur, pays)**Episode** (titre\*, année, numéro, resume)**Acteur** (noact, nom, prénom, naissact, pays)**Role** (titre\*, année\*, noact\*, role)**Client** (nocli, nom, prénom, naisscli, pays)**Abo**(nocli\*, titre\*, datedebut, nbecrans)**Question 2** (4 points)

Les noms des acteurs qui ont joué pendant au moins deux saisons (années) dans la série “The House of Cards”.

**Réponse : Calcul relationnel****SQL****Question 3** (4 points)

Les titres des séries allemandes sans abonnés français.

**Réponse : Calcul relationnel****SQL**

**Serie** (titre, genre, agemin, prix, distributeur, pays)**Episode** (titre\*, année, numéro, resume)**Acteur** (noact, nom, prénom, naissact, pays)**Role** (titre\*, année\*, noact\*, role)**Client** (nocli, nom, prénom, naisscli, pays)**Abo**(nocli\*, titre\*, datedebut, nbecrans)**Question 4** (4 points)

Les titres des séries les plus anciennes (par rapport à l'année de leur premier épisode) ?

**Réponse : Calcul relationnel****SQL****Question 5** (2 points)

On cherche des informations statistiques sur les abonnées des séries françaises. La requête retourne le titre de la série et le nombre d'abonnés par pays. Pour chaque couple (série, pays), la requête retourne, s'il y en a, le nombre d'abonnés de ce pays à cette série. Par exemple ('Les Revenants', 'Canada', 2067) indique qu'il y a 2067 abonnés canadiens pour la série "Les Revenants".

**Réponse : SQL uniquement**

**Serie** (titre, genre, agemin, prix, distributeur, pays)**Episode** (titre\*, année, numéro, resume)**Acteur** (noact, nom, prénom, naissact, pays)**Role** (titre\*, année\*, noact\*, role)**Client** (nocli, nom, prénom, naisscli, pays)**Abo**(nocli\*, titre\*, datedebut, nbecrans)**Question 6** (4 points)

Les titres des séries du genre 'Animation' avec plus de 1000 abonnés mineurs (âge < 18 ans).

**Réponse : SQL uniquement****Question 7** (4 points)

Pour chaque série, le pays avec le plus de clients abonnés à cette série.

**Réponse : SQL uniquement**

**Serie** (titre, genre, agemin, prix, distributeur, pays)**Episode** (titre\*, année, numéro, resume)**Acteur** (noact, nom, prénom, naissact, pays)**Role** (titre\*, année\*, noact\*, role)**Client** (nocli, nom, prénom, naisscli, pays)**Abo**(nocli\*, titre\*, datedebut, nbecrans)**Question 8** (4 points)

Les noms et les prénoms des clients qui sont abonnés à toutes les séries.

**Réponse : SQL uniquement**



### 3 Création de tables, contraintes et mises à jour (11 pts)

On considère le schéma relationnel suivant où la clé primaire de chaque relation est soulignée et les attributs qui forment une clé étrangère sont suivis de \*.

**Serie** (titre, genre, prixAbont)      **Saison** (titre\*, année)      **Abonnement** (noCli\*, titre\*, pays)

#### Question 1 (5 points)

Compléter les instructions de création des tables **Serie** et **Saison** en considérant que la base accepte uniquement trois genres de séries qui sont "comédie", "action" et "animation" et qu'elle ne stocke que les saisons datant de 1990 ou plus.

**Réponse :**

```
create table serie (  
  titre varchar2(20), genre varchar2(10), prix Number,
```

.....

.....

```
create table saison (  
  titre varchar2(20), annee Number,
```

.....

.....

.....

**Serie** (titre, genre, prixAbont)**Saison** (titre\*, année)**Abonnement** (noCli\*, titre\*, pays)**Question 2** (2 points)

On voudrait retirer les séries qui n'ont pas beaucoup de succès, c'est-à-dire dont le nombre d'abonnements ne dépasse pas 10. Proposer une instruction qui permet de supprimer les nuplets de Serie en conséquence.

**Réponse :****delete from serie**

.....  
.....  
.....  
.....  
.....  
.....

**Question 3** (2 points)

On voudrait baisser de 10 % le prix d'abonnement des séries qui sont suivies par des abonnés en France. Proposer une instruction qui permet de mettre à jour la table Serie en conséquence.

**Réponse :****update Serie**

.....  
.....  
.....  
.....  
.....  
.....

**Serie** (titre, genre, prixAbont)**Saison** (titre\*, année)**Abonnement** (noCli\*, titre\*, pays)**Question 4** (2 points)

On considère la table **StatsPays** dans la quelle on voudrait insérer les statistiques d'abonnement par pays et par genre de série.

**StatsPays** (Pays, genre, NombreAbon)

Proposer une instruction qui permet d'effectuer cette insertion.

**Ne pas donner l'instruction de création la table StatsPays**

**Réponse :****insert into** StatsPays

.....  
.....  
.....  
.....  
.....  
.....

## 4 PL/SQL (6 pts)

On considère le schéma relationnel suivant et les instances valides ci-dessous.

**Serie** (titre, genre, prix)

**Saison** (titre\*, année)

**Episode** (titre\*, année\*, numéro, résumé)

**Abonnement** (noCli\*, titre\*, pays)

Serie			Saison		Abonnement		
titre	genre	prix	titre	annee	noCli	titre	pays
'lucy'	'comédie'	120	'lucy'	2001	665	'lucy'	'fr'
'benly'	'comédie'	330	'lucy'	2002	123	'yaris'	'dz'
'yaris'	'action'	300	'benly'	2010	745	'benly'	'it'

Episode			
titre	annee	numero	resume
'lucy'	2001	1	'mort de ...'
'lucy'	2001	2	'naissance ...'
'lucy'	2002	1	'depart ...' ;
'lucy'	2002	2	'arrivee ...' ;

### Question 1 (6 points)

Compléter le programme suivant qui retourne, pour chaque série, le nombre de saisons, le nombre d'abonnés et le nombre moyens d'épisode par saison. Appliqué sur la base décrite ci-dessus, le programme retourne le résultat suivant :

```
***La serie lucy
comporte 2 saison(s)
et 1 abonne(s)
et 2 episodes par saison en moyenne
***La serie benly
comporte 1 saison(s)
et 1 abonne(s)
et aucun episode
***La serie yaris
comporte 0 saison(s)
et 1 abonne(s)
et aucun episode
```

### Réponse :

```
Declare
nbSaisons number:=0;
nbAbonnes number:=0;
moyEpisode number:=0;
Begin
for s in (.....)
Loop

dbms_output.put_line('***La_serie_' || s.titre);

.....
```

```
.....  
dbms_output.put_line('comporte_' || nbSaisons || '_saison(s)');  
.....  
.....  
dbms_output.put_line('et_' || nbAbonnes || '_abonne(s)');  
.....  
.....  
.....  
  
if (..... is not null) then  
    dbms_output.put_line('et_' || moyEpisode || '_episodes_par_saison_en_moyenne');  
else  
    dbms_output.put_line('et_aucun_episode');  
end if;  
  
end loop;  
  
End;  
/
```