Département Licence	ANNÉE : 2005/2006 SESSION DE JANVIER 2006 **PARCOURS :** INF 3 **UE :** INF 109, Bases de données **Épreuve :** INF 109 EX **Date :** Vendredi 6 janvier 2006 **Heure :** 8 heures 30 **Durée :** 1 heure 30 Documents : non autorisés Épreuve de M. Alain GRIFFAULT

Code d'anonymat :

Avertissement

- La plus part des questions sont indépendantes.
- Le barème total est de 25 points car le sujet est assez long.
- Le barème est (approximativement) proportionnel à la difficulté de la question.
- L'espace laissé pour les réponses est suffisant (sauf si vous utilisez ces feuilles comme brouillon, ce qui est fortement déconseillé).

Le sujet

Le sujet porte sur une gestion simplifiée de courses et de coureurs automobiles.
Soit $GrandPrix(Licence, Nom, Dossard, Equipe, Voiture, Annee, Victoires, Courses)$ une relation et ses dépendances fonctionnelles :

- $\{Dossard\} \longrightarrow \{Equipe, Voiture\}$ qui indique que le numéro de dossard détermine l'équipe et la voiture.
- $\{Equipe\} \longrightarrow \{Voiture\}$ qui indique que les équipes ne changent jamais de voiture.
- $\{Dossard, Annee\} \longrightarrow \{Nom, Victoires, Courses\}$ qui indique que tous les ans, un dossard est attribué à un seul coureur, et que l'on enregistre son nombre de victoires et son nombre de courses.
- $\{Nom, Annee\} \longrightarrow \{Dossard\}$ qui indique que tous les ans, un coureur a un seul dossard.
- $\{Licence\} \longrightarrow \{Nom\}$ qui indique que la licence est attribuée à vie.
- $\{Nom\} \longrightarrow \{Licence\}$ qui indique qu'il n'y a pas deux coureurs ayant le même nom.

Par exemple

Licence	Nom	Dossard	Equipe	Voiture	Annee	Victoires	Courses
321	BUTTON	3	BAR	Honda	2005	0	19

spécifie que BUTTON possède la licence 321 ; qu'en 2005, avec le dossard 3 qui désigne l'équipe BAR qui possède des voitures Honda, il a participé à 19 courses sans en gagner.

Exercice 1 (Requêtes SQL sur *GrandPrix* (6 points))

Question 1.1 (1 point) *Écrire une requête SQL qui caractérise les "Equipe" ayant gagné au moins une course.*

Question 1.2 (1 point) *En supposant que "(Dossard, Annee)" soit une clef candidate, écrire une requête SQL qui caractérise les "Dossard" n'ayant pas de "Victoires" en 2003, bien qu'ayant participé cette année là.*

Question 1.3 (1 point) *En supposant que "(Dossard, Annee)" soit une clef candidate, écrire une requête SQL qui caractérise les "Dossard" ayant gagné des courses en 2000 et n'ayant pas de "Victoires" en 2003, bien qu'ayant participé cette année là.*

Question 1.4 (1 point) *Écrire une requête SQL qui caractérise les "Nom" des coureurs et le total de leurs "Victoires". Le résultat sera ordonné par ce total.*

Question 1.5 (2 points) *En supposant qu'*ALONSO *et le "Dossard" 5 soient déjà dans la base ; écrire une requête SQL qui ajoute l'information qu'en 2005,* ALONSO *avec le dossard 5 a gagné 7 courses sur les 19 qu'il a disputées.*

Exercice 2 (Normalisation (6 points))

Question 2.1 (1 point) *Donnez toutes les clefs candidates de la relation “GrandPrix”.*

Question 2.2 (1 point) *Même si l’on suppose qu’il n’y a aucun doublon dans “GrandPrix”, justifiez pourquoi la relation “GrandPrix” n’est pas en troisième forme normale.*

Question 2.3 (2 points) *Appliquez un algorithme de normalisation pour obtenir une décomposition, sans perte d’information et sans perte de dépendance fonctionnelle, de la relation “GrandPrix” en un ensemble de relations en troisième forme normale.*
Vous n’écrirez sur la copie que les nouvelles relations et les dépendances fonctionnelles qui sont à la base des projections effectuées.

Question 2.4 (1 point) *Donnez une contrainte (de bon sens) entre les attributs “Victoires” et “Courses”. Indiquez comment il est possible de la décrire en SQL.*

Question 2.5 (1 point) *Indiquez, pour vos relations, les éventuelles clefs étrangères et les conséquences en cas de suppression de tuples.*

Exercice 3 (Requêtes SQL sur votre décomposition (9 points))

Question 3.1 (1 point) *Écrire une requête SQL qui caractérise les "Equipe" ayant gagné au moins une course.*

Question 3.2 (1 point) *Écrire une requête SQL qui caractérise les "Dossard" n'ayant pas de "Victoires" en 2003, bien qu'ayant participé cette année là.*

Question 3.3 (1 point) *Écrire une requête SQL qui caractérise les "Dossard" ayant gagné des courses en 2000 et n'ayant pas de "Victoires" en 2003, bien qu'ayant participé cette année là.*

Question 3.4 (1 point) *Écrire une requête SQL qui caractérise les "Nom" des coureurs et le total de leurs "Victoires". Le résultat sera ordonné par ce total.*

Question 3.5 (1 point) *En supposant qu'*Alonso *et le "Dossard" 5 soient déjà dans la base ; Écrire une requête SQL qui ajoute l'information qu'en 2005,* Alonso *avec le dossard 5 a gagné 7 courses sur les 19 qu'il a disputées.*

Question 3.6 (2 points) *Après en avoir donné une écriture algébrique, écrire une requête SQL qui caractérise les "Nom" des coureurs ayant couru au cours de leur vie avec toutes les "Voiture".*

Question 3.7 (2 points) *Après en avoir donné une écriture prédicative, écrire une requête SQL qui caractérise les "Dossard" ayant gagné au moins une course toutes les années.*

Exercice 4 (Transactions)

Question 4.1 (2 points) *Imaginez une suite de requêtes, telles que si elles ne sont pas mises dans une transaction, la sélection de la question 3.3 ne donne pas un résultat correct.*

Exercice 5 (Subsidiaire)

Question 5.1 (2 points) *Expliquez en quelques lignes pourquoi la requête suivante, pour n'importe quelle base de données ayant un attribut "DateDeNaissance", caractérise toujours un ensemble vide.*

```
-- Les personnes nés dans la nuit du 3 au 13 septembre 1752.
SELECT   *
FROM     MaTable
WHERE    DateDeNaissance BETWEEN '1752-9-4' AND '1752-9-12';
```