

## TD Base de données - Extraits d'épreuves écrites

### 1 Extrait CCINP PSI 2024

Le sujet traitait du jeu de l'Awalé, jeu qui se joue à deux joueurs.

On dispose d'une base de données pour stocker l'historique des parties jouées.

Joueur				
id_Joueur	nom	prenom	niveau	naissance
18571	Martin	Jean	2048	23/02/1958
18572	Dupond	Marie	2103	03/01/1972
18573	Develion	Théo	1857	05/10/2004

Partie					
id_Partie	id_joueur1	id_joueur2	resultat	jour	jeu
1	1547	1568	0.5	08/01/2001	'egai...'
2	1204	3	0	12/07/1998	'egaj...'
3	4	2	1	15/07/2018	'egbi...'

La table Joueur contient les attributs suivants :

- id\_Joueur : identifiant d'un joueur (entier, clé primaire);
- nom : nom du joueur (chaîne de caractères);
- prenom : prénom du joueur (chaîne de caractères);
- niveau : niveau maximal atteint par le joueur au cours de sa carrière (entier);
- naissance : date de naissance du joueur (date).

La table Partie contient les attributs suivants :

- id\_Partie : identifiant de la partie (entier, clé primaire);
- id\_joueur1 : identifiant du joueur débutant la partie (entier);
- id\_joueur2 : identifiant du second joueur (entier);
- resultat : 1 est une victoire du joueur1, 0.5 une égalité et 0 une victoire du joueur2 (flottant);
- jour : date du jour de la partie (date);
- jeu : liste des coups successifs de la partie, stockée sous forme d'une chaîne de caractères. 'egai...' signifie que le joueur1 a joué la 5<sup>e</sup> case (d'indice 4) représentée par la lettre 'e', puis le joueur2 a joué la 7<sup>e</sup> case (d'indice 6) représentée par la lettre 'g', puis le joueur1 a joué la 1<sup>re</sup> case et ainsi de suite.

*Remarque : L'opérateur LIKE est utilisé pour comparer des chaînes de caractères dans la clause WHERE des requêtes SQL. Ce mot-clé permet d'effectuer une recherche sur un modèle particulier. Il est par exemple possible de rechercher les enregistrements dont la valeur d'une colonne commence par telle ou telle lettre. Le caractère \_ (underscore) représente n'importe quel caractère, mais un seul caractère uniquement alors que le caractère pourcentage % peut être remplacé par un nombre quelconque (et possiblement nul) de caractères. Par exemple parmi une recherche dans les communes de France, nom LIKE '\_ff%f%' ne renvoie que Offendorf alors que remplacer le \_ par un % renvoie Pfaffenhoffen et Staffelfelden en plus de Offendorf.*

**Q1.** Écrire une requête SQL permettant d'extraire les identifiants des joueurs ayant un niveau strictement supérieur au score 1900.

**Q2.(★)** Écrire une requête SQL permettant de déterminer le pourcentage de victoires du joueur1 pour les parties où la case d'indice 0 a été jouée en premier.

**Q3.** Écrire une requête SQL permettant d'afficher le nom et le prénom des 3 joueurs ayant le niveau le plus élevé.

**Q4. (★)** Écrire une requête SQL permettant de déterminer les joueurs ayant plus de cent victoires lorsqu'ils commencent la partie. La requête doit renvoyer le nom, le prénom et le nombre de victoires de ces joueurs classés par ordre décroissant du nombre de victoires.

## 2 Extrait Mines-Ponts PSI 2024

Il s'agissait de répertoire des caractères numérisés.

La numérisation de larges corpus littéraires a permis la création de bases de données contenant des informations relatives au nombre d'occurrences des différents caractères alphanumériques dans diverses langues. Nous allons supposer disposer d'une base de données constituée de trois tables :

- caractere(idCar, symbole, typeCaractere, codeHTML) ;
- corpus(idLivre, titre, auteur, annee, nombreCaracteres, langue) ;
- occurrences(idCar, idLivre, nombreOccurrences) ;

dont voici quelques extraits : Table **caractere** :

idCar	symbole	typeCaractere	codeHTML
65	'A'	'lettre'	'&#65;'
48	'0'	'nombre'	' &#48;'

Table **corpus** :

idLivre	titre	auteur	annee	nombreCaracteres	langue
1	'Germinal'	'Zola'	1885	1152365	'Français'
2	'Les Misérables'	'Hugo'	1862	2245300	'Fraçais'

Table **occurrences** :

idCar	idLivre	nombreOccurrences
62	31	155
37	21	1550

**Q1** - Écrire en langage SQL une requête permettant d'obtenir la liste sans doublon des auteurs présents dans la table **corpus**.

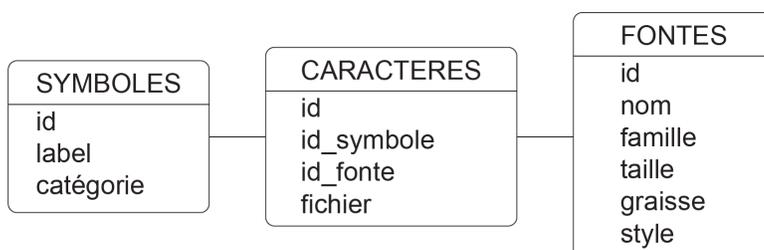
**Q2** - (★★) Écrire une requête SQL permettant d'obtenir la fréquence d'occurrences (donc comprise entre 0 et 1) de chaque caractère en 'Français'. Plus précisément, la requête devra renvoyer une table à deux attributs : une colonne contenant le symbole de chaque caractère, et une autre colonne contenant le rapport entre le nombre total d'occurrences du caractère et le nombre total de caractères des textes de tout le corpus.

## 3 Extrait CCINP PSI 2023

Le sujet traitait de la reconnaissance des caractères dans un document.

Une base de données contient des informations sur chaque caractère selon le type de fonte, la taille, la graisse...

Trois tables sont utilisées :



La table SYMBOLES contient les attributs suivants :

- id : identifiant d'un symbole (entier), clé primaire ;
- label : nom du symbole ("A", "a", "1", "é", "!"...) (chaîne de caractères) ;
- catégorie : parmi majuscule, minuscule, chiffre, spécial (dont accent) (chaîne de caractères).

La table CARACTERES contient les attributs suivants :

- id : identifiant d'un caractère (entier), clé primaire ;
- id\_symbole : identifiant du nom du symbole (entier) ;
- id\_fonte : identifiant du type de fonte (entier) ;
- fichier : nom du fichier image correspondant (chaîne de caractères).

La table FONTES contient les attributs suivants :

- id : identifiant d'une fonte (entier), clé primaire ;
- nom : nom de la fonte ("Arial", "Times new roman", "Calibri", "Zurich", ...) (chaîne de caractères) ;
- famille : nom de la famille dont fait partie la fonte ("humane", "garalde", "réale", "didone", "scripte", ...) (chaîne de caractères) ;
- taille : dimension en hauteur des caractères en pixels (entier) ;
- graisse : type de graisse ("léger", "normal", "gras", "noir", ...) (chaîne de caractères) ;
- style : type de style ("romain", "italique", "ombré", "décoratif", ...) (chaîne de caractères).

**Q1.** Écrire une requête SQL permettant d'extraire les identifiants des fontes dont le nom est "Zurich", de style "romain" et dont la taille est comprise entre 10 et 16 pixels.

**Q2.** Écrire une requête SQL permettant d'extraire tous les noms de fichiers des caractères qui correspondent au symbole de label "A".

**Q3.** (★) Écrire une requête SQL permettant d'indiquer le nombre de caractères correspondant à la fonte "Zurich", de style "romain" et dont la taille est comprise entre 10 et 16 pixels groupés selon les labels des symboles.

## 4 Extrait Mines-Ponts PSI 2023

*Le sujet traitait de la typographie informatisée.*

Une base de données stocke les informations liées aux polices de caractères dans 4 tables ou relations. Famille décrit les familles de polices, avec fid la clé primaire entière et fnom leur nom. Police décrit les polices de caractères disponibles, avec pid la clé primaire entière, pnom le nom de la police et fid de numéro de sa famille.

Caractere décrit les caractères, avec code la clé primaire entière, car le caractère lui-même, cnom le nom du caractère.

Glyphe décrit les glyphes disponibles, avec gid la clé primaire entière, code le code du caractère correspondant au glyphe, pid le numéro de la police à laquelle le glyphe appartient, groman un booléen vrai pour du roman et faux pour de l'italique et gdesc la description vectorielle du glyphe.

Voici un extrait du contenu de ces tables :

Famille		Police			Caractere		
fid	fnom	pid	pnom	fid	code	car	cnom
1	Humane	1	Centaur	1	65	A	lettre majuscule latine a
2	Garalde	2	Garamond	2	66	B	lettre majuscule latine b
3	Réale	3	Times New Roman	3	...	...	...
4	Didone	4	Computer Modern	4	97	a	lettre minuscule latine a
5	Mécane	...	...	...	98	b	lettre minuscule latine b
6	Linéale	21	Triangle	6	99	c	lettre minuscule latine c
...	...	...	...	...	...	...	...

Glyphe				
gid	code	pid	groman	gdesc
1	65	20	True	[[[0,0], [1,2], [2,0]], [[0.5,1], [1.5,1]]]
2	65	20	False	[[[0,0], [2,2], [2,0]], [[1,1], [2,1]]]
...	...	...	...	...
501	97	21	True	[[[0,0], [0.5,1], [1,0], [0,0]]]
502	98	21	True	[[[0,2], [0,0], [1,0.5], [0,1]]]
...	...	...	...	...

**Q1.** Proposer une requête en SQL sur cette base de données pour compter le nombre de glyphes en roman (cf. description précédente).

**Q2. (★)** Proposer une requête en SQL afin d'extraire la description vectorielle du caractère *A* dans la police nommée Helvetica en italique.

**Q3. (★)** Proposer une requête en SQL pour extraire les noms des familles qui disposent de polices et leur nombre de polices, classés par ordre alphabétique.

## 5 Mines-Ponts PSI 2022

Il existe des bases de données contenant les propriétés de nombreux matériaux, dont des propriétés magnétiques. Dans cette partie, on donne un modèle simplifié d'une telle base, et on souhaite effectuer quelques requêtes sur celle-ci. La base de données possède la structure suivante :

- La table **matériaux** contient un champ **id\_materiau**, clé primaire de la table de valeur entière, un champ **nom** de type chaîne de caractères pour le nom du matériau et un champ **t\_curie** de valeur entière pour la température de Curie du matériau en kelvin.

id_materiau	nom	t_curie
4534	cobalt	1 388
1254	dioxyde de chrome	386
8713	nickel	627
8284	YIG	560
...	...	...

- La table **fournisseurs**, contenant un champ **id\_fournisseur**, clé primaire de type entier qui précise le code de chaque fournisseur, et un champ **nom\_fournisseur** de type chaîne de caractères pour le nom du fournisseur.

id_fournisseur	nom_fournisseur
145	Worldwide Materials
13	Materials Company
...	...

- La table prix qui contient un champ `id_prix`, clef primaire de type entier, un champ `id_mat` dont les valeurs sont incluses dans l'ensemble des valeurs de la clé `id_materiau` de la table `materiaux`, un champ `id_four` dont les valeurs sont incluses dans l'ensemble des valeurs de la clé `id_fournisseur` de la table `fournisseurs`, et un champ `prix_kg` de type flottant qui précise le prix au kg que ce fournisseur propose pour ce matériau, en euros. Un fournisseur qui ne propose pas un matériau donné n'a pas d'entrée correspondante dans cette table.

id_prix	id_mat	id_four	prix_kg
1	4567	145	50.40
2	8671	13	1357.30
3	1763	145	52.75
...	...	...	...

Les requêtes demandées dans cette partie sont à écrire en langage SQL.

**Q1.** Écrire une requête permettant d'obtenir le nom de tous les matériaux qui ont une température de Curie strictement inférieure à 500 kelvins.

Un client potentiel souhaite acheter 4,5 kilogrammes de nickel (d'identifiant 8713, que l'on pourra utiliser directement dans les requêtes) et sélectionner le fournisseur le moins cher.

**Q2.** Écrire une requête permettant d'obtenir les noms de tous les fournisseurs proposant du nickel et le prix proposé par chacun pour 4,5 kilogrammes de nickel.

**Q3.** (★★) Modifier ou compléter la requête précédente afin d'obtenir le nom du fournisseur de nickel le moins cher ainsi que le prix à payer chez ce fournisseur pour ces 4,5 kilogrammes de nickel. En cas d'égalité du prix optimal entre plusieurs fournisseurs, on obtiendra les noms de tous les fournisseurs possibles.

**Q4.** (★) Écrire une requête permettant d'obtenir le nom de tous les matériaux et le prix moyen pour un kilogramme de chacun de ces matériaux (la moyenne étant calculée pour tous les fournisseurs proposant ce matériau), en se limitant aux prix moyens strictement inférieurs à 50 euros par kilogramme.

## 6 CCINP PSI 2021

Les fabricants de montres connectées offrent la possibilité d'enregistrer les activités dans une base de données afin qu'elles soient accessibles sur PC ou tablettes et qu'elles puissent être partagées avec des amis.

La base de données est constituée des tables suivantes :

Table **activite** :

- `Ida` : entier permettant d'identifier l'activité
- `Idm` : entier correspondant à l'identifiant du membre "propriétaire" de l'activité
- `Date` : correspondant à la date (type date) de l'activité
- `Type` : chaîne de caractères correspondant au type d'activité "course", "marche",
- `Distance` : entier correspondant à la distance parcourue en mètre de l'activité
- `Temps` : entier correspondant à la durée en secondes de l'activité
- `Fichier` : contient le lien vers le fichier de l'activité

Table **amis** qui établit une relation entre 2 membres :

- `Idl` : entier, identifiant du lien
- `Membre1` : `Idm` d'un membre
- `Membre2` : `Idm` d'un second membre

Il ne peut pas y avoir de doublons dans la table amis : si A est le membre 1 et B le membre 2 d'une relation d'amitiés, la ligne membre 1=B et membre 2=A n'existe pas dans la table.

On considère pour les questions suivantes un membre dont l'identifiant est 1 ( $Id_m = 1$ ).

**Q1.** Écrire une requête SQL permettant de récupérer la liste des identifiants des activités du membre d'identifiant 1.

**Q2.** Écrire une requête SQL permettant de donner la date, la distance parcourue et la vitesse moyenne en km/h des activités de type "course" du membre dont l'identifiant est 1.

On donne la requête suivante :

```

1 SELECT activite.Ida FROM activite
2 JOIN (SELECT membre1 AS idam1 FROM amis WHERE membre2 = 1
3     UNION
4     SELECT membre2 AS idam1 FROM amis WHERE membre1 = 1) AS amis1
5 ON activite.idm = amis1.idam1
6 WHERE Type = 'marche'
```

**Q3. (★)** Décrire chaque instruction de la requête et expliquer ce qu'elle renvoie.

## 7 Extrait Mines-Ponts PSI 2021

Lors de la préparation d'une randonnée, une accompagnatrice doit prendre en compte les exigences des participants. Elle dispose d'informations rassemblées dans deux tables d'une base de données :

- la table **Rando** décrit les randonnées possibles : la clef primaire entière **rid**, son nom, le niveau de difficulté du parcours (entier entre 1 et 5), le dénivelé (en mètres), la durée moyenne (en minutes) :

rid	rnom	diff	deniv	duree
1	La belle des champs	1	20	30
2	Lac de Castellane	4	650	150
3	Le tour du mont	2	200	120
4	Les crêtes de la mort	5	1200	360
5	Yukon Ho!	3	700	210
...	...	...	...	...

- la table **Participant** décrit les randonneurs : la clef primaire entière **pid**, le nom du randonneur, son année de naissance, le niveau de difficulté maximum de ses randonnées :

pid	pnom	ne	diff_max
1	Calvin	2014	2
2	Hobbes	2015	2
3	Susie	2014	2
4	Rosalyn	2001	4
...	...	...	...

Donner une requête SQL sur cette base pour :

**Q1.** Compter le nombre de participants nés entre 1999 et 2003 inclus.

**Q2.** Calculer la durée moyenne des randonnées pour chaque niveau de difficulté.

**Q3. (★)** Extraire le nom des participants pour lesquels la randonnée n°42 est trop difficile.

**Q4. (★)** Extraire les clés primaires des randonnées qui ont un ou des homonymes (nom identique et clé primaire distincte), sans redondance.

## 8 Extrait CCINP PSI 2019

Une base de données médicale contient des informations administratives sur les patients et des informations médicales. Pour simplifier le problème, on considère deux tables : PATIENT et MEDICAL.

La table PATIENT contient les attributs suivants :

- id : identifiant d'un individu (entier), clé primaire ;
- nom : nom du patient (chaîne de caractères) ;
- prenom : prénom du patient (chaîne de caractères) ;
- adresse : adresse du patient (chaîne de caractères) ;
- email : (chaîne de caractères) ;
- naissance : année de naissance (entier).

La table MEDICAL contient les attributs suivants :

- id : identifiant d'un ensemble de propriétés médicales (entier), clé primaire ;
- data1 : donnée (flottant) ;
- data2 : donnée (flottant) ;
- ...
- idpatient : identifiant du patient représenté par l'attribut id de la table PATIENT (entier) ;
- etat : description de l'état du patient (chaîne de caractères).

Les attributs data1, data2... sont des données relatives à l'analyse médicale souhaitée (dans notre cas des données biomécaniques). L'attribut «etat » permet d'affecter un label à un ensemble de données médicales : «normal », «hernie discale », « spondylolisthésis ».

- Q1.** Écrire une requête SQL permettant d'extraire les identifiants des patients ayant une «hernie discale ».
- Q2.** Écrire une requête SQL permettant d'extraire les noms et prénoms des patients atteints de "spondylolisthésis".
- Q3.** Écrire une requête SQL permettant d'extraire chaque état et le nombre de patients pour chaque état.

# Base de données - Compléments sur les relations entre tables

## 9 Relation 1 à 1, notée 1-1

Deux tables A et B sont dites en relation 1-1, lorsque **chaque tuple (= une ligne) de la table A se rapporte à un unique tuple de la table B**, et que **chaque tuple de la table B se rapporte à un unique tuple de la table A**.

Par exemple, si on dispose d'une table "Ministres" qui donne les renseignements sur tous les ministres en activité, et d'une table "Ministères" qui donne les renseignements sur chaque ministère, alors la table "Ministères" est en relation 1 :1 avec la table "Ministres", car chaque ministre est à la tête d'un seul ministère et un ministère ne comporte qu'un seul ministre.

## 10 Relation 1 à plusieurs, notée 1-\*

Deux tables A et B sont dites en relation 1-\* lorsque **chaque tuple de la table A se rapporte à un ou plusieurs tuples de la table B**, et que **chaque tuple de la table B se rapporte à un unique tuple de la table A**

Par exemple, dans la BDD "Vols" étudiée en cours, la table "Pilote" est en relation 1-\* avec la table "Vol" car chaque ligne de la table "Pilote" se rapporte à une ou plusieurs lignes de la table "Vol" (un pilote peut faire plusieurs vols, mais un vol ne concerne qu'un seul pilote)

Ce type de relation va pouvoir se faire grâce à une clé étrangère : la clé étrangère d'une table B permet de renvoyer à une clé primaire de la table A.

## 11 Relation plusieurs à plusieurs, notée \*-\*

Deux tables A et B sont dites en relation \*-\* lorsque **un tuple de la table A se rapporte à un ou plusieurs tuples de la table B** et **un tuple de la table B se rapporte à un ou plusieurs tuples de la table A**.

Par exemple, dans dans la BDD "Vols" , la table "Avion" est en relation \*-\* avec la table "Pilote", car un avion va pouvoir être piloté par différents pilotes, et un pilote va pouvoir piloter plusieurs avions.

*Remarque* : une relation \*-\* peut toujours être décomposée en deux relations 1-\*, grâce à une troisième table (appelée table d'association) qui comportera deux clés étrangères faisant référence chacune à une clé primaire de la table A et de la table B. On aura donc besoin de faire une jointure entre trois tables.

PILOTE					AVION			
PLNUM	PLNOM	PLPRENOM	VILLE	SALAIRE	AVNUM	AVNOM	CAPACITE	LOCALISATION
1	MIRANDA	SERGE	PARIS	21000	1	A300	300	NICE
2	LETHANH	NAHN	TOULOUSE	21000	2	A310	300	NICE
3	TALADOIRE	GILLES	NICE	18000	3	B707	250	PARIS
...	...	...	...	...	...	...	...	...

VOL						
VOLNUM	PLNUM	AVNUM	VILLEDEP	VILLEARR	HEUREDEP	HEUREARR
100	1	1	NICE	TOULOUSE	11:00	12:30
101	1	8	PARIS	TOULOUSE	17:00	18:30
102	2	1	TOULOUSE	LYON	14:00	16:00
...	...	...	...	...	...	...

*A vous* : Ecrire une requête donnant le nombre de fois où le pilote nommé Serge Miranda a piloté un A310.