

Le barème est donné à titre indicatif.

L'utilisation des outils numériques (ordinateur, téléphone, montre connecté, etc.) est interdite.

Exercice 1 : 5 points

Soit le code JSON pour la collection restaurant :

```
{
  "_id": {
    "$oid": "55f14312c7447c3da7051b26"
  },
  "URL": "http://www.just-eat.co.uk/restaurants-cn-chinese-cardiff/menu",
  "address": "228 City Road",
  "address line 2": "Cardiff",
  "name": ".CN Chinese",
  "outcode": "CF24",
  "postcode": "3JH",
  "rating": 5,
  "type_of_food": "Chinese"
},
{
  "_id": {
    "$oid": "55f14312c7447c3da7051b27"
  },
  "URL": "http://www.just-eat.co.uk/restaurants-atthai-ss9/menu",
  "address": "376 Rayleigh Road",
  "address line 2": "Essex",
  "name": "@ Thai",
  "outcode": "SS9",
  "postcode": "5PT",
  "rating": 5.5,
  "type_of_food": "Thai"
},
{
  "_id": {
    "$oid": "55f14312c7447c3da7051b28"
  },
  "URL": "http://www.just-eat.co.uk/restaurants-atthairestaurant/menu",
  "address": "30 Greyhound Road Hammersmith",
  "address line 2": "London",
  "name": "@ Thai Restaurant",
  "outcode": "W6",
  "postcode": "8NX",
  "rating": 4.5,
  "type_of_food": "Thai"
}
```

```
},
{
  "_id": {
    "$oid": "55f14312c7447c3da7051b29"
  },
  "URL": "http://www.just-eat.co.uk/restaurants-atthairestaurant/menu",
  "address": "30 Greyhound Road Hammersmith",
  "address line 2": "London",
  "name": "@ Thai Restaurant",
  "outcode": "W6",
  "postcode": "8NX",
  "rating": 4.5,
  "type_of_food": "Thai"
},
{
  "_id": {
    "$oid": "55f14312c7447c3da7051b2a"
  },
  "URL": "http://www.just-eat.co.uk/restaurants-indiancom-ch4/menu",
  "address": "9 Broughton Hall Road",
  "address line 2": "Broughton",
  "name": "@Indian.com",
  "outcode": "CH4",
  "postcode": "0QR",
  "rating": 6,
  "type_of_food": "Curry"
},
{
  "_id": {
    "$oid": "55f14312c7447c3da7051b2b"
  },
  "URL": "http://www.just-eat.co.uk/restaurants-007takeaway-s65/menu",
  "address": "6 Drummond Street",
  "address line 2": "Rotherham",
  "name": "007 Takeaway",
  "outcode": "S65",
  "postcode": "1HY",
  "rating": 6,
  "type_of_food": "Pizza"
},
{
  "_id": {
    "$oid": "55f14312c7447c3da7051b2c"
  },
  "URL": "http://www.just-eat.co.uk/restaurants-042-restaurant-e11/menu",
  "address": "885 High Road Leytonstone",
```

```
"address line 2": "London",  
"name": "042 Restaurant & Bar",  
"outcode": "E11",  
"postcode": "1HR",  
"rating": 3  
}]
```

Ecrire les requêtes permettant la :

1. Recherche des restaurants avec un rating inférieur ou égal à 5 et situés à Londres
2. Mise à jour de l'adresse du restaurant "@ Thai Restaurant" à "40 New Road Hammersmith"
3. Ajout d'un champ "delivery" à tous les restaurants et initialisation à true
4. Recherche des restaurants dont l'adresse contient "Road" et trier par rating décroissant
5. Suppression des restaurants qui n'ont pas de type de nourriture défini
6. Recherche des restaurants avec un type de nourriture non "Pizza" et un rating entre 3 et 6 (inclus)

Exercice 2 : 4 points

L'université dispose d'un service de reprographie qui permet aux étudiants et aux enseignants de faire reproduire divers documents. Proposer un MCD et un MLD respectant les exigences suivantes, en identifiant les clés primaires et les clés étrangères :

Un utilisateur est soit un étudiant, soit un enseignant. Il possède également les caractéristiques suivantes : nom, prénom, numéro de téléphone et département.

Chaque document reproduit a un titre, une description, un type (thèse, article, cours, etc.) et une date de dépôt. Les documents peuvent être en format papier ou électronique.

Une demande de reproduction est faite par un utilisateur pour un ou plusieurs documents. La demande est caractérisée par la date, l'état de la demande (en attente, en cours, terminée), et le nombre de copies à réaliser. Chaque demande est associée à une ou plusieurs machines de reprographie.

Une machine a un modèle, une marque, et une capacité (nombre de copies par minute). Il est nécessaire de connaître l'état de chaque machine (en service, en panne, en maintenance).

Chaque demande de reproduction est conservée dans un historique des demandes, avec la date et le nombre de copies réalisées.

Exercice 3 : 6 points

Soit le fichier XML suivant :

```
<?xml version="1.0" encoding="UTF-8"?>
<ordinateurs>
  <ordinateur>
    <marque>Apple</marque>
    <modele>MacBook Pro</modele>
    <processeur>Intel Core i9</processeur>
    <ram>16GB</ram>
    <prix>2399</prix>
  </ordinateur>
  <ordinateur>
    <marque>Dell</marque>
    <modele>XPS 13</modele>
    <processeur>Intel Core i7</processeur>
    <ram>8GB</ram>
    <prix>999</prix>
  </ordinateur>
  <ordinateur>
    <marque>HP</marque>
    <modele>Spectre x360</modele>
    <processeur>Intel Core i7</processeur>
    <ram>16GB</ram>
    <prix>1349</prix>
  </ordinateur>
  <ordinateur>
    <marque>Lenovo</marque>
    <modele>ThinkPad X1 Carbon</modele>
    <processeur>Intel Core i5</processeur>
    <ram>8GB</ram>
    <prix>1299</prix>
  </ordinateur>
  <ordinateur>
    <marque>Asus</marque>
    <modele>ZenBook 14</modele>
    <processeur>AMD Ryzen 7</processeur>
    <ram>16GB</ram>
    <prix>1099</prix>
  </ordinateur>
</ordinateurs>
```

1. Ecrire la requête XPath pour extraire toutes les informations sur les ordinateurs Asus
2. Ecrire la requête XSLT pour extraire le modèle des ordinateurs avec le processeur AMD Ryzen 7
3. Ecrire une requête XPath pour compter le nombre des ordinateurs qui ont un prix < 1300
4. Ecrire une requête XQuery pour calculer la moyenne des prix des ordinateurs

5. Ecrire une requête XQuery pour extraire toutes les informations sur les ordinateurs et ajoute une nouvelle balise <info> qui combine le modèle, le processeur et la RAM dans une seule chaîne.
Par exemple : <info>MacBook Pro - Intel Core i9 - 16GB</info>
6. Ecrire une requête XQuery pour extraire tous les modèles d'ordinateurs qui coûtent plus de 1000

Exercice 4 : 5 points

Soit la base de données suivante pour la gestion de tickets de cinéma.

client_id [PK] integer	nom character varying (50)	prenom character varying (50)	email character varying (100)	date_naissance date
1	Dupont	Jean	jean.dupont@example.com	1985-02-15
2	Leroy	Marie	marie.leroy@example.com	1990-06-23
3	Martin	Paul	paul.martin@example.com	1978-11-02
4	Bernard	Lucie	lucie.bernard@example.com	2000-04-12
5	Petit	Sophie	sophie.petit@example.com	1995-08-19

film_id [PK] integer	titre character varying (100)	genre character varying (50)	duree integer	date_sortie date
1	Inception	Science-Fiction	148	2010-07-16
2	Avatar	Action	162	2009-12-18
3	Titanic	Romance	195	1997-12-19
4	The Dark Knight	Action	152	2008-07-18
5	La La Land	Comédie musicale	128	2016-12-09

seance_id [PK] integer	film_id integer	date_heure timestamp without time zone	salle character varying (50)	places_totales integer
1	1	2024-06-18 19:00:00	Salle 1	200
2	2	2024-06-18 21:00:00	Salle 2	250
3	1	2024-06-19 18:00:00	Salle 1	200
4	4	2024-06-19 20:00:00	Salle 3	180
5	5	2024-06-20 17:00:00	Salle 2	250

ticket_id [PK] integer	client_id integer	seance_id integer	date_achat timestamp without time zone	prix numeric (5,2)
1	1	1	2024-06-17 15:30:00	12.50
2	2	2	2024-06-17 16:00:00	14.00
3	1	3	2024-06-17 17:45:00	13.50
4	1	4	2024-06-17 18:30:00	12.00
5	5	5	2024-06-17 19:00:00	15.00

- Sélectionner les 2 Films avec le plus grand nombre de séances programmées.
- Classer les films par revenu total. Résultat attendu :

titre character varying (100)	revenu_total numeric	classement bigint
Inception	26.00	1
La La Land	15.00	2
Avatar	14.00	3
The Dark Knight	12.00	4

- Afficher le nombre de tickets vendus par mois et par film. Résultat attendu :

mois double precision	titre character varying (100)	total_tickets bigint
6	Avatar	1
6	Inception	2
6	La La Land	1
6	The Dark Knight	1
6	[null]	5
[null]	[null]	5

- Afficher les clients ayant acheté des tickets pour le même film plusieurs fois. Résultat attendu :

nom character varying (50)	prenom character varying (50)	titre character varying (100)	nombre_tickets bigint
Dupont	Jean	Inception	2

Bonne chance