

Sujet d'examen pratique Java

Objectif : développer une application Java selon un thème libre (logiciel de gestion, jeu vidéo, etc.) tout en respectant les contraintes techniques suivantes.

1. Contraintes techniques :

a. Utilisation de JDBC pour la base de données :

- Implémenter des opérations CRUD (Create, Read, Update, Delete) sur une base de données.
- Les informations de connexion (login et mot de passe) doivent être stockées dans un fichier « config.ini » que l'application lira pour se connecter à la base de données.

b. Fichier « config.ini » :

- Créer un fichier « config.ini » contenant les informations de connexion à la base de données.
- Exemple de format :

```
[database]
url=jdbc:mysql://localhost:3306/votrebaze
username=votreutilisateur
password=votremotdepasse
```

c. Collections et tri personnalisé :

- Créer une collection d'objets (par exemple, une liste de produits, d'employés, de scores, etc.).
- Implémenter un tri personnalisé sur cette collection (par exemple, trier les produits par prix, les employés par nom, les scores par points, etc.).

d. Polymorphisme et héritage :

- Utiliser au moins une hiérarchie de classes qui démontre l'héritage et le polymorphisme.
- Exemple : une classe « Personne » avec des sous-classes « Employe » et « Client » qui héritent de « Personne ».

e. Utilisation de threads :

- Implémenter une fonctionnalité de votre application qui utilise des threads.
- Exemple : un thread pour mettre à jour l'interface utilisateur régulièrement, ou pour exécuter des tâches en arrière-plan comme la sauvegarde automatique.

f. Utilisation de sockets :

- Implémenter une fonctionnalité réseau utilisant des sockets.
- Exemple : communication entre deux instances de votre application pour échanger des messages ou des données en temps réel.

g. Utilisation d'exceptions personnalisées :

- Créer et utiliser au moins une exception personnalisée.
- Exemple : une exception « *InvalidUserInputException* » pour gérer des erreurs spécifiques aux entrées utilisateurs incorrectes.

h. Interface graphique :

- L'application doit être en mode fenêtré.
- Utilisez soit Swing soit JavaFX pour créer l'interface graphique.

2. Éléments évalués :

a. Architecture de l'application :

- Respect des principes de conception orientée objet.
- Séparation claire des responsabilités (modèle, vue, contrôleur, ...).

b. Fonctionnalités implémentées :

- Complétude et adéquation par rapport aux contraintes techniques.
- Qualité et robustesse du code.

c. Interface utilisateur :

- Utilisabilité et esthétique de l'interface graphique.
- Réactivité et gestion des événements utilisateurs.

d. Documentation et lisibilité du code :

- Commentaires et documentation interne du code.
- Utilisation de noms de variables et de méthodes explicites.

e. Test et validation :

- Gestion des erreurs et des exceptions.

3. Livrables :

Code source complet de l'application (zip ou rar).

Fichier `config.ini` avec les informations de connexion à la base de données.

Rapport décrivant l'architecture de l'application, les choix de conception, et les fonctionnalités implémentées.

4. Déroulement de l'examen :

- Vous aurez une période de 21 heures pour réaliser ce projet.
- Vous pouvez choisir le thème de votre application selon vos préférences (logiciel de gestion, jeu vidéo, etc.), mais vous devez respecter les contraintes techniques imposées.

5. Conseils :

a. Planification :

- Commencez par définir clairement le thème de votre application.
- Identifiez les principales fonctionnalités que vous souhaitez implémenter.

b. Conception :

- Dessinez un schéma de votre architecture avant de commencer à coder.
- Définissez les classes et les relations entre elles.

c. Implémentation :

- Avancez par étapes et testez régulièrement votre code.
- Assurez-vous que chaque fonctionnalité est bien implémentée avant de passer à la suivante.

d. Gestion du temps :

- Gardez un œil sur l'horloge et ajustez votre rythme de travail en conséquence.
- Prévoyez du temps pour les tests et la documentation à la fin.

Bonne chance et soyez créatifs !