



Examen 2008 - 2 h
PAR : Programmation d'Applications Réparties
Troisième année



Tous documents autorisés

★ **Exercice 1. Questions de cours RMI** (2pts). Plusieurs réponses sont possibles à certaines questions.

▷ **Question 1.** Quel interface doivent étendre toutes les interfaces de services distants ?

- `java.rmi.RemoteService`
- `java.rmi.Remotable`
- `java.rmi.server.Remote`
- Autre : _____

▷ **Question 2.** Que signifie l'erreur suivante :

```
1 java.rmi.ServerException: RemoteException occurred in server ...
2   java.rmi.UnmarshalException: error unmarshalling ...
3   java.lang.ClassNotFoundException: ...
```

- Le firewall du système bloque les communications RMI.
- Le stub n'est pas dans le CLASSPATH du serveur.
- Le stub n'est pas dans le CLASSPATH du client.
- Le stub n'est pas dans le CLASSPATH du rmiregistry.
- Le `java.policy` du client ne convient pas.
- Le `java.policy` du serveur ne convient pas.
- Autre : _____

▷ **Question 3.** Comment corriger l'erreur suivante lors du démarrage du rmiregistry :

```
java.rmi.server.ExportException : Port already in use : 1099
```

- Tuer le programme utilisant le port 1099 de la machine.
- Démarrer le rmiregistry sur un autre port.
- Passer l'option `--force` au rmiregistry.
- Désactiver le firewall de la machine.
- Autre : _____

▷ **Question 4.** De quelle(s) source(s) un serveur RMI **ne peut pas** charger de classes ?

- Le CLASSPATH du client.
- Le CLASSPATH du serveur.
- Un serveur HTTP ou FTP précisé par le client.
- Un serveur HTTP ou FTP précisé par le serveur.

★ **Exercice 2. Exercice RMI** (6pts).

On souhaite réaliser une application bancaire permettant de :

- Obtenir un gestionnaire de compte bancaire (toujours le même).
- Créer un compte bancaire en précisant le nom de son propriétaire.
- Créditer et débiter un compte bancaire.
- Afficher le contenu d'un compte.
- Afficher la liste de tous les comptes depuis le serveur (fonctionnalité pas accessible aux clients).

▷ **Question 1.** Donner le diagramme de classe UML décrivant l'application. Toutes les classes et interfaces doivent y figurer ; On supposera que les classes sont chargées statiquement ; La rigueur de la syntaxe UML n'est pas le plus important ici.

▷ **Question 2.** Indiquer pour chaque élément s'il doit être dans le CLASSPATH du client et/ou du serveur.

▷ **Question 3.** Écrivez les interfaces et classes composant l'application.

★ **Exercice 3. EJB** (12pts).

▷ **Question 1.** Quel service de l'environnement JEE permet la transparence de l'exécution de transactions distribuées ?

▷ **Question 2.** Ecrivez un Stateless Session Bean qui fournit une méthode de conversion de livre sterling en euro (1,05 pound = 1 euro).

▷ **Question 3.** A quoi sert la passivation des Session Beans ? Pourquoi est-elle inutile pour les Stateless Session Bean ?

▷ **Question 4.** Quels sont les tables créées dans la base de données relationnelle selon la stratégie Join Strategy en fonction du schéma de classes simplifié suivant. Quel est l'inconvénient de cette stratégie ?

```
public class Courrier {
    private Long id;
    private String destinataire;
    private String adresse;
}

public class Suivi extends Courrier {
    private boolean arrivé
}

public class Recommande extends Suivi {
    private Date envoi;
    private Date reçu;
    private boolean signature;
}
```

▷ **Question 5.** Complétez la classe Courrier avec les méthodes et les annotations nécessaires pour la rendre persistente.

▷ **Question 6.** Ecrivez le code du client permettant de retrouver toutes les instances de la classe Courrier pour une valeur donnée de destinataire.

▷ **Question 7.** On suppose qu'on dispose d'une classe Personne permettant de représenter l'auteur d'un courrier. Une personne est susceptible d'envoyer plusieurs courriers. Complétez la classe Courrier pour représenter la relation avec la classe Personne (annotation et méthodes).

▷ **Question 8.** Pourquoi peut-on dire que l'utilisation des Message Driven Bean simplifie la répartition de la charge sur plusieurs serveurs ?

▷ **Question 9.** Une transaction est démarrée. Que se passe-t-il si à l'intérieur de cette transaction une méthode d'un session bean ayant l'attribut transactionnel RequireNew est appelée ?