



Examen Final

Architecture et Développement Logiciels

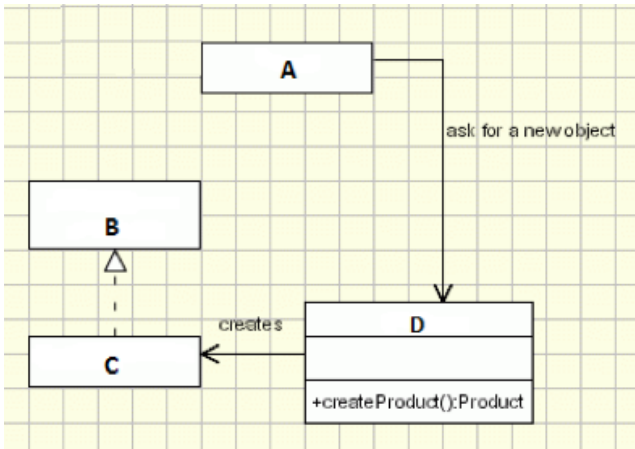
Nom	
Prénom	
Numéro Etudiant	

Remarques :

- Les documents ne sont pas autorisés ainsi que les appareils électroniques (PC, Tablette, téléphone, etc).
- La première partie A est sous forme de QCM. **Attention : une réponse fausse annule une réponse juste.**
- La seconde partie B est sous forme de questions libres à répondre sur la double feuille.

Partie A : Cochez la (ou les) bonne(s) réponse(s).

Questions	Réponses
1. Un design pattern est :	<input type="checkbox"/> Un paradigme des langages de POO.
	<input type="checkbox"/> une définition des implémentations spécifiques à des principes de conceptions
	<input type="checkbox"/> une définition des principes de la POO.
	<input type="checkbox"/> aucune réponse juste.
2. Le design pattern Adapter :	<input type="checkbox"/> permet de parcourir des collections d'objet d'implémentation différentes.
	<input type="checkbox"/> est un patron de structure.
	<input type="checkbox"/> permet de s'adapter à des interfaces et des classes qui utilisent ces interfaces, sans les faire évoluer.
	<input type="checkbox"/> aucune bonne réponse.
3. En UML une interface est :	<input type="checkbox"/> une classe abstraite.
	<input type="checkbox"/> un composant graphique
	<input type="checkbox"/> une agrégation composite.
	<input type="checkbox"/> aucune bonne réponse.
4. Quelle pattern définit une interface pour la création d'un objet en déléguant à ses sous-classes le choix des classes à instanciée :	<input type="checkbox"/> Façade.
	<input type="checkbox"/> Factory.
	<input type="checkbox"/> Composite
	<input type="checkbox"/> Iterator.

Questions	Réponses
La figure suivante représente le pattern Factory :	
 <pre> classDiagram class A class B class C class D { +createProduct() Product } A --> D : ask for a new object B .. > C D --> C : creates </pre>	

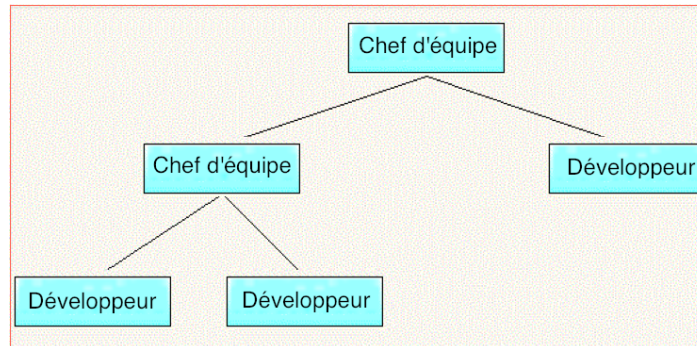
Questions	Réponses
1. Les lettres A, B, C et D sont définies comme suite :	<input type="checkbox"/> A=Client ; B=Factory ; C=Product (interface) ; D=Concrete Product . <input type="checkbox"/> A = Client ; B=Product (interface) ; C=Concrete Product ; D = Factory. <input type="checkbox"/> A=Factory ; B = Concrete Product ; C=Product (Interface) ; D=Client. <input type="checkbox"/> aucune bonne réponse.
2. Les EJB (Entreprise Java Bean) :	<input type="checkbox"/> permettent de construire des applications distribuées. <input type="checkbox"/> définissent un standard JavaBean pour faciliter la réutilisation et l'interopérabilité des composants middleware. <input type="checkbox"/> définissent l'un des modèles de composants principaux de J2EE <input type="checkbox"/> aucune bonne réponse.
3. Un EJB entité :	<input type="checkbox"/> est exécuté du côté client. <input type="checkbox"/> modélise une donnée persistante. <input type="checkbox"/> composé obligatoirement de deux interfaces local et distante. <input type="checkbox"/> aucune bonne réponse.
4. Un EJB session Stateful :	<input type="checkbox"/> maintient une conversation entre le client et le serveur. <input type="checkbox"/> est dédié à plusieurs clients. <input type="checkbox"/> représente la logique métier de l'application. <input type="checkbox"/> aucune bonne réponse.
5. Dans les EJB qu'est ce qui est utilisé pour effectuer les opérations d'accès aux données :	<input type="checkbox"/> l'entité manager. <input type="checkbox"/> Le conteneur. <input type="checkbox"/> l'entité bean. <input type="checkbox"/> aucune bonne réponse.

Partie B : Questions libres.

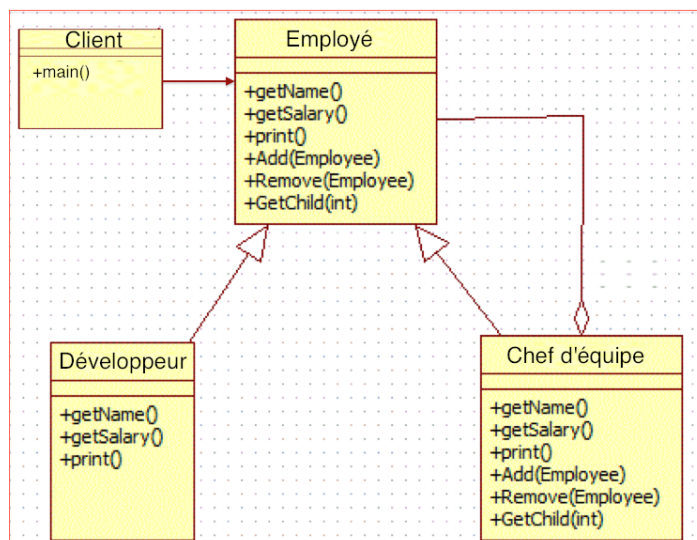
Questions	Réponses
-----------	----------

Exercice 1 :

Dans une petite boîte d'informatique, il y a 5 employés : 2 chefs d'équipe et 3 développeurs. L'organisation de la boîte est illustrée dans la figure suivante :



On souhaite imprimer le nom et le salaire des employés de haut en bas. Voici le diagramme UML proposé :



Questions :

- 1 Quelle est le design pattern utilisé ? Justifier.
- 2 Donner les avantages et inconvénients de ce design pattern.
- 3 Écrire en Java un exemple de code de Client qui crée la boîte d'informatique et affiche le nom et le salaire de ses employés de haut en bas.

Questions	Réponses
<p><u>Exercice 2 :</u> Soit une application appartenant à un fournisseur de matériel informatique. L'application est composée d'une Entité stockée dans une BD. Vous y accéder à travers une Session Façade qui sera implémentée comme un composant EJB Session. <u>Questions :</u></p> <ol style="list-style-type: none"> 1 Écrire la classe du bean entité, nommé Produit, elle contient : le libellé du produit et sa quantité en stock. 2 Écrire la classe correspondant à un EJB session qui consiste à gérer le produit. Quel type d' EJB session pensez-vous utiliser ? Pourquoi ? 3 Donner les différentes étapes nécessaires pour que cette application soit opérationnelle sur un serveur d'application ? 4 Proposer l'implémentation d'un client Java simple. 5 Avant d'accéder à un produit, le client devra s'authentifier. Proposer une solution élégante en utilisant la programmation orientée aspect. 	

Bon courage et bonne continuation.