



RECHERCHE

Search bar with 'Votre recherche' and 'OK' button

EVALUATION EN JAVA EE TEST 08

ExamenJEE, examen corrige java ee, examenjee

EXERCICE 1 (4 PTS)

- 1. Citer deux avantages et deux inconvénients d'une architecture multicouche?
2. Dans le domaine des serveurs d'applications, citer une solution propriétaire et payanté, et une solution libre et gratuite. (1 pt)
3. Donner l'arborescence des fichiers d'une application web de type JEE. (1 pt)
4. Donner la signature de la méthode d'une servlet qui traite une commande GET du protocole HTTP ? (1 pt)

EXERCICE 2 (5 PTS)

- 1. Quels sont les éléments constitutifs d'un serveur d'application? (1pt)
2. Quelle est la différence entre l'architecture client-serveur et le modèle MVC? (1pt)
3. Décrire par un schéma le modèle MVC sous le prisme de Java EE. Le schéma devra contenir les différentes couches de Java EE et préciser leur rôle. (1,5pts)
4. Définir le terme portée dans une application. Citer quatre types de portées (1,5pts)

Problème (11 pts)

Dans ce problème, l'utilisation des tags JSTL est préférable pour écrire les JSPs. Pour le mapping des tables de la base de données, vous pouvez utiliser JDBC. L'objectif est de développer une application Web permettant de gérer des contacts (créer un contact, rechercher un contact, modifier un contact, ...). Cette application doit être conforme au design pattern MVC (ie, des JSPs pour les vues, une Servlet comme contrôleur et des classes Java (JDBC) pour le modèle). Pour cela, on crée des JSPs pour afficher un contact, éditer un contact, ... Chacune de ces JSPs communiquera avec une Servlet qui aura le rôle de contrôleur: ControllerServlet.java. L'idée ici est que toutes vos requêtes doivent être sous la forme: ControllerServlet?do_this=[action à exécuter], le paramètre do_this est récupéré par la ControllerServlet pour connaître l'action à effectuer (par exemple, sélectionner la JSP qui aura à afficher la réponse).

- Partie Modèle:

Vous devez commencer par créer la table Contact:

```
CREATE TABLE Contact (
ID CONTACT generated always as identity primary key,
FIRSTNAME VARCHAR(255),
LASTNAME VARCHAR(255),
EMAIL VARCHAR(255),
PHONE INTEGER,
ADDRESS VARCHAR(255),
PROFESSION VARCHAR(125)
)
```

Dans le projet on suppose avoir un package nommé model sous le répertoire src. Il est nécessaire d'implémenter la classe Contact (un simple JavaBean à mapper avec JDBC) correspondant au mapping de la table Contact et la classe ContactFacade.java dans le package model.

-Partie Contrôleur:

Dans un package nommé controller, une servlet ControllerServlet.java implémente les fonctions provenant des besoins des JSPs développées dans la partie Vues.

- Partie Vues:

L'utilisateur arrive sur une page d'accueil accueil.jsp Cette page affiche la liste des contacts déjà créés. En plus, la page accueil.jsp contient les actions suivantes (liens):

- Créer un nouveau contact (redirection vers addContact.jsp)
-Supprimer un contact (redirection vers removeContact.jsp)
- Modifier un contact (redirection vers updateContact.jsp)
-Rechercher un contact (redirection vers searchContact.jsp)

La page accueil.jsp se trouve dans le repertoire /Webapp de votre application en supposant vous utilisez Eclipse. Cette page utilise le tag JSTL dans le but de générer dynamiquement, un tableau HTML, sans utiliser une boucle << for >> en code Java.

Attention: les JSPs ne peuvent pas accéder directement à la base de données car dans MVC les Vues n'appellent pas directement le Modèle. En effet, tous les appels vers le Modèle (à travers ContactFacade.java) doivent passer par le Contrôleur ControllerServlet.java. Par exemple, pour accéder à la page d'accueil accueil.jsp, l'utilisateur doit appeler la ControllerServlet (URL : http://localhost:8080/Contacts/ControllerServlet). Ensuite, la ControllerServlet transmet la requête vers accueil.jsp. Enfin, accueil.jsp récupère la liste des contacts à partir de la requête et l'affiche.

La page addContact.jsp propose un formulaire qui contient des champs pour renseigner un contact. Ses champs doivent correspondre aux attributs de la classe Contact.java. Le formulaire doit également avoir un bouton submit et un bouton annuler. Une fois le formulaire envoyé, il est traité par la servlet ControllerServlet. Par conséquent, il faut préciser à ControllerServlet l'action à effectuer.

La page addContact.jsp doit être créée dans Contacts/webapp/. Ensuite, la Servlet ControllerServlet récupère les informations du contact à partir des données du formulaire puis elle utilise le ContactFacade pour appeler la méthode addContact(long id, String firstname, String lastname, String email, ...). Cette méthode crée une instance de Contact, fixe ses attributs puis le sauvegarde dans la base de données. Même principe pour removeContact.jsp. Cette page-propose un formulaire avec un champ nommé idContact et un bouton << submit >>. Une fois cliqué, l'utilisateur est invité à confirmer la suppression par une boîte de dialogue JavaScript une fois confirmée; le formulaire fait appel à la

TUTORIELS

Table listing various technologies with dropdown menus: NODEJS (5), JAVA (1), Algorithme Avancés (4), KOTLIN (11), PHP (15), RUST (7), ANDROIDE DEVELOPPEMENT (8), JAVASCRIPT (7), UML (2), Langage Pascal (2), SECURITE INFORMATIQUE (1), IONIC (1), RESEAU (4), LINUX (2), CLOUD COMPUTING (8), LANGAGE C (10), Systeme d'exploitation (3), CSS (6), C++ (1), SQL (2), COMPLEXITE DES ALGORITHMES (0), CRYPTOGRAPHIE (1), RECHERCHE OPERATIONNELLE (1), HTML (FACILE), CSS (FACILE)

Quiz

Table listing various quizzes: QUIZ PROGRAMMATION C (0), QUIZ PHP (0), QUIZ HTML5 (0), QUIZ JAVASCRIPT (0), QUIZ CSS (0), QUIZ ARCHITECTURE DES ORDINATEURS (0), Quiz PYTHON (0), QUIZ C PLUS PLUS (0), QUIZ REACTJS (0), QUIZ OCL (0)

Forum

servlet ControllerServlet qui récupère l'id du contact à partir des données du formulaire puis elle appelle la méthode deleteContact(long id) de ContactFacade. Cette méthode se charge d'effacer le contact portant cet id dans la base de données. Ainsi de suite pour les autres fonctionnalités.

1. Proposer un contenu du fichier web.xml pour effectuer le mapping pour des différentes fonctionnalités: afficher contacts, nouveau contact», «recherche et modifier contact. (1,5 pts)
2. Créer le bean représentant un contact et la classe ContactFacade pour l'accès à la base de données (avec ses différentes méthodes). (2,5 pts) (1,5 pts)
3. Créer le formulaire un nouveau contact (redirection vers addContact.jsp). (1,5 pts)
4. Créer la page modifier contact (redirection vers updateContact.jsp). (2 pts)
5. Créer la page Home contact» (redirection vers accueil.jsp)
6. Proposer un code pour le contrôleur ControllerServlet pour gérer les fonctionnalités du projet (2 pts)



ACCÉDER A LA CORRECTION



CORRECTION PROBLEME :

TÉLÉCHARGER L'EXERCICE SOUS FORME DE PDF

Algorithmme : Exercices corrigés #03 Exécution Manuelle d'un Algorithmme

Un algorithme est une suite ordonnée d'instructions qui indique la démarche à suivre pour résoudre une série de problèmes équivalents et bien précis. Dans cet exercice corrigé nous allons aborder plusieurs notions clé en algorithmique. Le Lien Direct vers l'énoncé de L'exercice [ici](#) !!!"

ENCORE PLUS :



JAVA EE

SUJET 01 | JAVA EE

Quelle différence faites-vous entre une plateforme mobile et un système d'exploitation mobile ...

EXAMEN JAVA EE

JAVA EE

SUJET | JAVA EE

L'algorithme du tourniquet, circulaire ou round robin est un algorithme ancien, simple, fiable ...

EXAMEN JAVA EE



☆☆☆☆ Aucune note. Soyez le premier à attribuer une note!

AJOUTER UN COMMENTAIRE

Nom *

E-mail

Site Internet

A propos du site web	0
Présentation des membres	0
Programmation Mobile	0
Discussions générales	0
Algorithmique	0
C++	0
Flutter	0

A LA UNE !

- C'est quoi le C++ ?
Par joel_... Le 10/02/2023
- C'est quoi le phishing ?
Par joel_... Le 08/02/2023
- C'est quoi le ransomware ?
Par joel_... Le 29/01/2023

TRADUIRE LE SITE

- Français
- English
- Español
- Italiano
- Deutsch
- Nederlands
- Portuguesa

TAGS

- programme
- javascript
- language
- java
- PHP
- python
- programmation

Météo

-26 °C Partiellement nuageux

Min: -26 °C | Max: -26 °C | Vent: 8 kmh 33°

Derniers quiz

- QUIZ OCL #01
- QUIZ REACTJS #01
- QUIZ C PLUS PLUS 02



Nom d'utilisateur ou e-mail

Mot de passe

Rester connecté

[Créer un compte](#) | [Mot de passe perdu ?](#)

Valider

Twitter du Panda

Tweets de @Pandacodeur

Rien à voir ici. Pour le moment

Examen Algorithmme Avance & Complexite

Algorithmiques Avancés et Comp

RECHERCHE OPERATIONNELLE

Cours des Méthodes de Résoluti



Anti-spam *



Ajouter

RECHERCHE OPERATIONNELLE

Coloriage des Cartes - CSP

Exercices Methodes du Simplexe

CLOUD COMPUTING

Google Mashup

HEROKU

OpenStack

CloudStack

Google App Engine (GAE)

MICROSOFT AZURE

SERVEUR NAS

La Sécurité dans le Cloud

CLOUD COMPUTING

Exercice Microsoft Azure : Az

Exercice Microsoft Azure : Co

Exercice Microsoft Azure : Az

Travaux Dirigés : Sécurité des

EXERCICE Connaissances général

Travaux Dirigés : CloudStack

Travaux dirigés Serveurs Nas

Travaux Dirigés : VIRTUALISATI

Exercice Google App Engine

EXAMEN CLOUD COMPUTING

EXAMEN CLOUD COMPUTING SUJET 1

EXAMEN CLOUD COMPUTING SUJET 2

EXAMEN CLOUD COMPUTING SUJET 3

Ingénierie des connaissances Examen

Examen Ingénierie des connaissances

Examen JAVA

Examen Java 01

TP KOTLIN

Examen Fouille de données

Examen Recherche opérationnelle

Examen recherche opérationnelle