

Question atelier 2 ASI1 :

1. Qu'est-ce que des Services Web Full Rest ?

Un service Rest se base sur des principes de design d'API tels que :

- L'utilisation du protocole HTTP
- L'implémentation de 4 méthodes de requête comme : GET, POST, PUT, DELETE
- Utilisation d'URI, de chemins d'accès
- Le web service doit être stateless, donc pas de prise en compte de l'état du site donc pas d'enregistrement de l'information

Un service Rest Full respecte tous ces critères strictement

2. Quelles sont les contraintes imposées par ce type de service ?

3. Qu'est-ce qu'un gestionnaire de dépendance ?

Un gestionnaire de dépendance est un outil qui est mis en place pour gérer les différentes dépendances de notre projet.

4. Maven est-il le seul ?

Nous avons aussi la possibilité d'utiliser "Composer" ou "Bower" comme gestionnaire de dépendance.

5. Quel est l'avantage d'utiliser un gestionnaire de dépendances ?

L'avantage est d'avoir une meilleure supervision de votre projet.

6. Quelles sont les grandes étapes de fonctionnement de Maven ?

- créer une arborescence standard du code et de ses ressources
- télécharger, mettre à jour et configurer les bibliothèques nécessaires au projet
- compiler le projet, effectuer des tests unitaires sur le code et packager le résultat

7. Qu'est-ce qu'une application classique Entreprise Java Application ?

Une application classique Entreprise Java Application est une application web qui est réparti sous trois couches principales sous la forme d'une architecture en format 3-tiers ou plus.

8. Donner un exemple d'usage avec web service, JSP, JavaBean, EJB et JMS.

9. Qu'est-ce que Spring ?

Spring est un Framework libre. Il permet de construire l'infrastructure d'une application Java et d'en faciliter le développement.

10. Qu'apporte Spring boot vis-à-vis de Spring ?

Permet d'être bien plus productif en ce qui concerne le développement d'une API. Il permet de se concentrer sur la partie métier de son application en laissant de côté toute la complexité de la création d'une API.

11. Qu'est-ce que Spring Boot ?

Le projet Spring Boot est une extension du Spring Framework pour mettre en place rapidement des applications Java. Grâce à un système modulaire de dépendances et un principe de configuration automatique, il permet de disposer d'une structure de projet complète et immédiatement opérationnelle. La perte de souplesse dans les choix d'architecture est compensée par la rapidité de mise en place et un large choix de paramètres de configuration. Spring Boot n'est pas une nouvelle version du Spring Framework. Il s'agit d'une utilisation particulière du Framework, évitant aux développeurs de gérer une complexité technique souvent inutile au moment de l'initialisation d'un projet.

12. Quels sont les points communs/différences entre JEE et Spring Boot ?

Java EE :

- Java EE de l'industrie a approuvé la norme API basée sur le cadre
- Il est principalement basé sur les annotations et CDI
- JFC MVC Framework de développement web
- Spécification JPA pour traiter DB opération
- API JTA avec la mise en œuvre
- Conteneur d'EJB et de POJO en fonction de la mise en œuvre
- Oracle licence

Spring Framework :

- Basé sur du CIO et de l'AOP
- Basé sur un fichier XML de configuration (maintenant qu'ils sont en tirant parti de l'annotation)
- Utilise Printemps DAO cadre (basé sur le Modèle de design pattern) pour se connecter à la base de données
- Fournit une couche d'abstraction à l'appui de divers JTA mise en œuvre vendeur
- S'intègre avec les différents Java vendeurs pour le support des fonctionnalités
- Fournit de bout en bout plate-forme pour construire des applications web atteindre le couplage à l'aide de DI et AOP
- Licence Open source

13. Qu'est-ce qu'une annotation ?

Les annotations de Spring Boot sont une forme de métadonnées qui fournissent des données sur un programme. Elles ne font pas partie de l'application que l'on développe. Elles n'ont pas d'effet direct sur le fonctionnement du code qu'elles annotent.

14. Quels apports présentent les Annotations ?

Les annotations sont utilisées pour fournir des informations supplémentaires sur un programme.

15. Comment fait-on pour créer un Web Service Rest avec Spring Boot ?

- Initialisation du projet
- Créer le modèle de donnée
- Ajouter un contrôleur
- Lancer l'application

16. Qu'est-ce qu'un container de Servlet ?

C'est un logiciel qui exécute des servlets. Parfois, ils sont appelés moteur web ou moteur de servlets.

17. Comment fonctionne un container de Servlet ?

Le container de Servlet a la charge d'instancier, d'initialiser et de détruire les servlets d'une application. C'est également le conteneur qui fournit une instance de `HttpServletRequest` et de `HttpServletResponse` pour chaque requête.

18. Expliquer la philosophie « Convention over Configuration » de Spring boot ?

L'expression indique que le développeur doit seulement se concentrer sur les aspects non conventionnels de l'application.

19. Expliquer ce qu'il se passe lors de l'exécution « `SpringApplication.run(App.class,args)` »

Cette ligne démarre une application Spring comme une application autonome à partir de la méthode `main`. Elle crée une instance `ApplicationContext` appropriée et charge les beans.

20. Qu'est-ce qu'un DAO ?

Un DAO (Data Access Object) permet de relier les objets Java (couche métier) et la base de données (couche persistante). Il consiste en un ensemble d'objets capables de lire, écrire, modifier ou supprimer les éléments stockés. Tous les échanges entre la couche persistante et la couche métier se font par l'intermédiaire de la couche DAO.

21. En quoi est-ce intéressant d'utiliser ce pattern ?

Ce pattern garantit l'encapsulation de la partie mappage dans des objets spécifiques (de la couche DAO), ce qui évite d'effectuer des changements directement sur les objets métiers, qui doivent fonctionner quelles que soient les circonstances.

22. Qu'est-ce qu'un Singleton ?

Un singleton est un design pattern qui restreint l'instanciation d'une classe Java à un objet unique (single). Il permet l'amélioration des performances d'un système.

23. Que permet de réaliser les Entity dans Spring boot ?

Elles permettent de définir les informations sur la table représentant cette entité en base de données. Il est possible de définir le nom de la table grâce à l'attribut « name ». Par défaut le nom de la table correspond au nom de l'entité.

24. Est-ce spécifique à SpringBoot ?

...

25. Combien d'instances avez-vous créées lors de l'usage de « Service » en Spring boot ?

Nous n'avons créé qu'une seule instance du service de Springboot.

26. Pourquoi ?

L'instance se lance sur un port de la machine et si celui-ci est déjà occupé, nous ne pouvons en lancer une seconde.

27. Que fournit le CRUD Repository de Spring boot ?

28. Que sont les CRUD ?

29. Qui réalise l'implémentation de la méthode findByProperty lors de la création d'un repository en Spring Boot ?

30. Comment gère-t-on les relations One To One, One to Many et Many to Many avec JPA ?

31. Qu'est-ce qu'une Architecture SOA ?

Une architecture SOA, ou architecture orientée services est une architecture qui répond à deux critères :

- Cohérence interne forte
- Couplage externe faible

Une architecture SOA présente des avantages en termes de réusabilité.

32. Qu'est-ce qu'une architecture Micro-Service ?

Une architecture Micro-Service est une variante d'architecture SOA qui impose la structuration d'une application comme un ensemble de services faiblement couplés qui communiquent entre eux via des API. Ce type d'architecture permet d'apporter des modifications à une ressource sans créer de conflits avec les autres ressources de l'application.

33. Il y a-t-il d'autres architectures existantes ?

D'autres architectures existent. Par exemple, les architectures trois-tiers et MVC.

34. Quels sont leurs avantages/inconvénients ?

Un avantage de la MVC est qu'elle apporte une grande clarté d'architecture. La modification du traitement ne change pas la vue. En revanche, dès que le système désiré fait appel aux technologies du web, la MVC a besoin d'être complétée par des mécanismes d'inversion de contrôle et d'injection de dépendances.

L'architecture trois-tiers nécessite plus de rafraîchissements de la vue que la MVC étant donné que si une vue modifie des données, toutes les vues concernées par la modification doivent être mises à jour.