Master 1 informatique – Université Claude Bernard Lyon 1 – mercredi 25 janvier 2023

M1IF 03 – Conception d'applications Web – Examen

Durée : 1 heure 00 – Documents autorisés (4 pages max) – Ordinateurs, calculatrices, tablettes, téléphones portables… interdits

Toutes les questions ont au moins une bonne réponse.
Il faut cocher toutes les bonnes propositions d'une question pour avoir un point.
Si vous cochez une mauvaise proposition, vous annulez la question.
Remplissez au stylo noir ou bleu la case de la ou des bonne(s) réponse(s) ; une croix ne suffit pas.
Ne barrez pas une mauvaise réponse, mettez du blanc.
Ne redessinez pas une case que vous avez effacée, laissez blanc.
La première question n'est pas prise en compte dans la notation, mais elle est indispensable pour la notation des autres questions.

1. Identification du sujet :
   Cochez la case B

2. En JavaScript, la méthode fetch() :
   a) attend la réponse
   b) permet d'interroger une autre API que celle de l'application
   c) renvoie une promesse
   d) ne fonctionne qu'en HTTPS
   e) envoie une requête

3. Dans une Single-Page Application (SPA) simple telle que celle présentée en cours et utilisée en TP :
   a) l'affichage d'une vue nécessite un mécanisme de templating
   b) il faut modifier la structure de l'arbre DOM pour afficher une nouvelle vue
   c) une vue ne peut pas être affichée tant que toutes les données n'ont pas été chargées
   d) un changement de vue est déclenché par une requête du client
   e) un changement de vue est déclenché par un événement côté client

4. L'interface `Performance` de la spécification Web Performance du W3C permet de connaître les temps (timestamps) :
   a) du premier affichage de la page
   b) du traitement d'une réponse côté client
   c) de construction de l'application shell
   d) du traitement d'une requête côté serveur
   e) de chargement d'une ressource

5. Une transaction avec négociation de contenus en HTTP :
   a) comporte un header Content-Type dans la réponse
   b) comporte un header Retry dans la réponse
   c) comporte un header Content-Type dans la requête
   d) comporte un header Accept dans la requête
   e) comporte un header Refuse dans la réponse

6. Pour chiffrer les communications avec un serveur :
   a) un client utilise une clé générée par une autorité d'authentification
   b) un client utilise une clé privée auto-générée
   c) un client utilise une clé fournie par une autorité de certification
   d) un client utilise une clé envoyée par le serveur
   e) un serveur et un client interagissent avec une clé symétrique

7. L'approche DevOps appliquée au Web, c'est être capable :

a) de déployer une application web dans le metavers
b) d'administrer un serveur nginx
c) d'optimiser les performances d'une application web
d) de concevoir une application web à l'aide de design patterns
e) de sécuriser des ressources en les isolant du réseau

8. Pour passer des paramètres dans une requête HTTP DELETE :
   a) Il faut mettre ces paramètres dans le corps de la réponse en JSON
   b) Il faut mettre ces paramètres dans le corps de la réponse en URL-encoded
   c) Cela ne sert à rien car la ressource sera détruite de toutes façons
   d) On peut « hacker » la requête en les passant dans un header
   e) Il faut mettre ces paramètres dans l'URL en URL-encoded

9. Pour déployer une application web en Java sur le serveur Tomcat de votre VM, vous pouvez utiliser :
   a) l'interface web de l'application manager de Tomcat
   b) l'API de l'application manager de Tomcat
   c) la commande scp
   d) la commande ssh
   e) un goal maven

10. En REST :
    a) le serveur dirige le workflow de l'application
    b) une ressource est un objet sur lequel on peut faire des opérations de type CRUD
    c) une ressource est un objet qui est stateless (sans état)
    d) le client dirige le workflow de l'application
    e) une ressource est un objet qui expose une interface uniforme

11. Pour exposer une Web API sur le web, les bonnes pratiques recommandent de :
    a) la référencer une un site dédié (comme programmableweb.com)
    b) permettre la négociation de contenus
    c) fournir un client
    d) fournir une documentation OpenAPI
    e) fournir une version installable en local

12. La modularité d'une application Web peut être favorisée par :
    a) l'ajout d'un pattern façade
    b) la séparation des rôles entre serveur et client
    c) l'utilisation d'un framework
    d) l'utilisation d'un proxy
    e) l'utilisation d'un pattern MVC de type 2

13. Parmi les bonnes propriétés du web qui sont promues par l'approche REST, il y a :
    a) la scalabilité
    b) la sécurité
    c) le fait que les contenus sont hypermédias et distribués
    d) la simplicité d'accès
    e) l'extensibilité

14. Un ETag permet :
    a) d'identifier une version d'une ressource
    b) de définir une classe de ressources
    c) d'identifier l'utilisation d'une ressource par un client donné
    d) d'identifier une ressource
    e) d'identifier le chargement d'une ressource par un client donné

15. Un Content Delivery Network (CDN) :
    a) référence des outils utilisés par beaucoup de serveurs
    b) ne renvoie que des fichiers statiques
    c) référence des outils utilisés par beaucoup de clients
    d) nécessite une authentification
    e) permet de déployer son API plus facilement

16. Une redirection :
    a) peut se faire en HTTP sans en informer le client
    b) peut se faire côté serveur sans en informer le client
    c) provient soit d'un bug de l'application, soit d'une mauvaise requête du client
    d) renvoie un code d'erreur HTTP
    e) peut se faire en HTTP mais cela implique que le client renvoie une nouvelle requête

17. Un token JWT :
    a) est valide pour un et un seul utilisateur
    b) peut être révoqué par le client
    c) doit être authentifié par le serveur qui l'a émis
    d) permet d'accéder aux variables de session
    e) est valide pour un et un seul client

18. Avec le moteur de templates de Mustache, on peut :
    a) rechercher des variables dans le document
    b) générer un document à partir de données JSON
    c) générer un document à partir de données YAML
    d) générer du contenu optionnel
    e) boucler sur les éléments d'un tableau

19. Quel est le protocole de sécurité que vous avez utilisé pour sécuriser (passer en HTTPS) votre serveur nginx :
    a) SSL
    b) TLS
    c) SFTP
    d) XSS
    e) TCP

20. Parmi les propositions suivantes, cochez celles où la méthode appartient à la classe indiquée.
    a) `HttpFilter – doPost()`
    b) `GenericServlet – service()`
    c) `GenericServlet – init()`
    d) `JspServlet – doBean()`
    e) `HttpServlet – doDelete()`

21. Cochez les cases qui correspondent à des scopes de beans JSP.
    a) internal
    b) response
    c) request
    d) page
    e) context

22. Pour implémenter le pattern MVC côté serveur en Java, d'après les bonnes pratiques vues en cours et en TP :
    a) un generic handler est un contrôleur principal
    b) une JSP peut appeler un contrôleur
    c) un template génère une vue
    d) une servlet peut être un contrôleur délégué

e) un java Bean est un élément du modèle

23. Une réponse HTTP ne comportant pas de contenu :
    a) peut contenir un header Location
    b) peut contenir un header Content-Type
    c) ne doit pas être envoyée
    d) doit être une redirection
    e) ne peut être qu'une réponse à une méthode DELETE

24. Pour optimiser le chemin critique de rendu (CRP), on peut :
    a) regrouper plusieurs ressources en une seule
    b) charger certaines ressources en `async`
    c) charger certaines ressources en `eager`
    d) charger certaines ressources en `lazy`
    e) minifier les scripts JS

25. Parmi les design patterns suivants, quels sont ceux utilisés dans l'application Web que vous avez réalisée côté serveur ?
    a) Inversion de contrôle
    b) Chaîne de responsabilité
    c) Singleton
    d) Agrégation de données
    e) Contexte