

TP2- partie 2

Protocole de communication - HTTP

Date de remise : vendredi 7 décembre avant minuit

Procédure de remise : remettre tout le **projet Java** incluant le dossier **siteHTTP** dans le dossier **tp2-partie2** du répertoire du **dépôt** du cours :

/Haouari, Dorsaf/5A3-soutien technique/tp2-partie2

Ce travail peut être fait en équipe de 2 maximum

CE TRAVAIL EST NOTÉ SUR 100 ET COMPTE POUR 7% DE LA NOTE FINALE DU TRAVAIL PRATIQUE 2 (QUI VAUT 15 POINTS AU TOTAL)

Dans ce travail, vous devez écrire un programme Java qui simule une communication HTTP entre un client et un serveur web.

Vous devez:

- Créer un serveur web capable de recevoir des requêtes de la part d'un client et de lui répondre tout en utilisant le protocole HTTP
- Utiliser un client web existant (un navigateur)
- Le serveur doit pouvoir répondre au scénario suivant :
 - Le client demande un fichier HTML
 - Le serveur reçoit la requête et la décode afin de comprendre le besoin du client.
 - Si le fichier HTML existe sur le serveur alors le serveur doit retourner le fichier au client. Le navigateur affichera alors la page demandée. On suppose que les ressources demandées par le client se trouvent toujours dans le dossier **siteHTTP**. Ce dossier doit être mis à l'intérieur de votre projet Eclipse. Vous devez y accéder à l'aide **d'un chemin relatif**
 - Si le fichier HTML n'existe pas alors le serveur retourne le code d'état adéquat avec un message d'erreur approprié. La console d'Eclipse affichera alors la réponse du serveur.
- **Autres instructions à respecter :**
 - Le serveur doit être toujours fonctionnel. Il ne doit pas s'arrêter afin de pouvoir répondre à des requêtes successives.
 - Le serveur doit permettre la communication de plusieurs clients à l'aide de Thread

- **Codes Java :**
 - Vous devez organiser votre code dans des méthodes différentes. Ne mettez pas tout le code dans la méthode *main*
 - Le code doit être lisible. Mettez des commentaires pour expliquer le code. Des points seront accordés pour la présentation et la lisibilité du code
- **Directives pour normaliser l'exécution :**
 - Dans le dossier siteHTTP, vous devez mettre 3 fichiers HTML nommés accueil.html, contact.html et formulaire.html. Vous y mettez du contenu html de votre choix.
 - Le port de connexion sera le 8085
 - La classe principale qui contient la méthode *main* doit être nommée ServeurSocket.java
 - Pour tester l'exécution du programme, je vais lancer la classe ServeurSocket.java et je taperais une requête du genre *http://localhost:8085/accueil.html*. Assurez-vous que votre programme sera capable de s'exécuter de cette façon
- **Bonus :** Le travail suivant est optionnel et permet de donner un bonus de 2 points à l'examen final
 - Au lieu d'utiliser un client existant, vous devez implémenter vous-même le client (le navigateur). Le navigateur doit comporter une barre d'adresse dans laquelle l'utilisateur est capable de faire rentrer la page HTML demandée. Juste en dessous, le navigateur doit avoir une zone qui permet d'afficher la réponse du serveur.
 - La classe pour implémenter le client doit être nommée ClientSocket.java
 - Directives pour normaliser l'exécution :
 - Pour tester l'exécution du programme, je vais lancer la classe ServeurSocket.java et la classe ClientSocket. Sur le client, je tape une requête du genre *http://localhost:8085/accueil.html*
 - Les deux points de bonus ne seront attribués que si tout le travail demandé est exécutable et fonctionne correctement.