	Faculté des Sciences de Sfax Département d'informatique et des Communications	Année Universitaire 2025 – 2026 Auditoire LSI 3
	Matière : Développement d'applications réparties Responsable du cours : Corinne Amel ZAYANI Enseignantes des TP : Fatma ELLOUZE, Zeineb DAOUD	

TP 1 : Les sockets en mode connecté TCP

Activité 1-1 : Première connexion Client/Serveur

Objectifs

- Apprendre à établir une communication Client/Serveur (C/S) en Java à l'aide des sockets en mode connecté TCP.

Environnements

- JDK ≥ 1.8
- Eclipse IDE for Java developers Ou IntelliJ IDAE > 1.8 OU VS code
- Git

Type de l'activité

- Individuel

Ressources

- Consultez séance 1 du cours "Développement d'applications réparties"
- Guide pratique : Intégration de Git avec Eclipse et IntelliJ

Livrables

- Repository git qui porte le nom "TP1-DAR2025"
- Le lien du repository dans l'espace de dépôt de la plateforme ENT
- Le fichier READ ME mis à jour

Enoncé

Dans cette activité, vous êtes demandés à développer une connexion entre un client et un serveur en utilisant des sockets en mode connecté.

Vous devez créer deux applications distinctes :

- une première application permettant de lancer le serveur,
- une deuxième application permettant de lancer le client.

Travail demandé

Cette activité comporte cinq étapes :

Étape 1 : Préparation de l'environnement Git

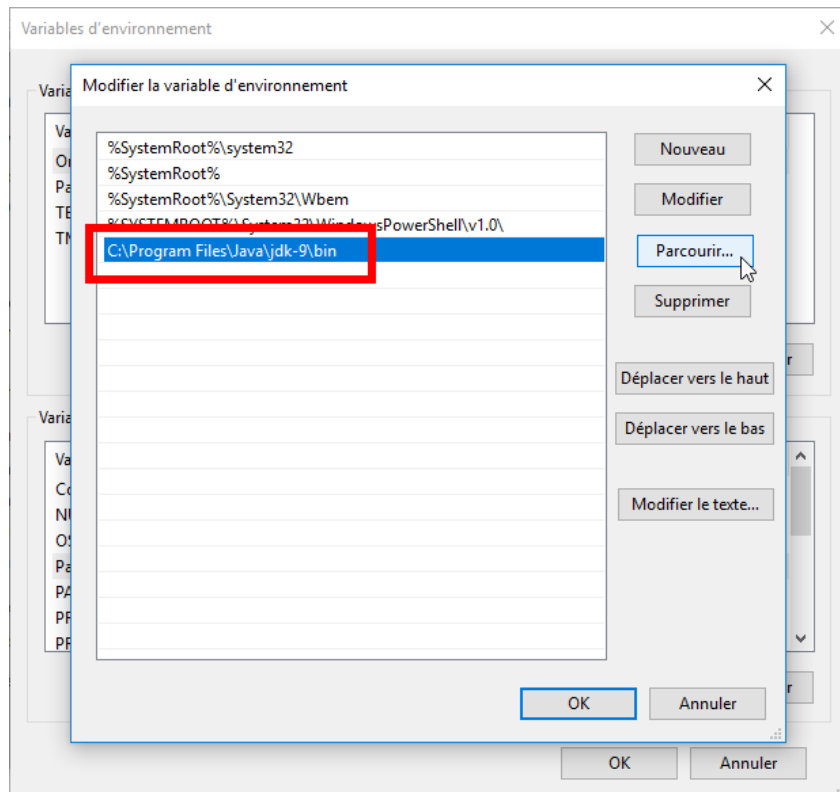
1. Vérifiez l'installation de Git sur votre système en tapant :
git --version

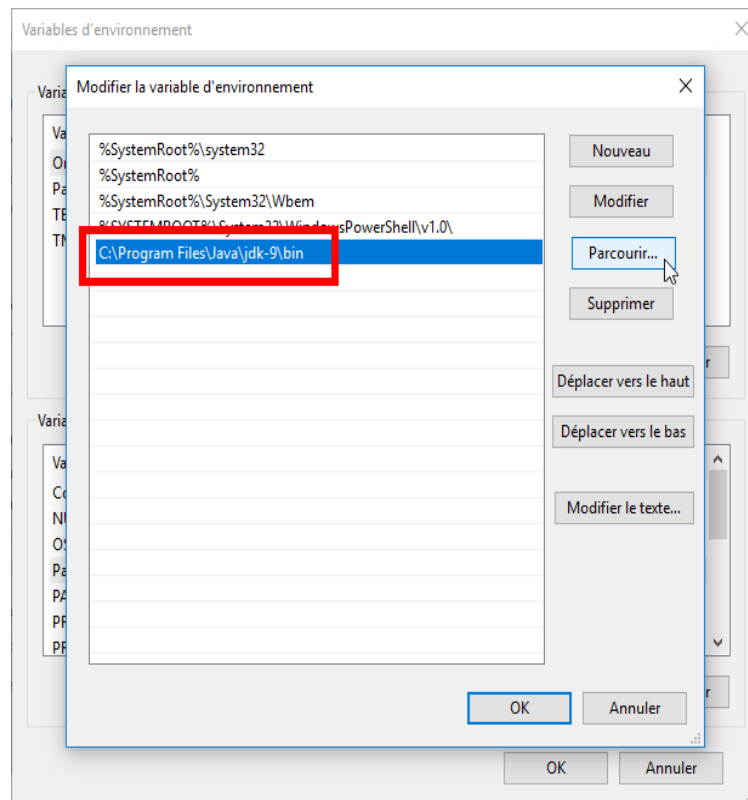
Étape 2 : Préparation de l'environnement Java

2. Vérifiez la version de Java installée sur votre système

Dans un terminal, tapez la commande suivante : **java --version**

Si cette commande ne fonctionne pas, il est alors nécessaire de mettre à jour la variable d'environnement PATH de votre système d'exploitation pour contenir le chemin d'accès au dossier bin de votre JDK comme le montre la figure ci-dessous.





3. Dans votre espace de travail, créez un répertoire sous le nom « **vousNomPrenom_LSI3_DevAppReparties** »
4. Créez un document Word qui porte votre « **VotreNomPrenom_Activite_1_1.docx** » dans lequel vous allez insérer les imprimés écrans de chaque étape demandée.

Étape 3 : Création du serveur

Dans cette étape, vous devez créer l'application **serveur**.

1. Lancez l'IDE Eclipse (ou IntelliJ) en vous plaçant dans le répertoire créé à l'étape 1.
2. Créez un nouveau projet Java en conservant les réglages par défaut et en le nommant « **TP1-DAR2025** » :
 - Dans **IntelliJ IDEA** : cochez la case **GitRepository** lors de la création du projet.
 - Dans **Eclipse** : suivez les étapes décrites dans le fichier "Guide pratique : Intégration de Git avec Eclipse et IntelliJ" en cochant sur la case **GitRepository** (dans IntelliJ)
3. Développez le code Java de l'**application serveur** :
 - a. Créez un package nommé « **serverPackage** » dans le répertoire « **src** ».
 - b. À l'intérieur de ce package, créez une classe appelée « **Server** », en activant la génération automatique de la méthode **main()**.
 - c. Dans le corps de la méthode **main** de la classe **Server**, saisissez le code suivant :

```
// La première étape :
//à compléter (TODO)
S.o.p("Je suis un serveur en attente la connexion d'un client ");
// La deuxième étape :
```

```
S.o.p("un client est connecté");  
// La dernière étape : Fermer socket  
// à compléter (TODO)
```

d. Complétez les parties du code signalées par **TODO** dans le programme saisi.

NB : N'oubliez pas d'importer les paquetages nécessaires et de gérer les exceptions appropriées.

e. Lancez l'application serveur.

f. Interprétez le résultat affiché dans la console dans le fichier README.

Étape 4 : Création du client

Dans cette étape, vous devez créer l'application client.

1. Développez le code Java de l'**application client** :

a. Créez un package nommé « **clientPackage** » sous le répertoire « **src** », dans lequel, vous créez une classe qui porte le nom « **Client** » et choisissez la génération automatique de la méthode main().

b. Dans le corps de la méthode main de la classe Client, saisissez le code suivant :

```
// La première étape :  
//à compléter (TODO)  
s.o.p("Je suis un client pas encore connecté..");  
//à compléter (TODO)  
s.o.p("je suis un client connecté");  
// La dernière étape : Fermer socket  
//à compléter (TODO)
```

c. Idem pour 3.c). de la 2ème étape.

d. Lancez le client.

e. Interprétez le résultat affiché dans la console dans le fichier README.

f. Changez le nom du localhost par une adresse IP.

NB : À chaque étape, insérez les captures d'écran correspondantes dans le document Word créé lors de la première phase.

Étape 5 : Dépôt des livrables

Dans cette dernière phase, vous êtes invités à :

- Déposer votre **document Word** dans l'espace de dépôt sur l'**ENT**.
- Mentionnez le lien du repository dans l'espace de dépôt de la plateforme **ENT**