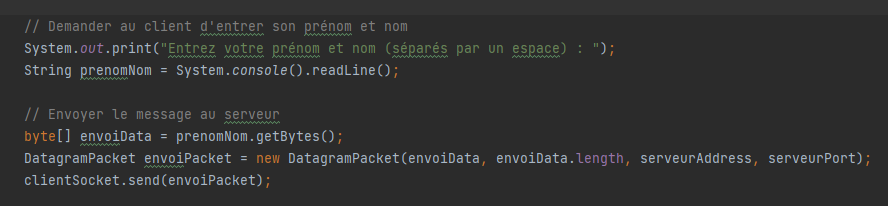
**Examen TP**

**Exercice 1 :**

****1.Le serveur démarre sur le port : 1234, et attend les demandes de connexions depuis un ou plusieurs clients.

2. Le client connecté envoie au serveur un message de la forme

« Prénom Nom »

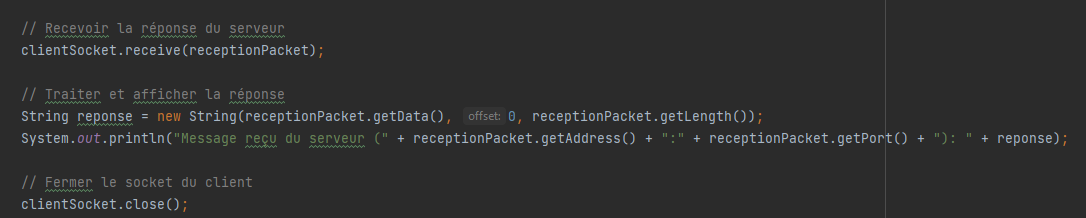


3. Le serveur lui répond par un message de la forme

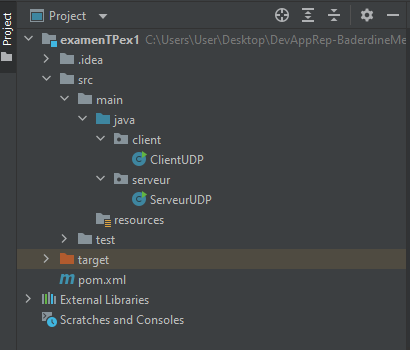
« Bienvenu Prénom Nom ».





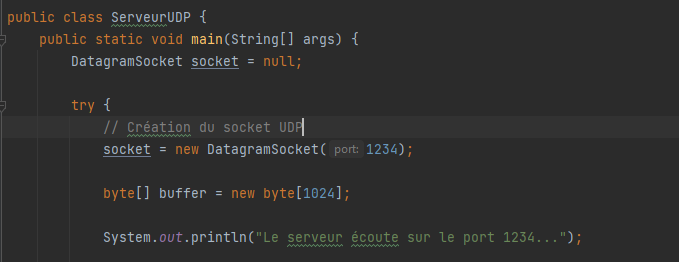
4. Le client affiche le message reçu, ainsi que l’adresse et le numéro de port de son serveur.

**Structure de projet :**

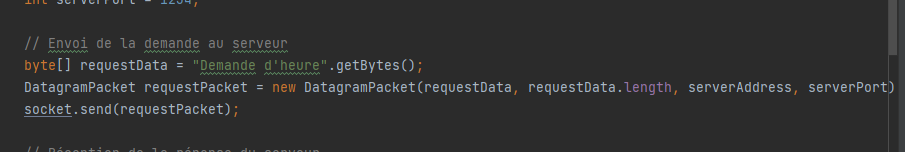


**Exercice 2 :**

1. Le serveur écoute, sur le port : 1234, pour les demandes entrantes des clients.

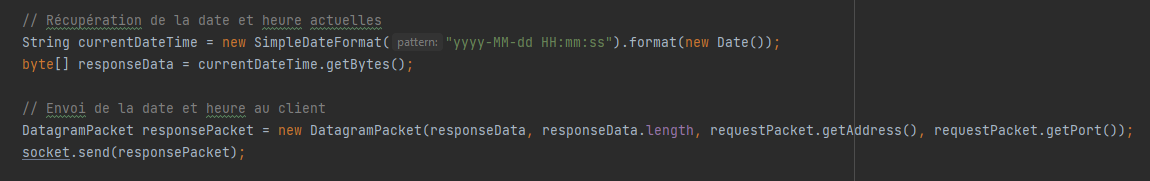
****

1. Le client demande l'heure au serveur.

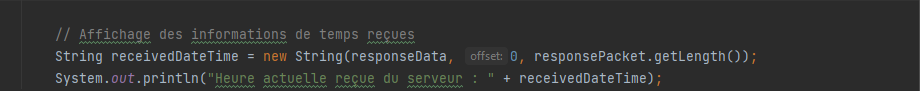
****

1. Lorsqu'une demande est reçue chez le serveur, il répond en renvoyant

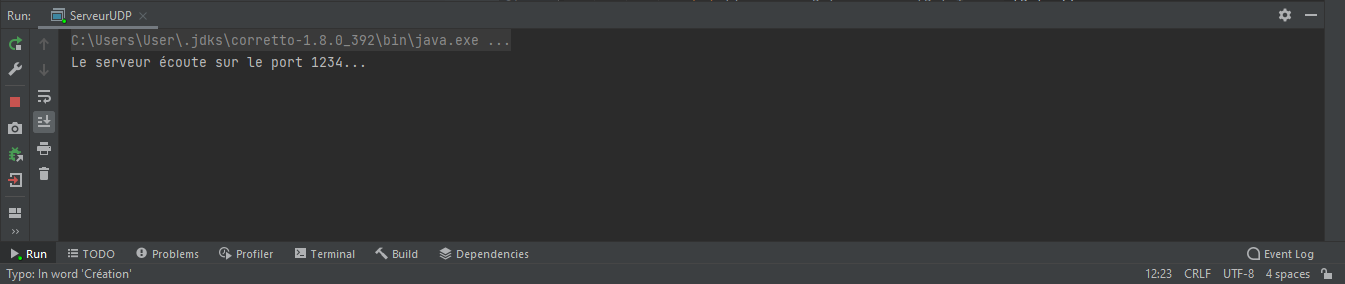
la date et l'heure actuelles. Le format de l'heure demandé est " yyyy-MM-dd HH:mm:ss ", mais vous pouvez le personnaliser selon vos préférences

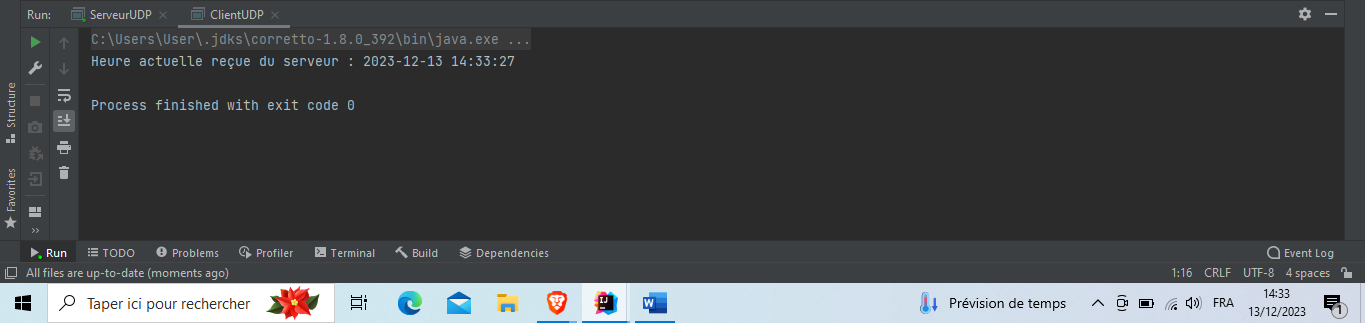


1. Le client affiche les informations de temps reçues.



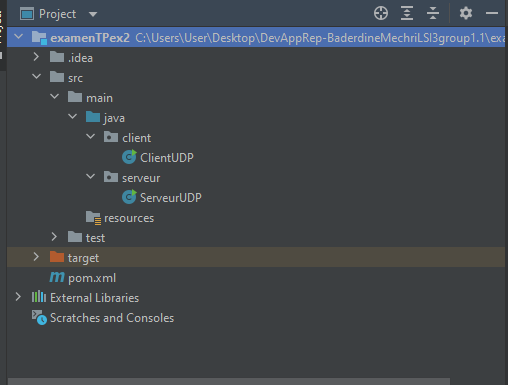
1. Pour tester le service de temps, exécutez d'abord le serveur, puis exécutez le client.





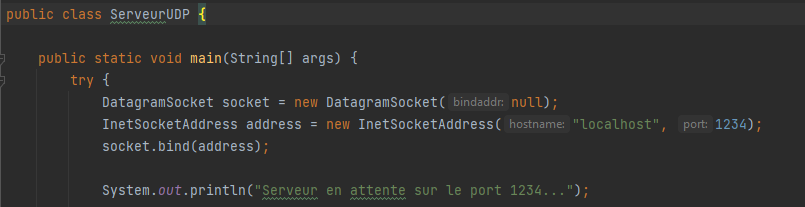


**Structure de projet**

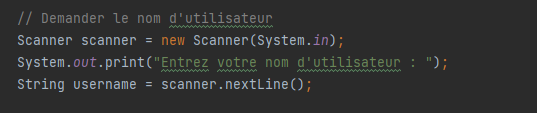
****

**Exercice 3 :**

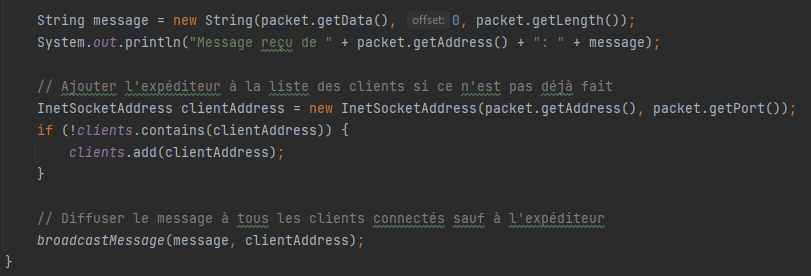
1. Le serveur écoute, sur le port : 1234, pour les demandes entrantes des clients distants en utilisant “InetSocketAddress”.

****

1. Le client connecté permet aux utilisateurs d'entrer leur nom d'utilisateur et d'envoyer des messages de chat, qui sont ensuite diffusés à tous les autres clients connectés en temps réel.



1. Lorsqu'un message est reçu chez le serveur, il le transmet à tous les clients connectés, à l'exception de l'expéditeur.



1. Pour tester le service de chat, exécutez d'abord le serveur, puis exécutez le client. Les clients peuvent envoyer et recevoir des messages de chat en temps réel.

