

Travaux Dirigés

Mise à niveau en JAVA

–M1–

Exercice 1: Eclipse

Le but de cet exercice est de se familiariser avec la programmation java dans un environnement de développement intégré. On utilisera comme environnement de développement eclipse. Eclipse est une IDE qui dispose de nombreuses fonctionnalités pour faciliter le développement en Java, c'est l'IDE que nous allons utiliser pendant ce cours.

- Ouvrez une console et lancer eclipse en tapant à partir du prompt : eclipse
- Créez un nouveau projet : file >new >JavaProject:
Une boîte de dialogue apparaît alors: précisez le nom du projet : TD1, choisissez la localisation ou utilisez la localisation par défaut, et enfin cliquez sur finish. Votre projet est alors créé, il porte le nom de test.
- Ouvrez le projet, un répertoire src apparaît. Avec un clic droit sur le répertoire src, choisissez: New/Class. Une boîte de dialogue apparaît alors. Précisez le nom de la classe : App1. Cliquez sur finish.
- Un fichier App1.java apparaît alors. Le modifier pour écrire : Bonjour à tous
- Compilez en cliquant sur la flèche verte et en sélectionnant : Run As > Java Application. Vous devez alors voir apparaître le message Bonjour à tous dans la fenêtre output.

Exercice 2: Visibilité

Considérons cinq classes A, B, C, D et E tel que A, B et C sont dans le même package **package1** et les classes D et E sont mis dans un autre package **package2**. Les classes B et D héritent de la classe A.

```

1 package package1;
2 public class A {
3     private int champPrive;
4     int champSansModificateur;
5     protected int champProtege;
6     public int champPublique;
7 }

```

- Complétez le tableau ci-dessous en cochant les cases pour lesquelles les variables d'instance de la classe **A** sont visibles.

	Classe A	Classe B	Classe C	Classe D	Classe E
champPrive					
champSansModificateur					
champProtege					
champPublique					

- Si la classe A n'était pas déclarée public, est ce que cela change la visibilité des variables?

Exercice 3: Classe Java

Soit la température T. Ecrire une classe qui affiche:

- froid si $T < 8$
- frais si $8 < T < 17$
- bon si $17 < T < 25$ chaud sinon

Exercice 4: Constructeurs

Ecrivez une classe Rationnel qui définit les nombres rationnels.

La classe a les attributs privés suivants :

- numérateur : Le numérateur;
- dénominateur : Le dénominateur.

La classe Rationnel doit disposer des constructeurs suivants :

- Rationnel();
- Rationnel(int numérateur, int dénominateur);
- Rationnel(Rationnel r).

Elle doit aussi contenir les méthodes :

- void additionner(Rationnel r) : ajout du Rationnel en paramètre;
- void soustraire(Rationnel r) : soustraction du Rationnel en paramètre;
- void multiplier(Rationnel r) : multiplication avec le Rationnel en paramètre;
- void diviser(Rationnel r) : division par Rationnel en paramètre;
- double evaluer() : renvoie le résultat sous la forme d'un nombre réel;
- Rationnel inverser() : renvoie le Rationnel inverse ; void afficher() : affiche une représentation du rationnel (a/b).

Exercice 5:

Le but de cet exercice est de créer une classe Complexe ($z = a+ib$). Cette classe va nous permettre de manipuler des nombres complexes.

- Déclarez la classe Complexe dans un fichier constituée de ses paramètres a et b, ainsi que de son constructeur par défaut.
- Implémentez le constructeur par défaut en attribuant à a et b la valeur 0.
- Ajoutez un constructeur permettant d'initialiser les valeurs de a et b (pouvant être différents de 0) en les passant comme paramètres.
- Définissez une fonction qui calcule la somme de deux nombres complexes.
- Définissez une fonction qui détermine le produit de deux nombres complexes.