



Le but du TP est de bien comprendre les mécanismes d’affichage et de saisie, le concept de variables et leur type et les conditionnelles (**if-then-else**).

Pour ce TP, vous devez partir du fichier `INF1_TP1.zip`, présent sur Moodle. Ce fichier est une archive qui contient les squelettes des exercices du TP1.

Pour programmer et compiler les fichiers `.java`, on utilisera l’environnement de développement **Eclipse**. Pour utiliser Eclipse, référez-vous au *Tuto Eclipse pour les TP* disponible sur Moodle.

Pour commencer à travailler, démarrez l’application Eclipse, et importez les squelettes du TP1 dans Eclipse. Puis, commencez avec l’Exercice 1, en éditant avec Eclipse le fichier `Exe1.java` présent dans le paquet `tp1` du répertoire `INF1_TP1/src`.

### Exercice 1 : Hello World !

Écrire le programme qui affiche `Hello World` ! Puis, exécuter le programme (après l’avoir compilé), et contrôler son bon fonctionnement.

### Exercice 2 : Saisie, opérations de base et conditionnelles

Écrire un programme qui demande à l’utilisateur deux nombres entiers, et affecte ces deux valeurs à deux variables `a` et `b`. Ensuite, le programme doit :

- afficher les valeurs de `a` et `b`,
- échanger les valeurs de `a` et `b`,
- afficher les nouvelles valeurs de `a` et `b`,
- afficher la valeur du double de `a`,
- afficher la valeur de la moitié de `b`,
- afficher le quotient et le reste de la division entière de `a` par `b`,
- afficher la plus grande valeur entre `a` et `b`.

**Remarques :** une valeur de type `java.util.Scanner` sert à lire des informations, par exemple à partir de l’entrée standard `System.in`, qui correspond en général au clavier. On peut utiliser plusieurs fois le même scanner pour lire des entiers en faisant des appels successifs à `nextInt()`.

- Vous n’avez pas besoin de créer un nouveau scanner à chaque fois que vous devez lire une valeur fournie par l’utilisateur ; réutilisez un scanner existant. Depuis l’entrée standard, un seul et même scanner doit être utilisé.
- N’oubliez pas de fermer le scanner avec la fonction `close()` dès que vous savez que vous n’en aurez plus besoin jusqu’à la fin de l’exécution de votre programme.

### Exercice 3 : Année bissextile

Une année est *bissextile* lorsqu'elle contient 366 jours au lieu de 365 jours. Pour déterminer si une année est bissextile, on utilise la règle suivante. Une année est *bissextile*

- si elle est divisible par 4 et non divisible par 100, ou
- si elle est divisible par 400.

Sinon elle ne l'est pas.

Écrire un programme qui demande à l'utilisateur de saisir un entier, et affiche si oui ou non cet entier correspond à une année bissextile. Exemples : l'année 2022 n'est pas bissextile, mais l'année 2016 est bissextile.

### Exercice 4 : Parité d'une somme

Écrire un programme qui demande à l'utilisateur de saisir 3 entiers, et affiche si leur somme est paire ou impaire. Par exemple,

- si l'utilisateur saisit les trois nombres 5, 5 et 5, on affichera  
La somme des trois entiers est impaire.
- si l'utilisateur saisit les trois nombres 5, 6 et 5, on affichera  
La somme des trois entiers est paire.

### Exercice 5 : Affichage dans l'ordre croissant

Écrire un programme qui affiche dans l'ordre croissant les valeurs de trois entiers saisies par l'utilisateur en utilisant des conditionnelles. Par exemple, si l'utilisateur saisit les trois nombres 42, 57 et 36, on affichera

36 <= 42 <= 57

### Exercice 6 : Conversion secondes

Écrire un programme qui demande à l'utilisateur un nombre entier de secondes. Le programme doit ensuite donner le nombre correspondant de jours, heures, minutes et secondes. Par exemple, si l'utilisateur saisit 654321, on affichera quelque chose du type

Cela donne 7 jours 13 heures 45 minutes 21 secondes