

TP 3 - JAVA



[Réponses Quizz](#)

[Exercices de programmation](#)

[Parole Chanson](#)

[Dessin table et chaises](#)

[Triangle](#)

[Correction de bugs](#)

[Cas Pratique](#)

[Cas problème](#)

La problématique : L'enjeu de ce TP est de se familiariser avec le langage de programmation Java est pouvoir être capable d'exécuter des commandes simples.

Réponses Quizz

1. Le langage machine le plus basique niveau circuit est: **le langage machine**
2. Les langages qui permettent d'utiliser un vocabulaire qui utilise les termes : read, write ou add sont : **haut niveau**
3. les règles du langage de programmation constituent : **la syntaxe**
4. Un **compilateur** traduit les instructions de langage de haut niveau en code machine
5. Les emplacements de mémoire nommés de l'ordinateur sont appelés : **variables**
6. Les opérations individuelles utilisées dans un programme informatique sont souvent regroupées en unités logiques appelées : **procédures**
7. Une instance de classe est : **un objet**
8. Java a une architecture : **neutre**
9. Vous devez compiler les classes écrites en Java dans : **un bytecode**
10. Toutes les instructions de programmation Java doivent se terminer par : **un point-virgule**

Exercices de programmation

1. Chacun des identifiants de classe suivant est-il légal, légal et conventionnel ou illégal ?
 - a. maClasse : légal
 - b. void : illégal
 - c. Golden Retriever : illégal
 - d. invoice# : illégal et conventionnel
 - e. 36535CodePostal : illégal
 - f. Appartement : légal
 - g. Fruit h. 8888 : illégal
 - i. EcranTotal() : légal et conventionnel
 - j. Acompte_recevable : illégal

2. Chacun des identifiants de la méthode suivant est-il légal, légal et conventionnel ou illégal?

- a. associationRoles() : légal et conventionnel
- b. void() : légal et conventionnel
- c. Golden Retriever() : illégal
- d. invoice#() : illégal
- e. 24500CodePostal() : illégal
- f. PayrollApp() : légal et conventionnel
- g. getReady() : légal
- h. 911() : illégal
- i. EcranTotal() : légal et conventionnel
- j. Acompte_Recevable() illégal

2 - Parole de Chanson

Parole Chanson

Code Java :

```
package tp3;

public class ParoleChanson {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        {
            System.out.println("This is thicker than metals, these conversations");
            System.out.println("These things money cannot buy, this should be enough for you");
            System.out.println("My darlin', I believe in real life, time is to have fun");
            System.out.println("Let's take this time to get high, leave confirmations");
            System.out.println("Uh, 'cause");
        }
    }
}
```

Résultat :

```
This is thicker than metals, these conversations
These things money cannot buy, this should be enough for you
My darlin', I believe in real life, time is to have fun
Let's take this time to get high, leave confirmations
Uh, 'cause
```

Dessin table et chaises

Code Java :

```

package tP3;

public class TableEtChaise {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        System.out.println("X                X");
        System.out.println("X                X");
        System.out.println("X      XXXXXXXXXX X");
        System.out.println("XXXXX X      X XXXXX");
        System.out.println("X  X  X      X  X  X");
        System.out.println("X  X  X      X  X  X");
    }
}

```

Résultat :

```

<terminated> TableEtChaise [Java Ap
X                X
X                X
X      XXXXXXXXXX X
XXXXX X      X XXXXX
X  X  X      X  X  X
X  X  X      X  X  X

```

Triangle

Code Java :

```

package tP3;

public class triangle {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        System.out.println("    T    ");
        System.out.println("   TTT   ");
        System.out.println("  TTTT  ");
        System.out.println(" TTTTTT ");
        System.out.println("TTTTTTTT ");
        System.out.println("TTTTTTTTT ");
        System.out.println("TTTTTTTTTT ");
        System.out.println("TTTTTTTTTTTT");
    }
}

```

Résultat :

```
Problems @ Java
<terminated> triangle [

  T
  TTT
  TTTT
  TTTTT
  TTTTTT
  TTTTTTT
  TTTTTTTT
  TTTTTTTTT
  TTTTTTTTTT
```

Correction de bugs

Avant correction FixBug1 :

Il manque des points entre "system out print", il manque également des guillemets au "salut" et un point figure à la place d'un point à la fin du programme.

À cause de cela le programme ne peut donc pas s'exécuter correctement.

```
1 public class Debug1
2
3     /* This program displays a greeting */
4     public static void main(String[] args)
5     {
6         Systemoutprintln("Salut).
7     }
```

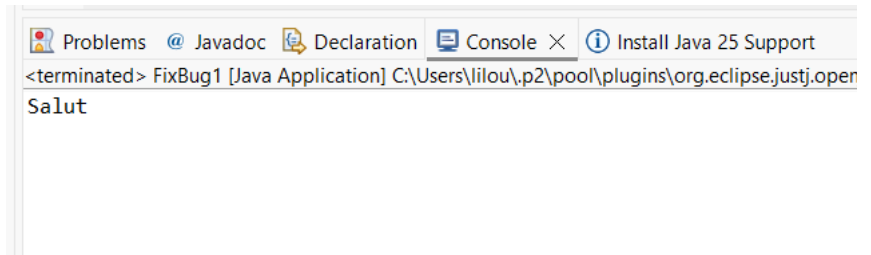
Après correction FixBug1:

Nous avons donc rajouté des points pour la commande "system out print", nous avons également ajouté des

```
1 package fixbug;
2
3 public class FixBug1 {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         System.out.print("Salut");
8
9     }
10
11 }
12
```

guillemets au mot "salut" et un point virgule à la place d'un point à la fin du programme.

Le résultat est l'affichage du mot "Salut" dans la console.
Désormais le programme est en état de fonctionner.



```
Problems @ Javadoc Declaration Console X Install Java 25 Support
<terminated> FixBug1 [Java Application] C:\Users\lilou\p2\pool\plugins\org.eclipse.justj.open
Salut
```

Avant correction FixBug2 :

Le programme ne peut pas fonctionner car /* lui attribue un rôle de commentaire. De plus le quatrième "system out print" contient une faute de frappe.

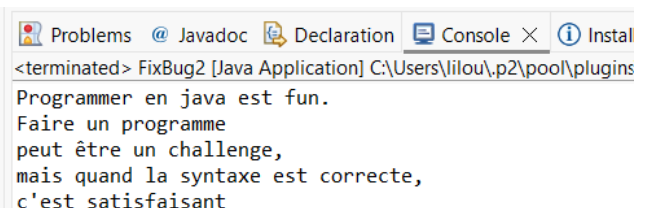
```
public class Debug2
{
    /* This program displays some output
    public static void main(String args)
    {
        System.out.println("Programmer en java est fun.");
        System.out.println("Faire un programme");
        System.out.println("peut être un challenge,");
        System.out.prnitln("mais quand la syntaxe est correcte,");
        System.out.println("c'est satisfaisant");
    }
}
```

Après correction FixBug2 :

Nous avons retiré /* et cela s'affiche désormais comme un code normal. De plus nous avons corrigés l'erreur de frappe.

Le programme affiche donc le bon texte dans la console, le code est donc bon.

```
1 package fixbug;
2
3 public class FixBug2 {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         System.out.println("Programmer en java est fun.");
8         System.out.println("Faire un programme");
9         System.out.println("peut être un challenge,");
10        System.out.println("mais quand la syntaxe est correcte,");
11        System.out.println("c'est satisfaisant");
12    }
13 }
14
15 }
```



```
Problems @ Javadoc Declaration Console X Instal
<terminated> FixBug2 [Java Application] C:\Users\lilou\p2\pool\plugins
Programmer en java est fun.
Faire un programme
peut être un challenge,
mais quand la syntaxe est correcte,
c'est satisfaisant
```

Avant correction FixBug3 :

Nous pouvons observer des erreurs dans le code tel que le premier "system out print" ayant un "O" majuscule. Le deuxième ayant un S minuscule et le troisième ayant un zéro à la place d'un "o".

```
1 public class Debug3
2 {
3     public static void main(String[] args)
4     {
5         System.Out.println("Derrière la rivière");
6         system.out.println("et au dela du bois");
7         SysTem.Out.println("à la maison du garde nous irons");
8     }
9 }
```

Après correction FixBug3 :

Après la corrections des erreurs précédentes, nous pouvons constater la bonne exécution du programme.

```
1 package fixbug;
2
3 public class FixBug3 {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         System.out.println("Derrière la rivière");
8         System.out.println("et au dela du bois");
9         System.out.println("à la maison du garde nous irons");
10    }
11 }
12
13 }
```

Avant correction FixBug4 :

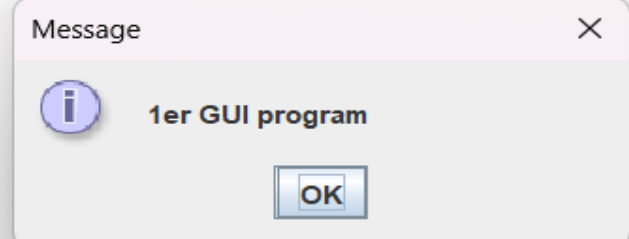
Nous pouvons remarquer qu'il manque le mot "void" dans "public static main". Il manque également des guillemets à "1er GUI program", et que c'est une virgule et non un point d'exclamation qui doit se trouver derrière la dernière parenthèse.

```
1 import javax.swing.JOptionPane;
2 public class Debug4
3 {
4     public static main(String[] args)
5     {
6         JOptionPane.showMessageDialog(null, 1er GUI program)!
7     }
8 }
```

Après correction FixBug4 :

Après avoir ajouté "void", les guillemets et le point virgule une boîte de dialogue s'ouvre et inscrit le texte que nous désirons.

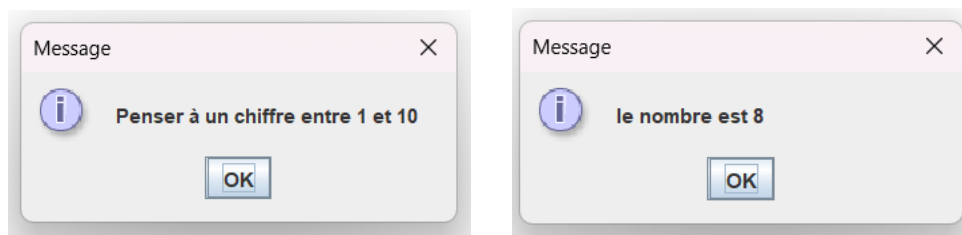
```
1 package fixbug;
2
3 import javax.swing.JOptionPane;
4
5 public class FixBug4 {
6
7     public static void main(String[] args) {
8         JOptionPane.showMessageDialog(null, "1er GUI program");
9     }
10 }
11
12 }
```



Cas Pratique

Pour afficher deux boîtes de dialogue en séquence il faut tout d'abord utiliser deux fois `JOptionPane.showMessageDialog(null, "");`
On écrira dans le premier "JOptionPane" à l'utilisateur de penser à un chiffre, il n'aura plus qu'à appuyer sur OK et un chiffre aléatoire entre 1 et 10 s'affichera. (écrit dans dans le deuxième "JOptionPane")

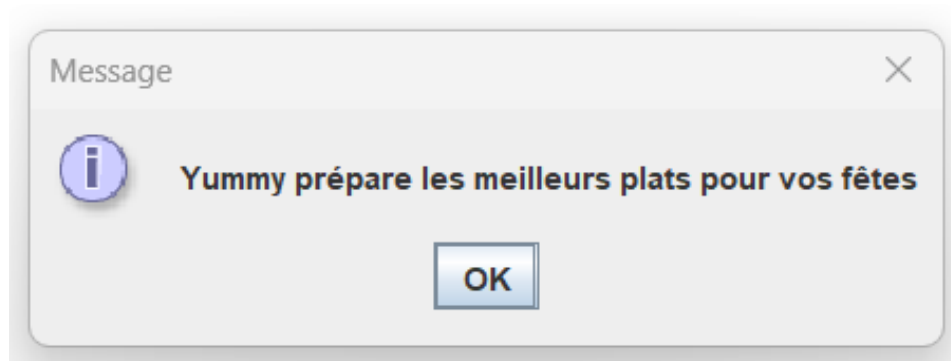
```
package fixbug;  
  
import javax.swing.JOptionPane;  
  
public class Jeu {  
  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        JOptionPane.showMessageDialog(null, "Penser à un chiffre entre 1 et 10");  
        JOptionPane.showMessageDialog(null, "le nombre est " + (1+(int)(Math.random()*10)));  
    }  
}
```



Cas problème

1.a -

```
1 package yummy.yummy;  
2  
3 import javax.swing.JOptionPane;  
4  
5 public class Yummy {  
6  
7     public static void main(String[] args) {  
8         // TODO Auto-generated method stub  
9         JOptionPane.showMessageDialog(null, "Yummy prépare les meilleurs plats pour vos fêtes");  
10  
11     }  
12  
13 }
```



1.b -

```

1 package yummy.yummy;
2
3 public class Yummy2 {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         System.out.println("*****");
8         System.out.println(" *   Yummy prépare les meilleurs   *");
9         System.out.println(" *   plats pour vos fêtes       *");
10        System.out.println("*****");
11    }
12
13 }
14

```

```

*****
*   Yummy prépare les meilleurs   *
*   plats pour vos fêtes       *
*****

```

2-

```

1 package yummy.yummy;
2
3 public class YummyFun {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         System.out.println("SsSsSsSsSsSsSsSsSsSsSsSsSsSsSsSsS");
8         System.out.println("Ss                                     sS");
9         System.out.println("Ss   Yummy prépare les meilleurs   sS");
10        System.out.println("Ss   plats pour vos fêtes       sS");
11        System.out.println("Ss                                     sS");
12        System.out.println("SsSsSsSsSsSsSsSsSsSsSsSsSsSsSsSsS");
13
14    }
15

```

```

SsSsSsSsSsSsSsSsSsSsSsSsSsSsSsSsS
Ss                                     sS
Ss   Yummy prépare les meilleurs   sS
Ss   plats pour vos fêtes       sS
Ss                                     sS
SsSsSsSsSsSsSsSsSsSsSsSsSsSsSsSsS

```

Conclusion :

Ce TP s'est avéré très utile et de plus nous avons pu voir que même avec un faible niveau il est tout de même possible de créer de simples dessins en Java. J'ai trouvé ce TP assez ludique et plaisant à faire.