

17/09/2025

TP03

Exercice Java

Table des matières

Introduction :	1
I. Réalisation théorie en pratique	1
1. Moyenne	1
2. Âge	2
II. Réalisation objectif suivant du TP	3
3. Dessin de la chaise et table en Java	3
4. Dessin pyramide en Java	4
III. Les debugs Java	4
IV. QCM Java	6
V. Exercice programmation	7
VI. Test	8
Conclusion	8

Fait par : Mattéo Mouranchon – Groupe 2

Compte-rendu numéro 7

Introduction :

Ce TP avait pour objectif de nous initier à la suite des bases de la programmation Java, en abordant des concepts clés comme les tableaux, les structures conditionnelles, la correction de bugs et la création d'applications interactives. Les exercices proposés couvraient plusieurs aspects l'algorithmique, la correction de bugs, et le développement d'applications interactives. Chaque partie visait à renforcer une compétence spécifique, tout en nous familiarisant avec l'environnement de développement Eclipse, spécialement pour aider sur le langage JAVA

I. Réalisation théorie en pratique

Certains exercice du TP été déjà vu en cours de théorie comme la moyenne de 4 note et l'âge pour calculer si la personne était mineure, majeur ou vétéran.

1. Moyenne

Cette exercice consiste à mesurer 4 notes en une note moyenne, déjà appris en théorie, on la reproduit en pratique avec le langage Java avec l'image ci-dessous.

```
package tp.java;
import java.util.Scanner;
public class MoyenneNotes {
    public static void main(String[] args) {
        Scanner clavier = new Scanner(System.in);
        double[] note = new double[4];
        System.out.print("Saisir la note 1 : ");
        note[0] = clavier.nextDouble();
        System.out.print("Saisir la note 2 : ");
        note[1] = clavier.nextDouble();
        System.out.print("Saisir la note 3 : ");
        note[2] = clavier.nextDouble();
        System.out.print("Saisir la note 4 : ");
        note[3] = clavier.nextDouble();
        double moyenne = (note[0] + note[1] + note[2] + note[3]) / 4;
        System.out.println("La moyenne des 4 notes est : " + moyenne);
        clavier.close();
    }
}
```

Dans le code, on retrouve ce que nous avons appris et le tout, écrit en langage Java. Le code va noter les notes 1,2,3 et 4 puis diviser le tout par 4 pour obtenir une moyenne.

Résultat à tout ceci :

```
Saisir la note 1 : 15
Saisir la note 2 : 20
Saisir la note 3 : 35
Saisir la note 4 : 45
La moyenne des 4 notes est : 28.75
```

Le code fonctionne, on a juste à entrer un nombre au hasard aux 4 notes puis diviser par 4 pour obtenir une moyenne.

2. Âge

Cette exercice consiste à comparer l'âge de l'utilisateur par des nombre prédéfini, déjà appris en théorie, on la reproduit en pratique avec le langage Java avec l'image ci-dessous.

```
1 package tp.java;
2
3 import java.util.Scanner;
4
5 public class age {
6     public static void main(String[] args) {
7         Scanner clavier = new Scanner(System.in);
8
9         int age;
10
11         System.out.print("Entrez votre age : ");
12
13         age = clavier.nextInt();
14
15         if (age < 18) {
16             System.out.println("Vous etes mineur");
17         }
18         else if (age > 40) {
19             System.out.println("Vous etes vétéran");
20         }
21         else {
22             System.out.println("Vous etes senior");
23         }
24     }
25     clavier.close();
26 }
27 }
```

Dans ce code, on a juste à entrer un âge et le script va comparer les limites d'âge fixer et va les comparer. Si l'âge entré est inférieur à 18, la personne est mineure, s'il est entre 18 et 40 ans, la personne est senior et supérieure à 50, la personne est vétéran.

Si la personne à 12 ans :

```
Entrez votre age : 12
Vous etes mineur
```

Si la personne à 46 ans :

```
Entrez votre age : 46
Vous etes vétéran
```

Si la personne à 18 ans :

```
Entrez votre age : 18
Vous etes senior
```

II. Réalisation objectif suivant du TP

La suite du TP consiste à résoudre des questionnaires, parvenir à codées du script complexe et à remplir un petit QCM sur les connaissance de Java.

Classe parole chanson java

Script :

```
1 package tp.java;
2
3 public class ParoleChanson {
4     public static void main(String[] args) {
5         System.out.println("ligne 1 des paroles");
6         System.out.println("ligne 2 des paroles");
7         System.out.println("ligne 3 des paroles");
8         System.out.println("ligne 4 des paroles");
9     }
10 }
```

Résultat :

```
ligne 1 des paroles
ligne 2 des paroles
ligne 3 des paroles
ligne 4 des paroles
```

3. Dessin de la chaise et table en Java

Script :

```
1 package tp.java;
2 public class TableEtChaises {
3     public static void main(String[] args) {
4         System.out.println("X                X");
5         System.out.println("X                X");
6         System.out.println("X      XXXXXXXXXXXX      X");
7         System.out.println("XXXX X                X XXXX");
8         System.out.println("X X X                X X X");
9         System.out.println("X X X                X X X");
10        System.out.println("X X X                X X X");
11    }
12 }
```

Resultat :

```
X                X
X                X
X      XXXXXXXXXXXX      X
XXXX X                X XXXX
X X X                X X X
X X X                X X X
X X X                X X X
```

4. Dessin pyramide en Java

Script :

```
1 package tp.java;
2 public class Triangle {
3     public static void main(String[] args) {
4         System.out.println("  T  ");
5         System.out.println("   TTT ");
6         System.out.println("  TTTT ");
7         System.out.println(" TTTTTT");
8     }
9 }
```

Résultat :

```
<terminated>
  T
   TTT
  TTTT
 TTTTTT
```

III. Les debugs Java

1. Debug 1

Erreur du script :

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

Après avoir résolue :

```
1 package ex3;
2
3 public class code {
4
5     public static void main(String[] args)
6     {
7         System.out.println("Salut");
8     }
9 }
```

Message :
« salut »

2. Debug 2

Erreur du script :

```
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 avoir résolue :

```
1 package ex3;
2
3 public class code {
4     public static void main(String[] args)
5     {
6         System.out.println("Programmer en java est fun.");
7         System.out.println("Faire un programme");
8         System.out.println("peut être un challenge,");
9         System.out.println("mais quand la syntaxe est correcte,");
10        System.out.println("c'est satisfaisant");
11    }
12 }
```

Message du script :

```
Programmer en java est fun.
Faire un programme
peut être un challenge,
mais quand la syntaxe est correcte,
c'est satisfaisant
```

3. Debug 3

Erreur du script :

```
public class Debug33
{
    public static void main(String[] args)
    {
        System.Out.println("Derrière la rivière");
        system.out.println("et au dela du bois");
        SysTem.Out.println("à la maison du garde nous irons");
    }
}
```

Après avoir résolue :

```
1 package ex3;
2
3 public class code {
4     public static void main(String[] args)
5     {
6         System.out.println("Derrière la rivière");
7         System.out.println("et au dela du bois");
8         System.out.println("à la maison du garde nous irons");
9     }
10 }
```

Message du script :

```
Derrière la rivière
et au dela du bois
à la maison du garde nous irons
```

4. Debug 4

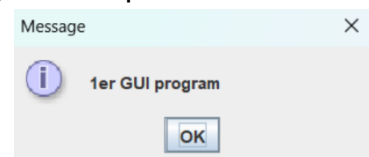
Erreur du script :

```
1 package tp.java;
2 import javax.swing.JOptionPane;
3 public class Debug4
4 {
5     public static main(String[] args)
6     {
7         JOptionPane.showMessageDialog(null, 1er GUI program)!
8     }
9 }
```

Après avoir résolue :

```
package tp.java;
import javax.swing.JOptionPane;
public class FixBug4
public static void main(String[] args) {
JOptionPane.showMessageDialog(null, "1er GUI program");
}
```

Message du script :



IV. QCM Java

- 1- Le langage machine le plus basique niveau circuit est?
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 _____ traduit les instructions de langage de haut niveau en code machine
Un compilateur
- 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 :
Une 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

V. Exercice 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**
 - e. 36535CodePostal – **Illégal**
 - f. Apartment – **Légal**
 - g. Fruit – **Légal**
 - h. 8888 – **Illégal**
 - i. Ecran Total() – **Illégal**
 - j. Acompte Recevable - **Légal**

2. Chacun des identifiants de méthode suivant est-il légal, légal et conventionnel ou illégal ?
 - a. AssociationRoles() - **Légal**
 - b. Void() - **Illégal**
 - c. Golden Retriever() - **Illégal**
 - d. Invoice#() - **Illégal**
 - e. 24500CodePostal() - **Illégal**
 - f. PayrollApp() - **Légal**
 - g. GetReady() - **Légal**
 - h. 911() - **Légal**
 - i. EcranTotal() - **Légal**
 - j. Acompte_Recevable() - **Légal**

VI. Test

Le TP à toujours été une phase difficile avec délicate, avec plusieurs échecs au départ. Tout ceci n'a jamais été une partir de plaisir à force toujours recommencer et ne pas savoir la cause des erreur mais finalement avec un peu de temps, min, heure et jours, je parviens à légèrement comprendre.

Conclusion

Le TP m'a permis de passer de la **théorie à la pratique** : manipuler les **conditions**, les boucles, la lecture/affichage, et une petite interaction graphique, tout en apprenant une méthode de travail efficace dans Eclipse (nom du fichier = nom de la classe publique, encodage UTF-8, exécutions fréquentes). La stratégie « **petits pas + tests rapides** » m'a aidé à corriger vite les erreurs (accolades, guillemets, System.out.println, bornes) et à obtenir des sorties **strictement conformes**. Pour aller plus loin, je pourrais paramétrer davantage les programmes (ex. hauteur de la pyramide, nombre de notes), ajouter une validation d'entrées et introduire quelques tests automatisés (capture de System.out, ou JUnit) afin de fixer les résultats attendus et gagner encore en fiabilité.