Programmation avancée Java

TP-6

Gherghouch Abdessalam

Exercice N1: autentification Partie 1

```java
package Inputs;
import java.io.*;
import java.util.Scanner;
public class Autentification {

//login verification
public static int Search(String [] info,BufferedReader buf) throws IOException {
  String str;
  int valide=0;
  String wrap[];
  while((str=buf.readLine()) != null) {
   wrap=str.split("/",-2);
   if(info[0].equals(wrap[0])&&info[1].equals(wrap[1])){
//
System.out.println(info[0].equals(wrap[0])&&info[1].equals(wrap[1])&&wrap[2].equals("1"));
    if(info[0].equals(wrap[0])&&info[1].equals(wrap[1])&&wrap[2].equals("1")) {
     valide=1;
    // System.out.println(1);
    }
    if(info[0].equals(wrap[0])&&info[1].equals(wrap[1])&&wrap[2].equals("0")) {
     valide=2;
    // System.out.println("admin");
    }
    break;
   }
  }
  return valide;
}

//exit
public static void exit() throws IOException {
 System.out.println("\tTo exit( type: 0 )");
 Scanner key = new Scanner(System.in);
 int value=0;
 try {
 value=key.nextInt();
 }catch(Exception e) {
  e.printStackTrace();
 }
 if(value==0) main();
 else exit();
}
```

Exercice N1: autentification Partie 2

```java
//user
// main menu
public static int menu() {
 int choice = 0;
 Scanner input =new Scanner(System.in);
 System.out.println("\t \t Autntification\n \t FOR USER (TYPE : 1)\t FOR ADMIN (TYPE :2)");
 try {
  choice=input.nextInt();
 }
 catch(Exception e) {
  System.out.println("Please enter a valide value :) ");
 }
 return choice ;
}

//login => return the login and the password
public static String[] login() {
 Scanner key = new Scanner(System.in);
 String login = null,pwd = null;
 try {
  System.out.println("Enter your login");
  login=key.nextLine();
  System.out.println("Enter Your password");
  pwd=key.nextLine();
 }catch(Exception e) {
  e.printStackTrace();
 }
 String info [] = {login,pwd};
 return info;
}

// route of all operations
public static void main() throws IOException {
 int userType = 0;
 do {
  userType=menu();

  if(userType==1) User();
  if(userType==2) Admin();
 }while(userType!=1 || userType!=2);
}
```

Exercice N1: autentification Partie 3

```java
//user operations
 public static void User() throws IOException {
  String inf[];
  File Log= new File("login.txt");
  FileReader l=new FileReader(Log);
  BufferedReader buf=new BufferedReader(l);

   System.out.println("\t USER");
   inf=login();
   int result=Search(inf,buf);
//   System.out.println(result);
   if(result==0) {
    System.out.println("INVALID Login OR PASSWORD :( !") ;
    main();}
   if(result==1) {
    System.out.println("Login successfully :( !") ;
    exit();}
   if(result==2) {
    System.out.println("ERROR 404 :(");
    main();
    }
 }
 //admin
 //display admin's menu
 public static void adminMenu() throws IOException {
  File Log= new File("login.txt");
  FileReader l=new FileReader(Log);
  BufferedReader buf=new BufferedReader(l);
  int choice = 0;
  System.out.println("\t \t DASHBOARD\n \t ADD A USER (TYPE : 1)\t LISTE OF USERS (TYPE
:2) \t  exit(TYPE: 4)");
  Scanner input =new Scanner(System.in);
 // System.out.println("\t Operations \n \t Add user (type :1) \t delete a User (type :2)\t
display all users (type:3)");
  try {
   choice=input.nextInt();
  }
  catch(Exception e) {
   e.printStackTrace();
  }
  if (choice==1)
   AddUser();
  if(choice==2)
   ListeUsers(buf);
  if(choice==4)
   exit();
  }
```

Exercice N1: autentification Partie 4

```java
//add A user
 public static void AddUser() throws IOException {
  File file = new File("login.txt");
  FileWriter fr = new FileWriter(file, true);
  BufferedWriter br = new BufferedWriter(fr);
  Scanner key = new Scanner(System.in);
  String Str="\n";
  int type=0;
  System.out.println("ENTER User'S login");
  Str+=key.nextLine()+"/";
  System.out.println("ENTER PASSWORD");
  Str+=key.nextLine()+"/";
  System.out.println("ENTER USER'S TYPE (ADMIN: 0) (NORMAL USER: 1)");
  Str+=key.nextInt();
      br.write(Str);
  br.close();
  fr.close();
  System.out.println("ADDED successfully");
  adminMenu();
 }

 //list of users
 public static void ListeUsers(BufferedReader buf) throws IOException {
  String str="";
  while((str=buf.readLine())!=null){
  System.out.println("login\t"+str.split("/",-2)[O]+"\tpassword   "+str.split("/",-2)[1]);
  }
  adminMenu();
 }
 //admin operations
 public static void Admin() throws IOException {
  File Log= new File("login.txt");
  FileReader l=new FileReader(Log);
  BufferedReader buf=new BufferedReader(l);
  String inf[];
  System.out.println("\t ADMIN");
  inf=login();
  int result=Search(inf,buf);
//  System.out.println(result);
  if(result==0) {
   System.out.println("INVALID Login OR PASSWORD :( !") ;
   main();}
  if(result==2) {
   System.out.println("Login successfully :( !") ;
   adminMenu();}
  if(result==1) {
   System.out.println("ERROR 404 :(");
   main();
  }
 }
```

Exercice N1: autentification Partie 5

```java
public static void main(String[] args) throws IOException {
// TODO Auto-generated method stub
File Log= new File("login.txt");
FileReader l=new FileReader(Log);
BufferedReader buf=new BufferedReader(l);
main();
}
}
```

Login.txt

```
1 abde/1234/1
2 rida/32451/1
3 amine/2341/1
4 imade/233/0
5 a/1/0
6 RACHIDE/1111/0
7 abdeee/1111/1
8 ddos/1212/1
9 yassin /DDOS???/0
10 ahmed/00000/0
11 ddpoooos/12ggg/1
12 ddos/ddos???/1
13 abde/ddos??<?php {{$pwd}}?>/1
14 admin/<?{{$pwd}}?>/1
15 qwe/sd/35345
16 DOOS/DD?????/12
17 abdesslam/<?{{$pwd}}?>/1
18 abde/abdeee<? ?>/1
```

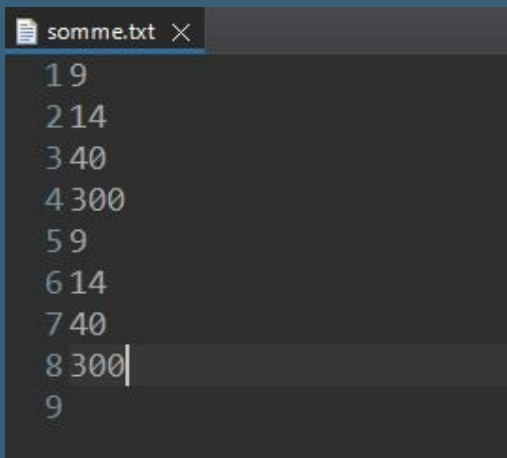## Résultat

```
                  Autntification
          FOR USER (TYPE : 1)     FOR ADMIN (TYPE :2)
1
          USER
Enter your login
abde
Enter Your password
1234
Login successfully :( !
        To exit( type: 0 )
0
                  Autntification
          FOR USER (TYPE : 1)     FOR ADMIN (TYPE :2)
2
          ADMIN
Enter your login
a
Enter Your password
1
Login successfully :( !
                  DASHBOARD
          ADD A USER (TYPE : 1)   LISTE OF USERS (TYPE :2)          exit(TYPE: 4)
1
ENTER User'S login
abde
ENTER PASSWORD
abdeee<? ?>
ENTER USER'S TYPE (ADMIN: 0) (NORMAL USER: 1)
1
ADDED successfully
                  DASHBOARD
          ADD A USER (TYPE : 1)   LISTE OF USERS (TYPE :2)          exit(TYPE: 4)
2
login   abde      password    1234
login   rida      password    32451
login   amine     password    2341
login   imade     password    233
login   a         password    1
login   RACHIDE   password    1111
login   abdeee    password    1111
login   ddos      password    1212
login   yassin    password    DDOS???
login   ahmed     password    00000
login   ddpoooos          password    12ggg
login   ddos      password    ddos???
login   abde      password    ddos??<?php {{$pwd}}?>
login   admin     password    <?{{$pwd}}?>
login   qwe       password    sd
login   DOOS      password    DD?????
login   abdesslam         password    <?{{$pwd}}?>
login   abde      password    abdeee<? ?>
                  DASHBOARD
          ADD A USER (TYPE : 1)   LISTE OF USERS (TYPE :2)          exit(TYPE: 4)
4
        To exit( type: 0 )
0
[
                  Autntification
          FOR USER (TYPE : 1)     FOR ADMIN (TYPE :2)
```
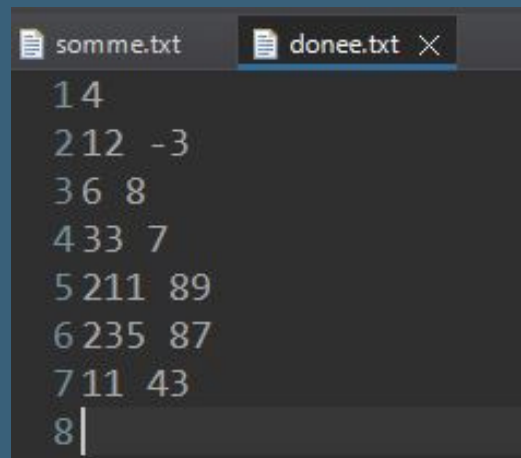
Exercice N2: permutation

```java
package donee;
import java.io.*;
public class Permute {
 public static void main(String[] args) throws IOException {
  // TODO Auto-generated method stub
  File file = new File("somme.txt");
  FileWriter fr = new FileWriter(file, true);
  try (BufferedWriter br = new BufferedWriter(fr)) {
   BufferedReader buf=new BufferedReader(new FileReader(new File("donee.txt")));
   String CurrentLine="";
   String struct[];
   String switchTable=""; ;
   while((CurrentLine=buf.readLine())!=null) {
   struct=CurrentLine.split(" ",-2);
   for(int i=0;i<struct.length;i++) {
    switchTable+=Integer.parseInt(struct[i])+" ";
   }
   //System.out.println(switchTable);
   }
   struct=switchTable.split(" ",-2);

   int tmpTable[]=new int [Integer.parseInt(struct[0])];
   for(int i=1,j=0;i<Integer.parseInt(struct[0])*2+1 && j<tmpTable.length;i+=2,j++) {
    //mettre la somme de couple dans une meme case
    tmpTable[j]=Integer.parseInt(struct[i])+Integer.parseInt(struct[i+1]);
    br.write(tmpTable[j]+"\n");
    //System.out.println(j+"||"+tmpTable[j]);
   }
   br.close();
  } catch (NumberFormatException e) { // TODO Auto-generated catch block
   e.printStackTrace();
  }
 }
}
```

somme.txt ✕

```
19
214
340
4300
59
614
740
8300
9
```

somme.txt   donee.txt ✕

```
14
212 -3
36 8
433 7
5211 89
6235 87
711 43
8
```

Exercice N3: contrôle d'affichage

```java
package controle;
import java.io.*;
import java.util.Scanner;
public class ControleFichier {
 public static void main(String[] args) throws  IOException {
 // TODO Auto-generated method stub
 Scanner key=new Scanner(System.in);
 System.out.println("Donner le nom du fichier a lister:");
 String file="";
 try {
  file=key.nextLine();
 }catch(Exception E) {
  E.getMessage();
 }
 System.out.println(file);

 File f=new File(file);
 //verifier l'existence de fichier
 if(f.exists()) {
  BufferedReader buf=new BufferedReader(new FileReader(new File(file)),22);
 String CurrentLine="";
 int i=0;
 while((CurrentLine=buf.readLine())!=null) {
  if(CurrentLine.length()<61) {
   System.out.println(i+1+"  "+CurrentLine);
   // System.out.println("longueur :"+CurrentLine.length());
  }
  else {
   //deviser la ligne qui passe 60 characteres en 2 ligne
   System.out.println(i+1+" "+CurrentLine.substring(0, 60));
    System.out.println(CurrentLine.substring(60));
    //System.out.println("longueur :"+CurrentLine.length());
  };
 ++i;
 }
 System.out.println("*** fin liste fichier test.txt ***");
 }else
  System.out.println("Fichier introuvable");
}
}
```

```
Donner le nom du fichier a lister:
text.txt
text.txt
1 Ceci est la première ligne d'un exemple de fichier texte:tex
t.txt
2 Il contient des lignes de chiffres de longueurs variables do
nt une de 59 caracteres, une de 60 caracteres et une de 61 caracteres
3  12345678901234567890
4  123456789012345678901234567890123456789012345678901234567890
5  12345678901234567890123456789012345678901234567890123456789
6 123456789012345678901234567890123456789012345678901234567890
1
7  12345678901234567890123456789012345678901234567890
8  la ligne suivante est vide
9
10  les deux lignes suivantes sont egalement vides
11
12
13  Ceci est la derniere ligne du fichier
*** fin liste fichier test.txt ***
```