**ملف الطالب لمعمل هياكل بيانات**

**Student File For Data Structures Lab**

**Name : Amira Bumadyan Kahtan**

**Specialization : Computer Science**

# واستمارة تقييم ملف الطالب

**إنجاز التكاليف**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **الاسبوع** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **الاجمالي** |
| 1 |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |  |  |  |  |  |
| 11 |  |  |  |  |  |  |  |  |  |  |  |
| 12 |  |  |  |  |  |  |  |  |  |  |  |

**إنجاز تمارين المعمل**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **الاسبوع** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **الاجمالي** |
| **1** |  |  |  |  |  |  |  |  |  |  |  |
| **2** |  |  |  |  |  |  |  |  |  |  |  |
| **3** |  |  |  |  |  |  |  |  |  |  |  |
| **4** |  |  |  |  |  |  |  |  |  |  |  |
| **5** |  |  |  |  |  |  |  |  |  |  |  |
| **6** |  |  |  |  |  |  |  |  |  |  |  |
| **7** |  |  |  |  |  |  |  |  |  |  |  |
| **8** |  |  |  |  |  |  |  |  |  |  |  |
| **9** |  |  |  |  |  |  |  |  |  |  |  |
| **10** |  |  |  |  |  |  |  |  |  |  |  |
| **11** |  |  |  |  |  |  |  |  |  |  |  |
| **12** |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **الاسبوع** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **الاجمالي** |
| **نسبة الإنجاز** |  |  |  |  |  |  |  |  |  |  |  |  |  |

**إنجاز المشروع**

اسم النظام :

**القسم الأول**

نظام تأمين سيارات.

## وصف مختصر للنظام :

البرنامج عبارة عن برنامج مصغر لشركة تأمين سيارات.

يعمل النظام على عمل حساب للمؤمن له في الشركة ,

حيث يتم تخزين بيانات المؤمن له والمؤمن عليه , ومعلومات عن التأمين

- وثيقة التأمين - ,وعن الحساب والمصاريف الصحية والسحب والحماية, وتحديد القسط السنوي وكذلك التحملات الإجبارية ..الخ.

كذلك يقوم النظام بالبحث عن الحساب وعرض معلومات الحساب , التعديل في بيانات العميل ,وحذف الحساب .

كما يقوم بإضافة حادث الى الحساب والقيام بالعمليات اللازمة التابعة لتعويضات الحادث.

## فريق تنفيذ النظام :

* أميرة بومدين محمد قحطان
* فاطمة محمد يحيى الجوبي
* أمة الرحمن لطفي السروري

## الخدمات التي يقدمها النظام :

* اضافة عميل.
* عرض.
* حذف.
* اضافة تعويضات.
* بحث.
* تعديل لبيانات العميل.

**المستخدمين للنظام :**

* الموظفين

## عمل تصور لكل نماذج وواجهات النظام

1. .......واجهة ................................
2. ....... واجهة................................

1. ........ واجهة...............................
2. ....... واجهة................................

**القسم الثاني**

# القسم الثاني

**معمل 1**

**حل التمارين التي تم حلها أثناء المعمل**

**1**

public class HelloWorld {  
 public static void main(String[] args) {  
 System.*out*.println("Hello World !");  
 }  
}

**2**

int i=3;  
System.*out*.println(i);

**3**

int s[]={1,2,3,4};  
System.*out*.println(s);

**4**

for(int l=0;l<s.length;l++)  
{  
 System.*out*.println(s[l]);  
}

**5**

Scanner m=new Scanner(System.*in*);  
  
int k[]=new int[5];  
  
for(int l=0;l<s.length;l++)  
{  
 k[l]=m.nextInt();  
}  
  
for(int l=0;l<s.length;l++)  
{  
 System.*out*.println(k[l]);   
}

**6**

System.*out*.println("[");  
  
for(int l=0;l<s.length;l++)  
{  
 if(l==s.length-1)  
 System.*out*.println(k[l]);  
 else  
 System.*out*.println(k[l]+",");  
}  
  
System.*out*.println("]");

**معمل 2**

**حل التمارين التي تم حلها أثناء المعمل**

1 - New Reverse an Array

public void reverse()  
{  
 int n= a.length-1;  
 for(int i=0;i<a.length/2;i++){  
 int t= a[i];  
 a[i]=a[n];  
 a[n]=t;  
 n--;  
 }  
  
}

2 -Shift Right Array Elements

public void shiftright(){  
 int i ;  
 for (i = a.length-1 ; i>0 ; i--) {  
 a[i]=a[i-1];  
 }  
 a[i]=0;  
}

3 -Shift Left Array Elements

public void shiftleft(){  
 int i ;  
 for (i =0 ; i<a.length-1 ; i++) {  
 a[i]=a[i+1];  
 }  
 a[i]=0;  
}

4-dd Element to Array

public void add(int newE){  
 if(num<s.length){s[num++]=newE;}  
  
 else  
 System.out.println("Array is full!");  
  
}

5-Add Sorted Elements to Array

public void addsort(int newE){  
 if (num<s.length){  
 num++;  
 int i=num-1;  
 while (i>0&&s[i-1]>newE)  
 {  
 s[i]=s[i-1];  
 i--;  
 }  
 s[i]=newE;  
 }  
 else  
 System.*out*.println("Array is full!");  
  
}

**حلول تكاليف المعمل**

1/Write a Java method that Reverse an array using another array.

int []x={1,2,3,4,5};  
int []y=new int[5];  
int num=0;  
  
public void Reverse(){  
 num=x.length-1;  
 for(int i=0; i<x.length;i++){  
 y[num]=x[i];  
 num--;  
 }  
}

array x: array y:

1 5

2 4

3 3

4 2

5 1

2/Write a Java method that Clone an array to a backup array.

public void backup(){  
 for(int i =0;i<x.length;i++){  
 y[i]=x[i];  
 }  
 System.*out*.println("done backup ");  
}

done backup

array x: array y:

1 1

2 2

3 3

4 4

5 5

5

3 -Write a Java method that remove elements from an array.

public void removing(int index){  
 int[] copy = new int[x.length - 1];  
 for (int i = 0, j = 0; i < x.length; i++) {  
 if (i != index) {  
 copy[j++] = x[i];  
 }  
 }  
 System.*out*.println("Done remove");  
 for (int i = 0; i < x.length - 1; i++) {  
 System.*out*.println(copy[i]);  
 }  
 }

Tasks R=new Tasks();

R.removing(3);

Done remove

1

2

3

4 -Write a Java method that repeatedly selects and removes a random entry from an array until the array holds no more entries.

import java.util.Random;

public void removeElements(){  
 Random z= new Random();  
 while (x.length > 0) {  
 int index = z.nextInt(x.length);  
 System.*out*.println("Index = " + index + ", Element = " + a[index]);  
 int[] array = new int[x.length - 1];  
 for (int i = 0; i < index; i++)  
 array[i] = x[i];  
 for (int i = index; i < x.length - 1; i++)  
 array[i] = x[i + 1];  
 x = array;  
 }  
  
 System.*out*.println("Done remove");  
 for (int i = 0; i < x.length - 1; i++) {  
 System.*out*.println(x[i]);  
 }  
}

Index = 1, Element = 2

Index = 2, Element = 4

Index = 0, Element = 1

Index = 0, ElementT = 3

Index = 0, Element = 5

Done remove

5/Push your assignments to your git hub repository

**معمل 3**

**حل التمارين التي تم حلها أثناء المعمل**

-Remove Array Element1

حذف من اليسار

public void remove(int a[],int index){  
 if(index>=0 && index<a.length){  
 int i=index;  
 while (i<a.length-1){  
 a[i]=a[i+1];  
 i++;  
 }  
 a[i]=0;  
 }  
}

2- Insertion Sort

public void insertionsort(int a[]){

for (int i=1;i<a.length;i++){

int cur=a[i];

int j=i;

while(j>0&&cur<a[j-1]) //shifting loop

{

a[j]=a[j-1];

j--;

}

a[j]=cur;

}

}

3-Fill Array With Random Number

import java.util.Random;

public void fillrandom(){  
 Random r=new Random();  
 r.setSeed(System.*currentTimeMillis*());  
 for(int i=0;i<q.length;i++){  
 q[i]=r.nextInt(10);  
 }  
}

**حلول تكاليف المعمل**

3.17

public class T317 {  
  
  
 int num = 1;  
  
 public int repeate(int []y) {  
  
 int index=0;  
 boolean f=false;  
  
  
 for(int j=0;j<y.length;j++) {  
 int i=j+1;  
 while (i < y.length && !f) {   
 if (y[i] == y[j]) {  
 f= true;  
 num++;  
 }  
 i++;  
 }  
 if (f = true){index++;}  
 else {num=0;}  
 }  
 return index;  
 }  
 static int findRepeating(int[] array, int x)  
 {  
 int sum = 0;  
 for (int i = 0; i < x; i++)  
 sum += array[i];  
 return sum - (((x - 1) \* x)/2 );  
 }

public static void main(String[] args) {  
 T317 B=new T317();  
 int[] Z = {2, 2};  
 int[] M = {7,7,7,7,7};  
  
System.*out*.println("there are "+B.num+"of integers are repeated "+B.repeate(Z)+" times in the array:");

System.*out*.println(B.*findRepeating*(Z,Z.length));

3.18

public class T318 {  
 int num = 1;  
  
 public int repeate(int []y) {  
  
 int index=0;  
 boolean f=false;  
  
  
 for(int j=0;j<y.length;j++) {  
 int i=j+1;  
 while (i < y.length && !f) {   
 if (y[i] == y[j]) {  
 f= true;  
 num++;  
 }  
 i++;  
 }  
 if (f = true){index++;}  
 else {num=0;}  
 }  
 return index;  
 }

static int findRepeating(int[] array, int x)  
{  
 int sum = 0;  
 for (int i = 0; i < x; i++)  
 sum += array[i];  
 return sum - (((x - 1) \* x) );  
}

System.*out*.println("there are "+B.num+" of integers are repeated "+B.repeate(M)+" times in the array:");  
  
 System.*out*.println(B.*findRepeating*(M,M.length));  
}

3

public class Game {  
   
 public class Gameentry{  
 private String Name ;  
 private int Score;  
  
 public Gameentry(String Name, int Score) {  
 this.Name = Name;  
 this.Score = Score;  
 }  
  
 public String getName() {  
 return Name;  
 }  
  
 public int getScore() {  
 return Score;  
 }  
  
 @Override  
 public String toString() {  
 return "Gameentry{" +  
 "Name='" + Name + '\'' +  
 ", Score=" + Score +  
 '}';  
 }  
 }  
   
 public class ScorBoard{  
 private int NumEntries =0;  
 private Gameentry[] Board;  
   
 public ScorBoard(int capa){  
 Board=new Gameentry[capa];  
 }  
   
 public void add(Gameentry E){  
 int NewScore =E.getScore();  
 if (NumEntries<Board.length||NewScore>Board[NumEntries-1].getScore()){  
 if(numEntries<board.length)  
 NumEntries++;  
   
 int j=NumEntries-1;  
 while(j>0&&Board[j-1].getScore()<NewScore){  
 Board[j]=Board[j-1];  
 j--;  
 }  
 Board[j]=e;  
 }  
 }  
public Gameentry remove(int i)throws IndexOutOfBoundsException{  
 if(i<0||i>NumEntries)  
throw new IndexOutOfBoundsException("invaled:"+i);  
   
 Gameentry temp =Board[i];  
   
 for(int j=i;j<NumEntries-1;j++){  
 Board[j]=Board[j+1];  
 }  
 Board[NumEntries-1]=null;  
 NumEntries--;  
 return temp;  
 }   
 }   
}

**معمل 4**

**حل التمارين التي تم حلها أثناء المعمل**

**حلول تكاليف المعمل**

1-

|  |  |
| --- | --- |
|  | import java.util.ArrayList;  import java.util.Arrays; |
|  | public class Merge { |
|  | public static void main(String[] args) { |
|  | ArrayList<Object> list1 = new ArrayList<>(); |
|  | list1.add("Hi"); |
|  | list1.add("You"); |
|  | list1.add("There"); |
|  | System.out.println("ArrayList 1 : "+list1); |
|  |  |
|  | ArrayList<Object> list2 = new ArrayList<>(); |
|  | list2.add(1); |
|  | list2.add(2); |
|  | list2.add(3); |
|  | System.out.println("ArrayList 2 : "+list2); |
|  |  |
|  | list1.addAll(list2); |
|  | System.out.println("Joind Array Lists : "+list1); |
|  | } |
|  | } |

2-

|  |  |
| --- | --- |
|  | import java.util.Arrays;  import java.util.ArrayList; |
|  | import java.util.List; |
|  |  |
|  | public class Rev { |
|  | public static void main(String[] args) { |
|  | List colors = new ArrayList(Arrays.asList(1,2,3,4,5)); |
|  | System.out.println("Before reversing : "+colors); |
|  | int j = colors.size()-1; |
|  | for (int i = 0 ; i <j ; i++) { |
|  | colors.add(i, colors.remove(j)); |
|  | } |
|  | System.out.println("After reversing : "+ colors); |
|  |  |
|  |  |
|  | } |
|  |  |
|  | } |

**معمل 5**

**معمل 6**

**معمل 7**

**معمل 8**

**معمل 9**

**معمل 10**

**معمل 11**

**معمل 12**