بسم الله الرحمن الرحيم

جامعة القدس المفتوحة

برنامج التكنولوجيا والعلوم التطبيقية

الملحق العملي

لمقرر البرمجة الكينونية

بلغة جافا(1295)

إعداد: أ. طروب سعد

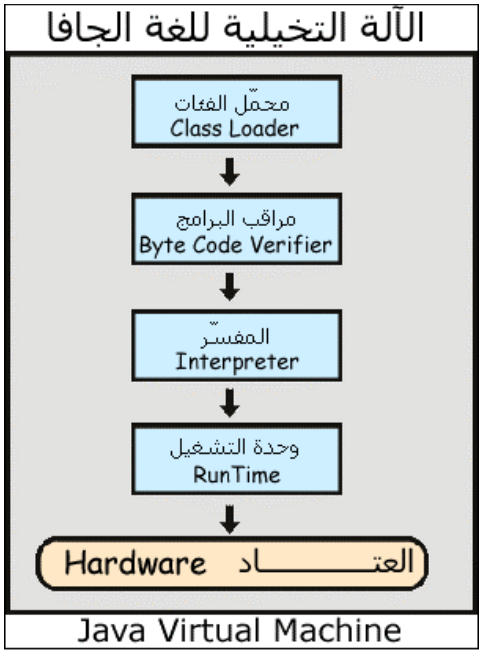
مقدمة لغة جافا

كثيراً ما نسمع في هذه الأيام عن لغة جافا، حتى يبدو وكأنها قي كل مكان حتى في المكتبات.كل هذا يجعلك تتساءل عن سبب انتشار هذه اللغة؟ والجواب ببساطة أنها تتيح للمستخدمين إمكانية تطوير تطبيقات تعمل على الويب والأجهزة الصغيرة كالهاتف النقال، كما أنها تستخدم في برمجة الشبكات.

كانت بدايات لغة جافا في أوائل التسعينيات من القرن العشرين في عام ١٩٩١، اخترعت لغة جافا شركة صن ميكروسيستمز. ولهذا الاختراع قصة عجيبة، حيث أن الشركة كانت قبل ذلك قد كلفت المهندس جيمس جوزلينج بوضع برامج لتشغيل الأجهزة التطبيقية الذكية مثل التليفزيون التفاعلي باستخدام لغة سي ++،وحينها وجد جيمس جوزلينج صعوبة في التعامل مع هذه اللغة، فقام هو وفريق العمل المساعد له بتطويرها، فولدت لغة جديدة تتوافق مع احتياجاته فكانت لغة جافا، وقد خططت شركة صن في تلك الأيام لاستغلال هذه اللغة الوليدة في التليفزيون التفاعلي لكي تربح المليارات، وحدث نوع من البطء في مشروع التلفزيون التفاعلي ربما عن قصد من الشركات الأخرى المنافسة، ونتيجة لذلك فكرت الشركة في ايقاف مشروع تطوير هذه اللغة الوليدة وتسريح العاملين في هذا المشروع أو نقلهم إلى قسم آخر. ولكن حدث ما لم يكن في الحسبان حيث أنه في هذه الفترة كانت الانترنت قد بدأت في الانتشار بسرعة مذهلة مع نزول نظام الويندوز للأسواق، وحيث أن لغة جافا الوليدة التي اخترعت أصلا لبرمجة الأجهزة التطبيقية فيها من السمات ما يجعلها أكثر توافقا مع الشبكة العنكبوتية الدولية الإنترنت، فقد كان لها السبق وأضافت الكثير إلى الإنترنت الذي كان قبلها مقصوراً على تبادل النصوص. ولكن المطورين بشركة صن ابتكروا طريقة تجعل برامج جافا تعمل بسهولة في صفحات الإنترنت وغيروا الاسم الذي كان قد أطلقه عليه مبتكرها من آواك شجرة السنديان إلى الجافا، ومن هنا أصبحت جافا مرتبطة في شهرتها بالإنترنت، حيث أن برنامج جافا صغير يوضع في صفحة من صفحات موقع على الشبكة الدولية يراه الملايين في جميع أنحاء العالم في نفس الوقت وقد كان هذا لا يتوفر إلا مع الجافا مما أعطاها شهرة واسعة. ولحسن حظ شركة صن أن لغة جافا أكدت نفسها في المجال الذي طورت له أصلاً، فقد بدأ الآن التليفزيون التفاعلي في الانتشار وما يسمى سينما المنزل والمشاهدة حسب الطلب. وليس هذا فقط بل أنتشر ما هو أكثر فائدة لشركة صن وهو الهاتف المحمول وللجافا أكبر دور في برمجة البرامج التي يعمل بها في أجياله السابقة واللاحقة، ولا نستغرب أن يحدث نوع من الغيرة بين شركة ميكروسوفت وشركة صن ميكروسيستمز، مما دفع ميكروسوفت إلى أن تحذف ماكينة الجافا الافتراضية من الإصدارة الأولى للويندوز اكس بي وهذه الماكينة الافتراضية مسئولة عن عرض برامج الجافا على الانترنت ولكن ميكروسوفت تراجعت أمام طلب ملايين المستخدمين حول العالم فوضعتها مرة ثانية في الإصدارات اللاحقة وقد كانت قضية مشهورة تناولتها الصحف والمجلات.

وقد ساهم في شهرة جافا أيضاً برامجها العلمية التفاعلية التي تصلح لمعظم المناهج التعليمية في جميع مراحل التعليم وبالتالي فان لها دوراً كبيراً في التعليم الالكتروني والتعليم عن بعد والفصول الافتراضية.

من مميزات جافا أن جميع البرمجيات التي تكتب بالجافا لا يتم ترجمتها للغة الآلة مثل باقي لغات البرمجة بل يتم ترجمتها إلى ما يسمى بالـ ByteCode (تحفظ كملف بإمتداد ".class")، وهذه الأخيرة عبارة عن لغة وسيطة بين لغة البرمجة ولغة الآلة، بالتالي فإن برمجيات الجافا تعمل على "جهاز جافا الإفتراضي" (Java Virtual Machine)،وهو عبارة عن برنامج مشغل لبرمجيات الجافا، مما يجعل برمجيات الجافا غير قابلة للتشغيل في حالة عدم وجود هذا البرنامج، ولكن في نفس الوقت فإن هذه الخاصية تجعل لغة جافا أقوى بكثير من نظيراتها، حيث أنه بفضل هذه الخاصية يمكننا تشغيل اي برنامج جافا كتب على أي نظام تشغيل على أي نظام تشغيل آخر دون الحاجة لإعادة عملية البرمجة والترجمة.



مكونات الآلة التخيلية لجافا

مزايا لغة جافا:

صنعت لغة جافا حتى تقدم بعض الخدمات المتميزة عن اللغات الأخرى وهي :

1. الإنتقالية (portability) : وهي إمكانية تشغيل برمجيات الجافا على أي نظام تشغيل كما ذكرنا بالأعلى.
2. الأمان (security) : كون برمجيات الجافا لا تعمل إلا عن طريق جهاز جافا الإفتراضي فبالتالي فإن هذا الجهاز يضع حدوداً لإمكانيات بعض البرمجيات مما يعيق إمكانية صنع برمجيات مخربة عن طريق هذه اللغة.
3. كائنية التوجه (Oriented Object): أن ابسط برنامج يمكن عمله بالجافا لا بد ان يتبع توجه البرمجة الكائنية التي تسهل الكثير والكثير من المشاكل بالنسبة لمتطور البرمجيات.
4. (Distributed): يمكن صنع برامج جافا تعمل على الشبكة و تتخاطب مع بعضها البعض بكل سهولة.

مع أن لغة جافا تختلف كثيرا عن قريناتها في طريقة عملها وفي خصائصها إلا أنها تتشابه كثيراً في قواعد تركيب وكتابة شفرتها بلغة الـ C++ المشهورة ، مما يسهل عملية تعلم الجافا لمبرمجي هذه اللغة.

منصات جافا

تقدم لغة برمجة جافا أكثر من منصة عمل للمبرمجين ومطوري النظم. كل من هذه المنصات لها خصائص وتقنيات تختلف عن بعضها البعض مما يجعل كل من هذه المنصات متخصصة في تطوير أنواع معينة من برمجيات جافا دون غيرها.

فلنرى كل من هذه المنصات و ما تقدمه لنا كمبرمجين ومطوري نظم:

1. (Edition (J2SE Java 2 Standard) :

تعتبر هذه المنصة الأكثر استخداماً في الجافا لتطوير برمجيات المكتبية و (Appletes). عموما توفر هذه المنصة وحدات البناء الأساسية لأي برنامج جافا مثلا:

- **الأمـور الأساسـية بالجافـا : تـوفر اساسـيات الكائنـات ووحـدات تعريـف المتغيـرات والنصـوص وغيرها من الأمور الأساسية في لغة الجافا في رزمة مثل Java.lang.**
- **إمكانيات القراءة والكتابة : التي توفر وسيلة الكتابة والقراءة للبرمجيات إما من وفي الملفات، قواعد البيانات ،الشبكات وغيرها متوفرة في رزمة Java.io.**
- **المكتبات المرئية : هي مكتبات تساعد في تكوين الواجهة الرسومية للمستخدم وأشهر رزمتين لتكوين الواجهات الرسومية هم : Java.awt, and Javax.swing.**
- **التعامل مع قواعد البيانات : توفير امكانية إدخال البيانات والبحث عنها، أغلبها في رزمة java.sql.**

و هنالك غير هذه الكثير من الرزم المفيدة المتوفرة مع هذه المنصة.

2. Java 2 Micro Edition (J2ME) :

مجموعة من مكتبات الجافا التي توفر امكانية صنع بريمجيات تعمل في بيئة تفتقر لقوة الحوسبة الكاملة مثل الأجهزة الصغيرة وأجهزة الهاتف النقال. اشتهرت الـ J2ME بسبب توفيرها بيئة سهلة لتصميم برمجيات اجهزة الهاتف النقال وأيضاً إمكانية أن يعمل البريمج على اكثر من نوع من انواع الهواتف.

3. Java 2 Enterprise Edition (J2EE) :

هذه المنصة تستعمل لتطوير بريمجات الجافا التي تعمل على جهة الخادم مثل صفحات الـ (JSP) والسيرفرليتس وغيرها من الأدوات والبريمجات المستعملة في تطوير المواقع والبرامج الشبكية.

كما أن هنالك منصات للجافا لا تأتي مع الجافا نفسها ولا هي من صنع صن ميكروسيستمس على سبيل المثال (وليس الحصر) الـ GWT والـ android و هاتين المنصتين هما من صنع جوجل: الأولى لبرمجة تطبيقات الويب والثانية لتطوير برامج تعمل على نظام تشغيل جوجل للجوالات الأندرويد.

إضافة لهذه المنصات هنالك العديد من اصناف او لغات مستخلصة من الجافا مثل الجـ روبي والـ جايثون و هما شبيهتان بالروبي و البايثون ولكنهما تعملان على جهاز جافا الإفتراضي مما يعني أن هذه اللغة تترجم إلى الجافا أولا ثم إلى البايت كود (طبعا هذا النوع من اللغات المعتمدة على الجافا لا تقتصر على هاتين اللغتين بل هنالك مجموعة كبيرة من لغات البرمجة تعمل بهذه الطريقة).

بيئة العمل في Java

يشـير المصـطلح IDE إلـى Integrated Development Environment وهـي البرنـامج الـذي يسـاعد المبـرمج ليكتـب ويتـرجم وينفـذ البـرامج المكتوبـة بلغـة جافـا، وتتـوافر البرامج التالية:

- **NetBeans**
- **Forte for Java**
- **Borland JBuilder**

لصناعة برامج باستخدام لغة جافا نحتاج إلى برنامجين وهما (المحرر والمترجم): و يتم تنزيلهم على الترتيب التالي:

1. **تنصيب المترجم:**

2. **تنصيب المحرر**

أما المترجم فهو تابع للغة التي نبرمج بها وهي لغة جافا و بالتالي فالمترجم يكون من اللغة ذاتها. ونستخدم برنامج اسمه Java Development Kit (JDK). هذا المترجم يكون مسئولا عن تحويل لغة الجافا التي نكتب بها برامجنا إلى لغة الهدف والتي تتحول إلى لغة الآلة.

أما المحرر, فيمكن أن يكون Jcreator , وهناك برامج أخرى مثل netbeans وذلك حسب ما يختاره مشرفك للتدريب. من مميزات Jcreator:

- **سهولة التعامل.**
- **تمييز مفردات لغة البرمجة.**

- سهولة ربطه بالمترجم.

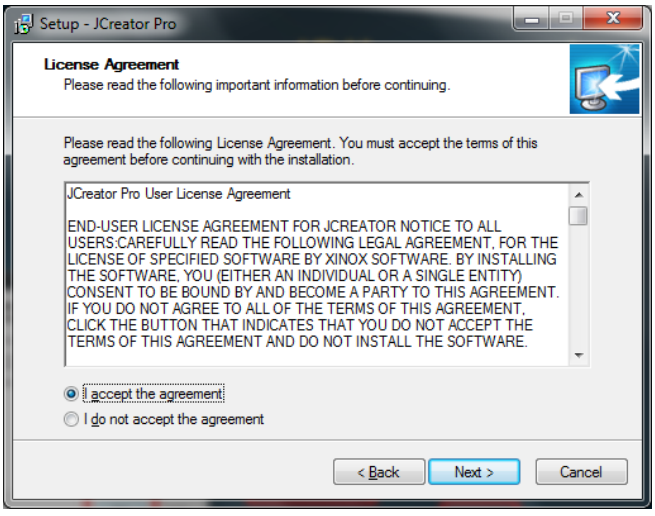
يمكن تحميل البرنامج والحصول عليه من خلال موقع الشركة على الإنترنت: http://www.jcreator.com، أو مراجعة مشرفك أو فني المختبر لتزويدك به.

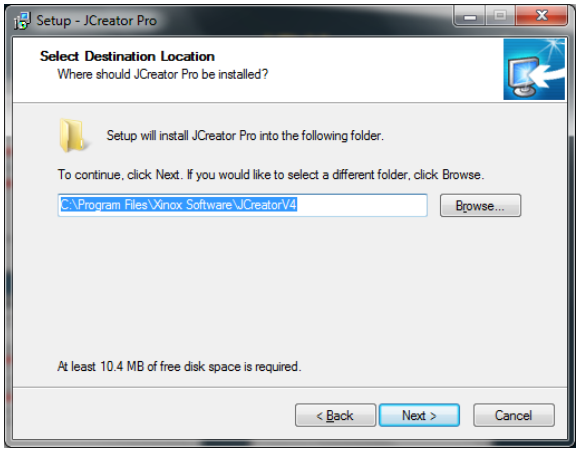
ومن فوائد محررات النصوص: تمييز الكلمات المحجوزة للغة بألوان معينة، تمييز المتغيرات بألوان معينة، اقتراح أسماء دوال و مساعدات على المبرمج.

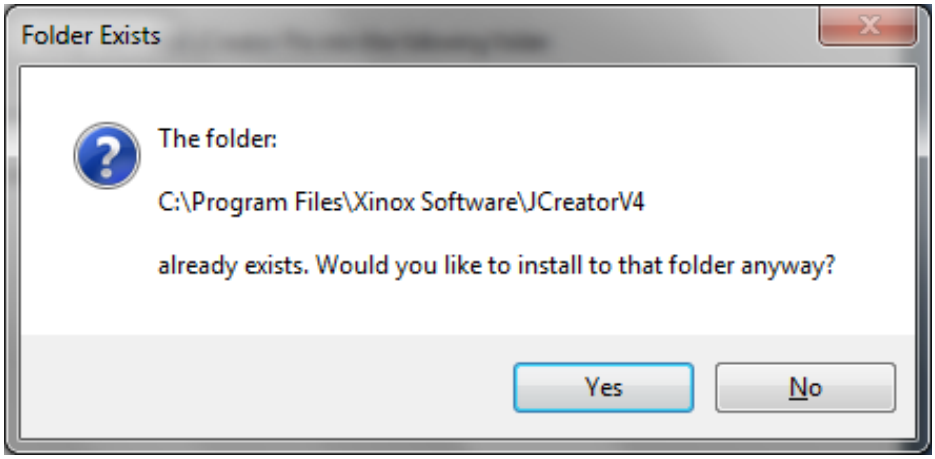
لتنصيب المترجم JDK، قم بتنزيل البرنامج كما أشرنا سابقًا، واتبع تتالي الخطوات.

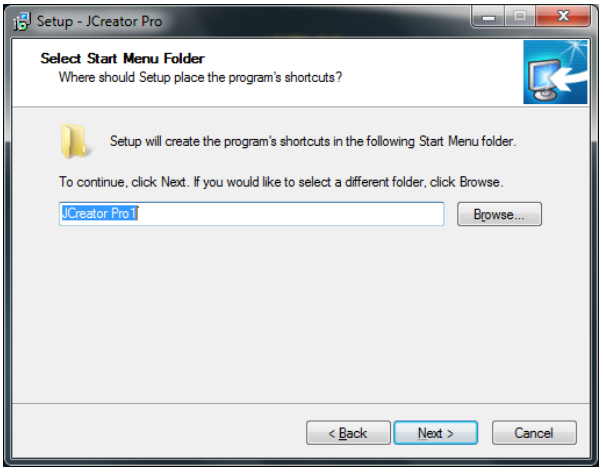
عزيزي الطالب: إذا اخترت Jcreator كمحرر نصوص إليك خطوات تنصيب المترجم Jcreator:

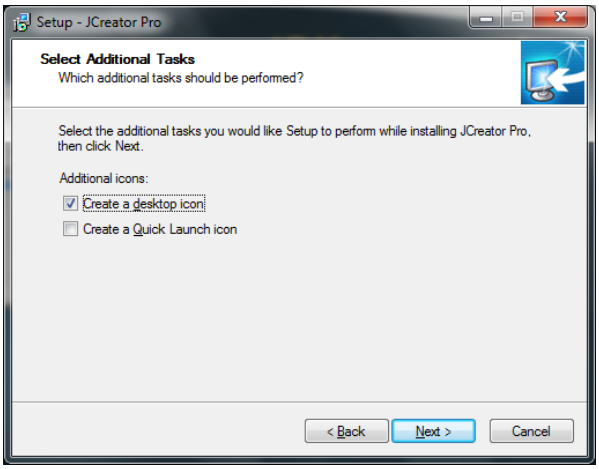
قم بالضغط على الملف التنفيذي للمترجم واتبع الخطوات كما في الأشكال التالية:

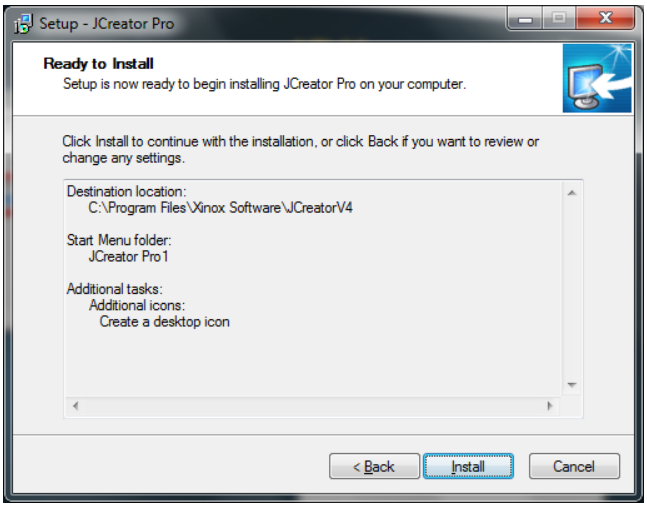


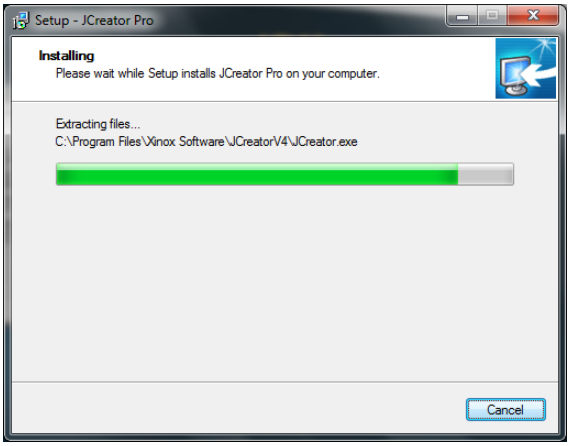

Setup - JCreator Pro
Select Destination Location
Where should JCreator Pro be installed?
Setup will install JCreator Pro into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\Program Files\Xinox Software\JCreatorV4
Browse...
At least 10.4 MB of free disk space is required.
< Back
Next >
Cancel


Folder Exists
The folder:
C:\Program Files\Xinox Software\JCreatorV4
already exists. Would you like to install to that folder anyway?
Yes
No

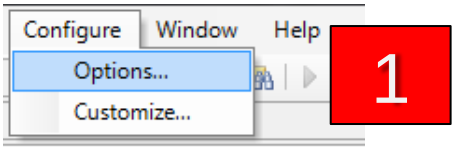

Setup - JCreator Pro
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
JCreator Pro1
Browse...
< Back
Next >
Cancel

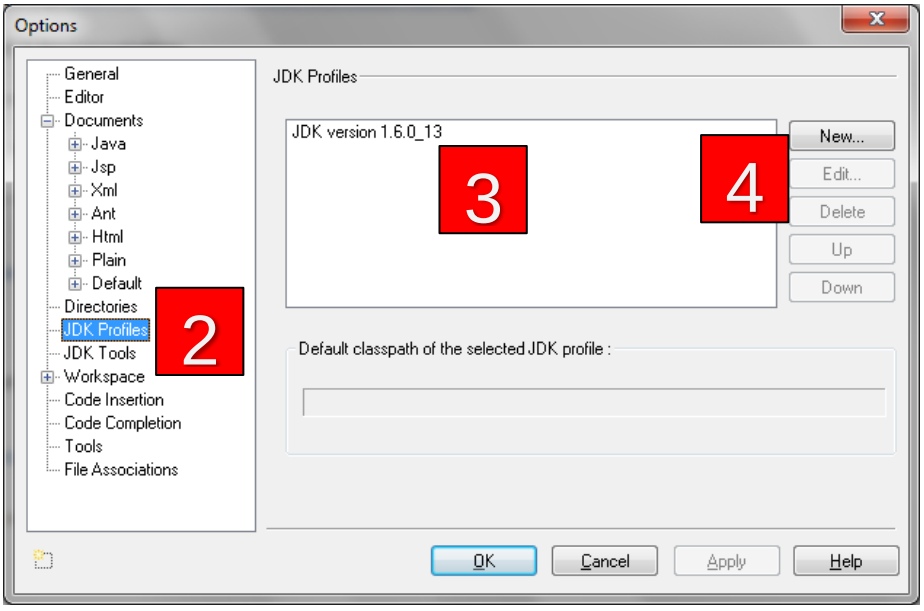

Setup - JCreator Pro
Select Additional Tasks
Which additional tasks should be performed?
Select the additional tasks you would like Setup to perform while installing JCreator Pro, then click Next.
Additional icons:
Create a desktop icon
Create a Quick Launch icon
< Back
Next >
Cancel

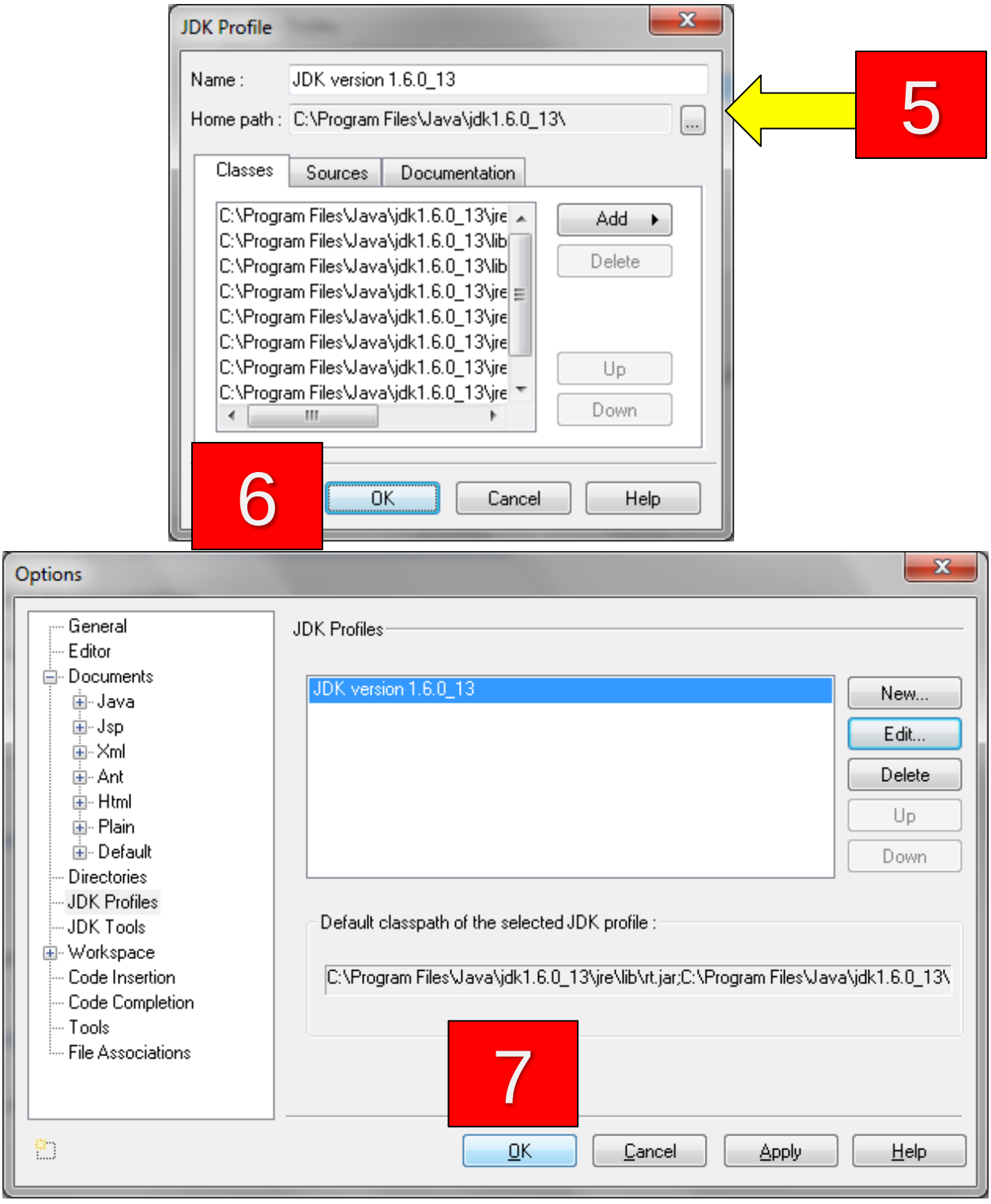

Setup - JCreator Pro
Ready to Install
Setup is now ready to begin installing JCreator Pro on your computer.
Click Install to continue with the installation, or click Back if you want to review or change any settings.
Destination location:
C:\Program Files\Xinox Software\JCreatorV4
Start Menu folder:
JCreator Pro1
Additional tasks:
Additional icons:
Create a desktop icon
< Back
Install
Cancel


Setup - JCreator Pro
Installing
Please wait while Setup installs JCreator Pro on your computer.
Extracting files...
C:\Program Files\Xinox Software\JCreatorV4\JCreator.exe
Cancel

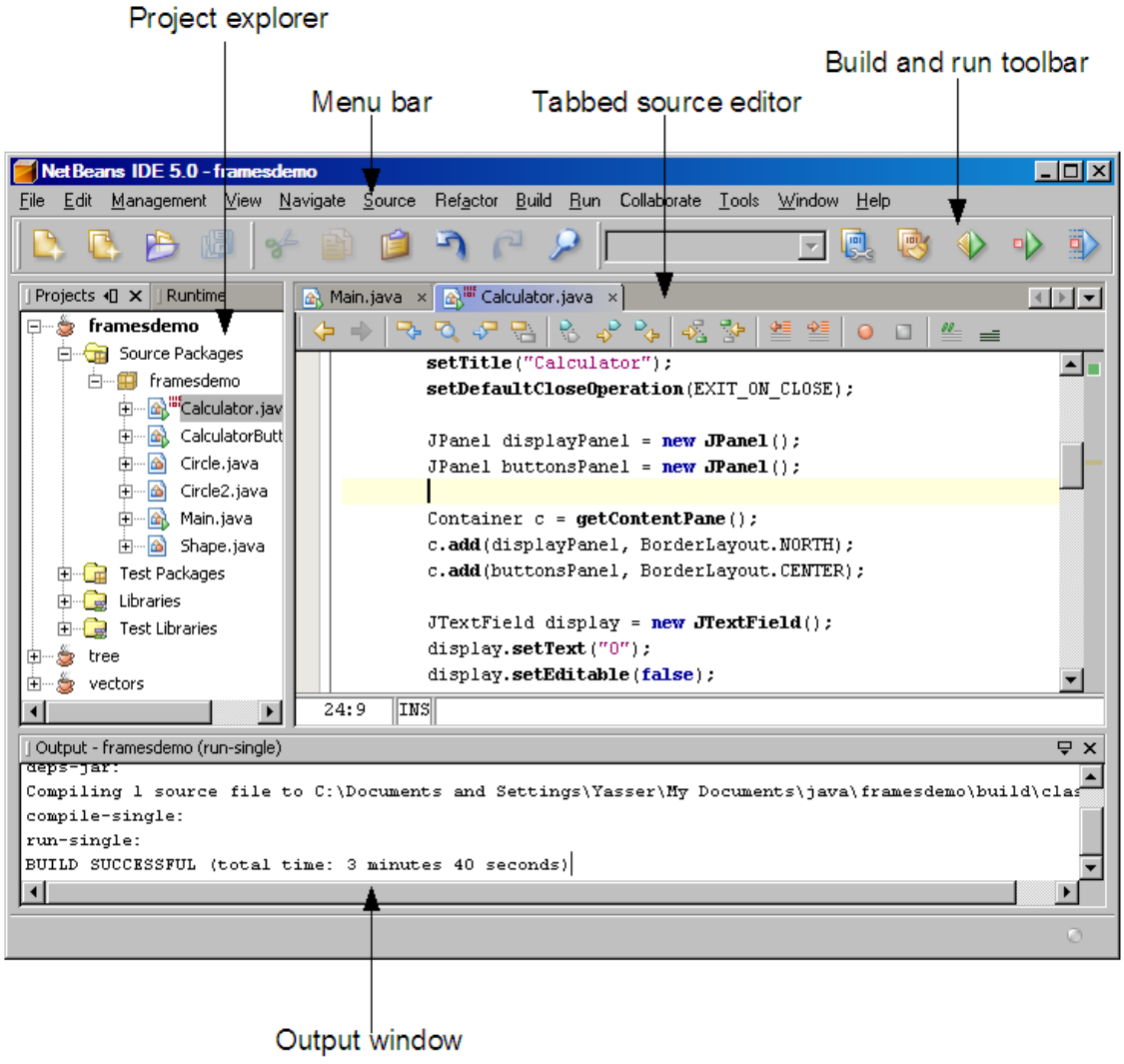
خطوات ربط Jcewaror ب JDK:






JDK Profile
Name :
JDK version 1.6.0_13
Home path :
C:\Program Files\Java\jdk1.6.0_13\
Classes
Sources
Documentation
C:\Program Files\Java\jdk1.6.0_13\jre
C:\Program Files\Java\jdk1.6.0_13\lib
C:\Program Files\Java\jdk1.6.0_13\lib
C:\Program Files\Java\jdk1.6.0_13\jre
C:\Program Files\Java\jdk1.6.0_13\jre
C:\Program Files\Java\jdk1.6.0_13\jre
C:\Program Files\Java\jdk1.6.0_13\jre
C:\Program Files\Java\jdk1.6.0_13\jre
Add
Delete
Up
Down
OK
Cancel
Help
5
6
Options
General
Editor
Documents
Java
Jsp
Xml
Ant
Html
Plain
Default
Directories
JDK Profiles
JDK Tools
Workspace
Code Insertion
Code Completion
Tools
File Associations
JDK Profiles
JDK version 1.6.0_13
New...
Edit...
Delete
Up
Down
Default classpath of the selected JDK profile :
C:\Program Files\Java\jdk1.6.0_13\jre\lib\rt.jar;C:\Program Files\Java\jdk1.6.0_13\
7
OK
Cancel
Apply
Help

إذا عملت عزيزي الطالب على برنامج NetBeans ، إليك الأجزاء الرئيسية للنافذة الخاصة به:



نافذة برنامج NetBeans الرئيسية

تمهيد الملحق العملي:

التركيبة الأساسية للبرنامج بلغة جافا:

البرنامج في أي لغة برمجة يتكون من مجموعة من الجمل و الكلمات الأساسية التي لا يمكن أن يعمل أبسط برنامج من دونها, وهي كما تظهر في الشكل:

```
public class first {

    public static void main(String[] args) {

        // TODO, add your application code
        System.out.println("Hello To !");
    }
}
```

- **كلمات إنشاء المشروع (public class) واسمه.**
- **الدالة الأساسية (main) وتركيبتها.**
- **أقواس البداية والنهاية.**
- **جمل الإدخال و الإخراج (من دونها قد يعمل البرنامج دون تأثير ملموس).**

شرح البرنامج الأول في لغة جافا:

```
public class first {

    public static void main(String[] args) {

        // TODO, add your application code
        System.out.println("Salamo 3likom, My Name is Mahmoud");
    }
}
```

- **كلمة بناء البرنامج: public class**
- **كلمة public تعني أن ما في داخل هذا الكائن البرمجي يمكن لأي كائن آخر استخدامه و التعامل معه فهو يعتبر (عام) داخل هذا المشروع.**

- كلمة class تعني كائن و هو البنية الأساسية للغة جافا, فكل برنامج في لغة جافا لابد أن يحتوي على الأقل على تعريف كائن واحد.
- كلمة first هي اسم (identifier) لهذا الكائن وهي متغيرة حسب طبيعة عمل الكائن.

الدالة الأساسية main:

- هي الدالة الأساسية في كل مشاريع البرمجة في لغة جافا.
- تبدأ ترجمة المشروع دائمًا من الدالة main ولها تركيبة واضحة وثابتة كما تشاهد في الأمثلة التالية:

```
public static void main(String[] args) {

    // TODO, add your application code
    System.out.println("Salamo 3likom, My Name is Mahmoud");
}
```

```
public static void main(String[] args) {

    // TODO, add your application code
    System.out.println("Salamo 3likom, My Name is Mahmoud");
    System.out.println("My Relegin is Islam");
    System.out.println("I'm a proud to be Muslim");
        }
```

تتكون الدالة الرئيسية من:

- التوقيع (Signature): هو رأس الدالة و تعرف الدالة به, و يتكون التوقيع من اسم الدالة , المعاملات و أنواعها, القيمة التي تعيدها الدالة, و درجة خصوصية الدالة.

```
public static void main(String[] args) {
```

- الجسم (Body): و هو المنطقة التي يتم فيها كتابة الأوامر و التعليمات التي تمثل المهمة التي نرغب في تنفيذها.

```
// TODO, add your application code
System.out.println("Salamo 3likom, My Name is Mahmoud");
```

- البداية و النهاية ({ }) : هي الأقواس التي تحدد بداية جسم الدالة و نهايتها, فالبداية هي { و النهاية } .

```
public static void main(String[] args) {


}
```

☜ عزيزي الطالب:

سيتم تناول لغة جافا في هذا الملحق العملي من خلاف مجموعة برامج تطبيقية تقسم إلى مختبرات، كل مختبر يتناول مجموع من المفاهيم في لغة جافا، ويجري التطبيق عليها في مختبر الحاسوب بإشراف مشرف المقرر، ومتابعة أداء الطلبة.

المختبر الأول: مكونات البرنامج الأساسية وطباعة أول برنامج بلغة جافا

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة:

1. **يفهم بعض المصطلحات والمفاهيم الأساسية في لغة جافا.**
2. **معرفة مكونات برنامج جافا الأساسية.**
3. **إعداد البيئة التشغيلية للغة جافا وتنفيذ البرنامج الأول.**
4. **التعرف على مكونات مناهج الصنف JOptionPane الخاصة باظهار صناديق الحوار.**
5. **طباعة مخرجات باستخدام جملة الطباعة الافتراضية System.out.println**

ثانياً: عزيزي الطالب:

يقسم اللقاء الأول في المختبر إلى الأنشطة التالية:

1. **يتم تهيئة الطلاب لعملية تحميل المترجم الخاص بلغة جافا وذلك حسب ما يراه المشرف مناسبا وما يختاره لعملية التدريب العملي من مترجمات.**
2. **وبعد عملية ارشاد الطلاب الى كيفية تحميل المترجم وما يلزم من برامج، يتم تنفيذ البرنامج التالي.**

عزيزي الطالب:

- **قم بتنفيذ البرنامج التالي على احد مترجمات لغة جافا، وارصد المخرجات كما هي اسفل البرنامج:**

```
1  //   Welcome1.java
2  // THE FIRST PROGRAMME IN JAVA
3
4  public class Welcome {
5
6    // main method begins execution of Java application
7    public static void main( String args[] )
8    {
9      System.out.println( "Welcome to Java First Lab" );
10
11    } // end method main
12
13 } // end class Welcome
```

ملاحظة:

استخرج من المثال السابق المكونات الاساسية لبرنامج لغة جافا، واذكرها في البنود التالية وذلك بتحديد رقم السطر:

.1

.2

.3

.4

المخرجات:

```
Welcome to Java First Lab
```

- **حاول عزيزي الطالب ان تقوم بتطوير البرنامج السابق باضافة سطر مخرجات اضافي بيحث تصبح المخرجات كالتالي:**

```
Welcome to Java First Lab

I study in Al Quds Open University
```

احد الاجابات المقترحة:

```
//     Welcome1.java
2     // THE FIRST PROGRAMME IN JAVA
3
4     public class Welcome {
5
6        // main method begins execution of Java application
7        public static void main( String args[] )
8        {
9           System.out.println( "Welcome to Java First Lab" );
10          System.out.println( " I study in Al Quds Open University

" );
11


12
13       }  // end method main
14
15    }  // end class Welcome
```

☜ اكتب البرنامج التالي بلغة جافا ولاحظ المخرجات التي تحصل عليها في صندوق الحوار:

```
// Welcome4.java
2   // Printing multiple lines in a dialog box

4   // Java extension packages
5   import javax.swing.JOptionPane;  // import class JOptionPane

7   public class Welcome4 {

9      // main method begins execution of Java application
10     public static void main( String args[] )
11     {
12        JOptionPane.showMessageDialog(
13           null, "Welcome\nto\nJava\nProgramming!" );

15        System.exit( 0 );  // terminate application
17     }  // end method main

19  }  // end class Welcome4
```

المخرجات:

المختبر الثاني: المصفوفات + السلاسل الرمزية + JOpionPane

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة:

1. **التعرف على المفاهيم الأساسية بلغة جافا**
2. **التعرف على اللبنات التي تكون لغة جافا من جمل وتعابير ومشغلات**
3. **أن يتعامل الطالب مع جمل التحكم والدوران**
4. **أن يتعامل الطالب مع السلاسل الرمزية والمصفوفات**
5. **أن يوظف المناهج الخاصة بالسلاسل الرمزية والمصفوفات في بناء برامج تطبيقية منوعة.**
6. **أن يوظف مناهج الصنف JOpionPane في تطبيقات مختلفة.**
7.

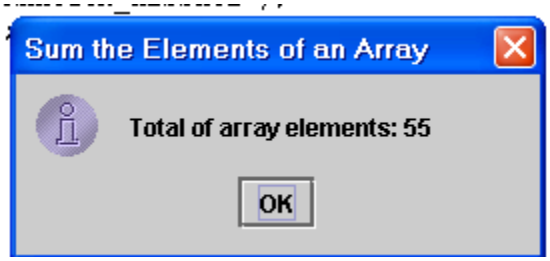
ثانياً: عزيزي الطالب:

[1]تأمل البرنامج التالي المكتوب بلغة جافا ثم اجب عن الأسئلة التي تليه:

```
import javax.swing.*;
public class sumarray {
// main method begins execution of Java application
public static void main( String args[] )
{
int array[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };
int total = 0;
// add each element's value to total
for ( int counter = 0; counter < array.length; counter++ )
total += array[ counter ];
JOptionPane.showMessageDialog( null,
"Total of array elements: " + total,
"Sum the Elements of an Array",
JOptionPane.INFORMATION_MESSAGE );
System.exit( 0 );}}
```

قم بطباعة البرنامج التالي وتنفيذه، وارصد النتائج التي تحصل عليها

1. ما وظيفة البرنامج؟
2. اكتب جملة تعريف المصفوفة.
3. اكتب جملة الدوران التي تقوم بعملية الجمع وحدد مكونات هذه الجملة؟
4. اكتب البرنامج في محرر مترجم لغة جافا وقم بتنفيذه ولاحظ صندوق الحوار الذي يظهر فيه المجموع كما في الأسفل:



5. استخرج الجملة التي تعمل على إظهار صندوق الحوار السابق وحدد مكوناتها الأساسية.
6. لو أردنا إظهار المجموع بدون أن يظهر صندوق الحوار السابق، وذلك عن طريق ظهور الجملة التالية في نافذة output في مترجم جافا:

Total of array elements is: 55

اكتب العبارة اللازمة لذلك.

[1] تأمل البرنامج التالي الذي يحوي استخدامات المناهج المختلفة للسلاسل الرمزية في جافا:

```
public class StringBufferFunctionsDemo {
   public static void main(String[] args) {

//        Examples of Creation of Strings
          StringBuffer strBuf1 = new StringBuffer("JAVA EXAMPLES");
          StringBuffer strBuf2 = new StringBuffer(100);  //With capacity 100
          StringBuffer strBuf3 = new StringBuffer();          //Default Capacity 16
          System.out.println("strBuf1 : "+strBuf1);
```

```
        System.out.println("strBuf1 capacity : "+strBuf1.capacity());
        System.out.println("strBuf2 capacity : "+strBuf2.capacity());
        System.out.println("strBuf3 capacity : "+strBuf3.capacity());
         System.out.println("strBuf1 length : "+strBuf1.length());
        System.out.println("strBuf1 charAt 2 : "+strBuf1.charAt(2));
//      A StringIndexOutOfBoundsException is thrown if the index is not valid.
        strBuf1.setCharAt(1, 't');
        System.out.println("strBuf1 after setCharAt 1 to t is : "+strBuf1);
        System.out.println("strBuf1 toString() is : "+strBuf1.toString());
        strBuf3.append("beginner-java-PROGRAMMING");
        System.out.println("strBuf3 when appended with a String : "+strBuf3.toString());
        strBuf3.insert(1, 'c');
        System.out.println("strBuf3 when c is inserted at 1 : "+strBuf3.toString());
        strBuf3.delete(1, 'c');
        System.out.println("strBuf3 when c is deleted at 1 : "+strBuf3.toString());
        strBuf3.reverse();
        System.out.println("Reversed strBuf3 : "+strBuf3);
        strBuf2.setLength(5);          }}
```

المخرجات:

```
strBuf1 : JAVA EXAMPLES
strBuf1 capacity : 29
strBuf2 capacity : 100
strBuf3 capacity : 16
strBuf1 length : 13
strBuf1 charAt 2 : V
```

strBuf1 after setCharAt 1 to t is : JtVA EXAMPLES

strBuf1 toString() is : JtVA EXAMPLES

strBuf3 when appended with a String : beginner-java-PROGRAMMING

strBuf3 when c is inserted at 1 : bceginner-java-PROGRAMMING

strBuf3 when c is deleted at 1 : b

Reversed strBuf3 : b

المختبر الثالث: البّناء والمناهج والتوارث

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة:

1. **التعرف على الصنف وعملياته المختلفة.**
2. **أن يستخدم الطالب المناهج بخصائصها وأدواتها المختلفة في كتابة حلول لمسائل رياضية وتطبيقية منوعة.**
3. **أن يتعرف الطالب على الصنف الفرعي الذي يرث الصنف الأساسي ويوظف ذلك في برنامج تطبيقي.**
4. **أن يتعرف على مفهوم البناء ويوظفه في برامج تطبيقية مختلفة.**

ثانياً:

1. **البرنامج الأول: تأمل البرنامج التالي وقم بطباعته وتنفيذه على احد مترجمات جافا ولاحظ النتائج عند استدعاء المناهج:**

```
public class Account {
    protected double balance;
    // Constructor to initialize balance
    public Account( double amount )      {
            balance = amount;    }
    // Overloaded constructor for empty balance
    public Account()          {
            balance = 0.0;          }
    public void deposit( double amount ) {
            balance += amount;  }
    public double withdraw( double amount )    {
```

```
            // See if amount can be withdrawn
                if (balance >= amount)                {
                        balance -= amount;
                return amount;                }
                else
            // Withdrawal not allowed
                return 0.0;     }
        public double getbalance()      {
            return balance;     }}
class AccountDemo {
        public static void main(String args[]) {
            // Create an empty account
            Account my_account = new Account();
            // Deposit money
                my_account.deposit(250.00);
            // Print current balance
                System.out.println ("Current balance " +
                        my_account.getbalance());
            // Withdraw money
                my_account.withdraw(80.00);
            // Print remaining balance
                System.out.println ("Remaining balance " +
                        my_account.getbalance());   }}
```

هذا المثال هام جدا، يتم فيه تعريف الطالب بالمفاهيم التالية:

1. مفهوم البناء بأنواعه (البناء الافتراضي والبناء الذي يأخذ عدة معاملات أو معامل واحد).
2. مفهوم المنهج وأنواعه(الذي يمرر معاملات أو لا يمرر، يعيد قيمة أو لا يعيد قيمة، ...الخ).
3. إنشاء مثيل للصنف من خلال جملة new.
4. كيفية استدعاء المناهج داخل main.

ناتج البرنامج:

Current balance 250.0

Remaining balance 170.0

2 . البرنامج الثاني:

اكتب البرنامج اللازم لاستدعاء المنهج اللازم لإيجاد الجذر ألتربيعي للأعداد من 5−50 (استخدم المناهج في الصنف Math).

☜ **احد الحلول المقترحة**

```
public class sqrooot {
 public static void getsqrtoot(){
        for (int i=5;i<=50; i++ )
                System.out.println(Math.sqrt ((double)i));
 }
  public static void main(String[] args) {
  getsqrtoot();          }}
```

☜ **المخرجات:**

2.23606797749979

2.449489742783178

2.6457513110645907

2.8284271247461903

3.0

…….

6.782329983125268

6.855654600401044

6.928203230275509

7.0

7.0710678118654755

البرنـامج الثالـث: تأمـل البرنـامج التـالي وقـم بطباعتـه وتنفيـذه علـى احـد مترجمـات جافـا ولاحـظ خاصـية التوارث:

```
class Box {
        double width;
        double height;
        double depth;
        Box() {}
        Box(double w, double h, double d) {
                width = w;
                height = h;
                depth = d;      }
        void getVolume() {
                System.out.println("Volume is : " + width * height * depth);     }}
 class MatchBox extends Box {
        double weight;
```

```
    MatchBox() { }
    MatchBox(double w, double h, double d, double m) {
        super(w, h, d);
        weight = m;   }
    public static void main(String args[]) {
        MatchBox mb1 = new MatchBox(10, 10, 10, 10);
        mb1.getVolume();
        System.out.println("width of MatchBox 1 is " + mb1.width);
        System.out.println("height of MatchBox 1 is " + mb1.height);
        System.out.println("depth of MatchBox 1 is " + mb1.depth);
        System.out.println("weight of MatchBox 1 is " + mb1.weight); }}
```

المخرجات:

```
Volume is : 1000.0
width of MatchBox 1 is 10.0
height of MatchBox 1 is 10.0
depth of MatchBox 1 is 10.0
weight of MatchBox 1 is 10.0
```

☜ **ينفذ الطالب البرنامج ويلاحظ النتائج ويقوم بإنشاء برنامج شبيه حول إنشاء صنف من أي نوع يختاره ويطبق مفاهيم الوراثة والبناء والاستدعاء للمناهج وإنشاء مثيلات الصنف، يقوم الطالب بإرسال قيم مختلفة عند الاستدعاء وملاحظة النتائج.**

المختبر الرابع: استخدام عدة صناديق حوار + parsing + محددات الوصول للمتغيرات

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة:

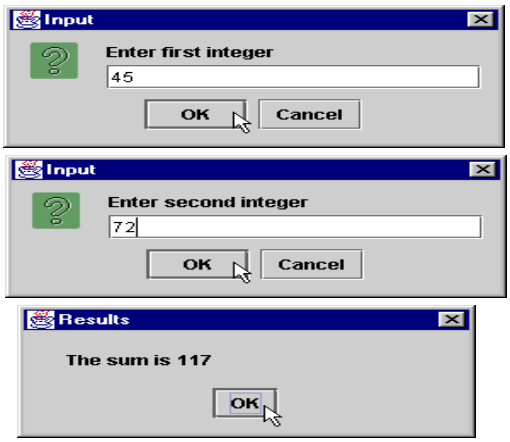
1. **التعرف على الصنف ومحددات الوصول إليه المختلفة.**
2. **تطبيق مفهوم parsing في تحويل نوع بيانات إلى آخر؟**
3. **أن يوظف مفهوم ومناهج JOptionPane في تطبيق برنامج حواري مع المستخدم لقراءة بيانات وطباعة مخرجات؟**

ثانياً: عزيزي الطالب:

☜ **نفذ البرنامج التالي باستخدام احد مترجمات جافا ولاحظ النتائج التي تظهر كما هو وارد أسفل البرنامج:**

```
//  Addition.java
   // An addition program.
      // Java extension packages
   import javax.swing.JOptionPane;  // import class JOptionPane
      public class Addition {
         // main method begins execution of Java application
    public static void main( String args[] )     {
       String firstNumber;   // first string entered by user
       String secondNumber;  // second string entered by user
       int number1;          // first number to add
       int number2;          // second number to add
       int sum;              // sum of number1 and number2
          // read in first number from user as a String
       firstNumber =
```

```
        JOptionPane.showInputDialog( "Enter first integer" );
        // read in second number from user as a String
      secondNumber =
        JOptionPane.showInputDialog( "Enter second integer" );
        // convert numbers from type String to type int
      number1 = Integer.parseInt( firstNumber );
      number2 = Integer.parseInt( secondNumber );
        // add the numbers
      sum = number1 + number2;
// display the results
      JOptionPane.showMessageDialog(
        null, "The sum is " + sum, "Results",
        JOptionPane.PLAIN_MESSAGE );
        System.exit( 0 );   // terminate application
      } // end method main
   } // end class Addition
```



☜ عزيزي الطالب: نفذ البرنامج التالي باستخدام احد مترجمات جافا ولاحظ النتائج التي تظهر كما هو وارد أسفل البرنامج: (البرنامج تطبيق حول أنواع محددات الوصول للمتغيرات)

```
public class StaticClass {
    public static int x = 0;
    private double vat = 14.5;
     public static void main(String p[] ) {
       byte largestByte=Byte.MAX_VALUE;
      StaticClass a = new StaticClass();
```

```
StaticClass b = new StaticClass();

a.vat=17.0; b.x=13;

System.out.println(a.vat+","+ b.vat);

System.out.println(a.x+","+ b.x);

x=15; a.x=10;b.x=20;

System.out.println(x+ "," +a.x + "," +b.x);


System.out.println("the biggest number is :     "+largestByte);   }}
```

المخرجات:

17.0,14.5

13,13

20,20,20

the biggest number is: 127

المختبر الخامس: Applets والمقارنة بينها وبين البرامج التطبيقية

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

1. **التعرف على أنواع البرامج في لغة جافا.**
2. **المقارنة بين البرامج التطبيقية في لغة جافا والبرامج من نوع Applets.**
3. **توظيف Applets في كتابة طباعة مخرجات تظهر في نافذة رسومية.**
4. **متابعة ملف html المرافق لتطبيق Applet.**

ثانياً: عزيزي الطالب:

1. **نفذ البرنامج التالي واحصل على النتائج كما هو موضح أسفل البرنامج:**

```
//  WelcomeApplet2.java

   // A first applet in Java.

    // Java core packages

   import java.awt.Graphics;    // import class Graphics

    // Java extension packages

   import javax.swing.JApplet;  // import class JApplet

    public class WelcomeApplet2 extends JApplet {

       // draw text on applet's background

     public void paint( Graphics g )     {

       // call inherited version of method paint

       super.paint( g );

          // draw a String at x-coordinate 25 and y-coordinate 25

       g.drawString( "Welcome to Java Programming!", 25, 25 );

       }  // end method paint

    }  // end class WelcomeApplet
```

المخرجات

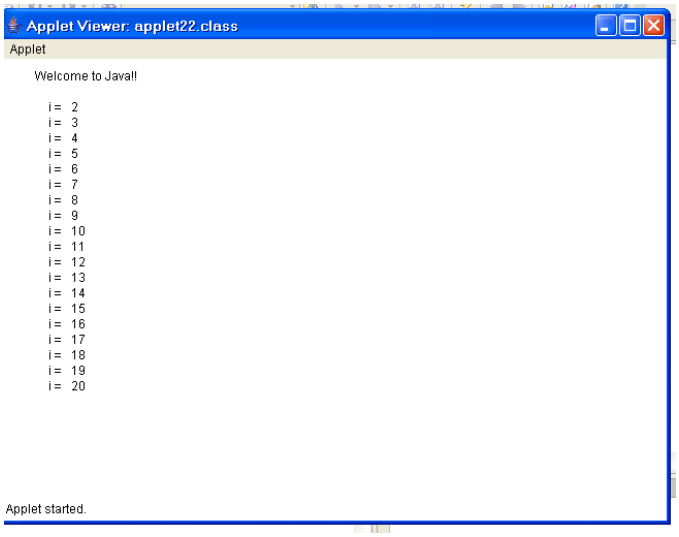
☜ يلاحظ الطالب المكونات الأساسية للبرنامج من نوع Applet واستبدال المنهج main بالمنهج paint، كما يوجه انتباه الطالب إلى الملفhtml الذي يظهر في نافذة المترجم ومكوناته كما يلي:

```
<html>
        <head>
        </head>
        <body bgcolor="000000">
                <center>
                        <applet
                                code    = "WelcomeApplet2.class"
                                width   = "500"
                                height  = "300"
                                >
                        </applet>
                </center>
        </body>
</html>
```

2. تأمل عزيزي الطالب البرنامج التالي الذي يعمل على طباعة مضاعفات العدد 2 في نافذة Applet، باستخدام جملة التكرار (لاحظ المخرجات أسفل البرنامج) :

```
import java.awt.*;
import java.applet.*;
public class applet22 extends Applet {
public void init() {    }
```

```
public void paint(Graphics g) {

g.drawString("Welcome to Java!!",30,20);

int f=40; int y=50;


for (int i=2; i<=20; i++){

g.drawString(" i =   "+i,f,y );y=y+15;        }                }}
```



- عزيزي الطالب، قم بتغيير لون نافذة الابليت إلى اللون الأزرق، ولون النص إلى الأحمر .
- ارسم شكلا هندسيا متوازي المستطيلات أسفل مضاعفات العدد 2 .

المختبر السادس: تطبيق مهارات أساسية في جافا

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

1. **استخدام المفاهيم الأساسية في لغة جافا لكتابة برامج تطبيقية منوعة.**

ثانياً: عزيزي للطالب:

1. **اكتب برنامج يقوم باحتساب العدد الأكبر بين ثلاثة أعداد (استخدم مفهوم المنهاج)؟**

احد الحلول المقترحة:

```
public class com {
        int a, b, c;
        public    com (int e, int f,int g ){
                a=e;
                b=f;
                c=g;    }
        public int getlarge(){
        int d;
        if (a>b)
                d=a;
                else
                        d=b;
        if (c>d)
                d=c;
        return d;       }

  public static void main(String[] args) {
```

```
com c1 = new  com(104,16,54);
System.out.println("The biggest number:"+ c1.getlarge ());    }}
```

2. اكتب برنامج لحساب مساحة الدائرة واستخدم صندوق الحوار (Dialog box):

احد الحلول المقترحة:

```
import javax.swing.*;
public class ComputeAreaNew {
public static void main(String[]args){
double r2,r1, area;
final double PI = 3.14159; // Declare a constant
r2 = Double.parseDouble (JOptionPane.showInputDialog("Enter the length"));
r1 = Double.parseDouble (JOptionPane.showInputDialog("Enter the width"));
area = r2*r1;
JOptionPane.showMessageDialog(null, "The area is " + area);}}
```

3. اكتب برنامج اللازم لطباعة النص al-quds open university بصورة عكسية:

```
public class Stringd{
public static void main (String []args){
String a= "al-quds open university";
int len = a.length();
StringBuffer b = new StringBuffer (len);
for (int i=(len-1); i>=0; i--) {
b.append (a.charAt(i));}
System.out.println(b.toString());}}
```

المخرجات:

ytisrevinu nepo sduq-la

المختبر السابع: استخدام جمل التشعبات والتحكم

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

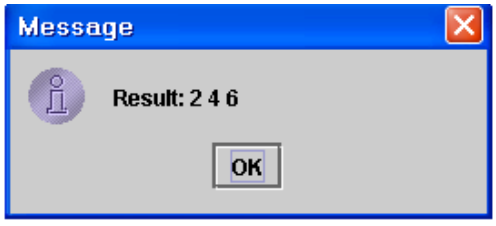
1. استخدام جمل التشعبات المختلفة في برامج منوعة.
2. التعرف على وظيفة جملة continue
3. التعرف على وظيفة جملة break
4. استخدام جمل التحكم والدوران في تطبيقات منوعة.

ثانياً: عزيزي الطالب:

1. **نفذ البرنامج التالي باستخدام احد مترجمات جافا ولاحظ تاثير جملة continue الواردة في البرنامج، ثم لاحظ النتائج التي تظهر كما هو وارد أسفل البرنامج:**

```
import javax.swing.JOptionPane;

public class adf {

public static void main( String args[] ){

String output = "Result: ";

for ( int count = 2; count <= 13; count +=2) {

if ( count > 7 )

continue;

output += count + " ";

}

JOptionPane.showMessageDialog( null, output );

}}
```

المخرجات:



2. عزيزي الطالب: نفذ البرنامج التالي باستخدام احد مترجمات جافا ولاحظ تأثير جملة **break** الواردة في البرنامج، ثم لاحظ النتائج التي تظهر كما هو وارد أسفل البرنامج:

```
import javax.swing.JOptionPane;
public class asdf {
public static void main( String args[] )
{int array[]={1,2,3};
xyz:
for (byte i=0;i<array.length;i++) {
switch(array[i]) {
case 1: System.out.println("Once");break;
case 2: System.out.println("More");break xyz;
case 3: System.out.println("Chance");}}}}
```

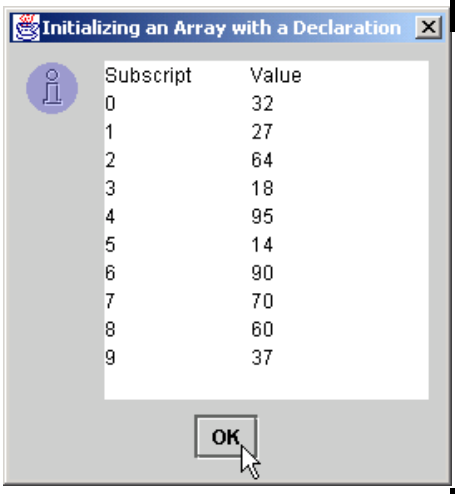
المخرجات:

```
Once
More
```

3. تأمل البرنامج التالي الذي يقوم بعرض عناصر مصفوفة على شكل أعمدة باستخدام صندوق الحوار:

```
import javax.swing.*;
     public class ArrayShow {
       // main method begins execution of Java application
    public static void main( String args[] )     {
      int array[] = { 32, 27, 64, 18, 95, 14, 90, 70, 60, 37 };  // dynamically allocate array
       String output = "Subscript\tValue\n";
```

```
      // append each array element's value to String output
      for ( int counter = 0; counter < array.length; counter++ )
         output += counter + "\t" + array[ counter ] +
"\n";
      JTextArea outputArea = new JTextArea();
      outputArea.setText( output );
      JOptionPane.showMessageDialog( null, outputArea,
         "Initializing an Array of int Values",
         JOptionPane.INFORMATION_MESSAGE );
         System.exit( 0 );      }   }
```



المختبر الثامن: استخدام الواجهات والحزم

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

1. التعرف على الواجهة كمفهوم بياني جديد.
2. تمكين الطالب من تعريف الواجهة ومكوناتها وتنفيذها.
3. التعرف على مفهوم الحزمة وأعضائها.
4. بناء برنامج تطبيقي كامل مستخدما مفهموم الواجهات والحزم.

ثانياً: عزيزي الطالب:

1. **تأمل البرنامج التالي الذي يقوم بإنشاء واجهة وتعريف صنف يوظف هذه الواجهة ويعيد تعرف كافة مناهجها، وارصد النتائج:**

```
interface Animal {
  public void eat();
  public void travel();}
public class II implements Animal{
  public void eat(){
    System.out.println("Mammal eats");  }
  public void travel(){
    System.out.println("Mammal travels");  }
  public int noOfLegs(){
    return 0;  }
  public static void main(String args[]){
    II m = new II();
    m.eat();
```

```
    m.travel();  }}
```

النتائج:

Mammal eats

Mammal travels

2. أطبع البرنامج التالي وارصد النتائج، ولاحظ استخدام مناهج الواجهة وإعادة تعريفها من جديد:

```
interface Shape {
        public double area();
        public double volume();}
public class Point implements Shape {
        static int x, y;
        public Point() {
                x = 5;
                y = 10;
        }
        public double area() {
                return 0;       }
        public double volume() {
                return 0;       }
        public static void print() {
                System.out.println("point: " + x + "," + y);         }
        public static void main(String args[]) {
                Point p = new Point();
                p.print();
```

```
        double w=p.area();
        double v=p.volume();
        System.out.println("area: " + w + "   volume is: " + v);  }}
```

المخرجات:

point: 5,10

area: 0.0   volume is: 0.0

المختبر التاسع: برمجة الاستثناءات

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

1. **التعرف على الاستثناءات كمفهوم جديد في جافا.**
2. **تطبيق قذف الاستثناء والتقاطه من خلال برامج تطبيقية منوعة؟**
3. **إعطاء أمثلة برمجية منوعة على برمجة وأنواع الاستثناءات.**
4. **أن يقوم الطالب بإنشاء الاستثناء الخاص به.**

ثانياً: عزيزي الطالب:

1. **أطبع البرنامج التالي في احد مترجمات جافا وقم بتنفيذه، ولاحظ استخدام الاستثناءات في عملية القسمة لتفادي القسمة على صفر (مثال تطبيقي القسمة على صفر):**

```
public class Dividezero {
public static int devide (int x, int y) throws Exception {
        if (y==0) throw new Exception ("can't divide");
        int z =x/y;
        return z;     }
  public static void main( String args[] )   {
        try {
                int z= Dividezero.devide(4,0);        }
        catch (Exception e) {
                System.out.println("there is an error,,,,"+e.getMessage());       }
  } }
```

المخرجات:

there is an error,,,,can't divide

3. نفس المثال السابق ولكن تم استدعاء المنهج أكثر من مرة، لاحظ تنفيذ جمل الاستثناء عند الاستدعاء الثاني للمنهج وعدم تنفيذ جملة الاستدعاء الثالث:

```
public class EX1 {
        float a;
        float b;
         public static float div1(float a, float b)throws Exception{
 float r;
 if (b==0){
        throw new Exception ("invalid number"); }
 r=a/b;
 return r; }
    public static void main(String[] args) {
   EX1 taroub=new EX1 ();
   try{           System.out.println("Division Okey:  "+EX1.div1(45,15));
        System.out.println("Invalid division....See the exception"+EX1.div1(45,0));
        System.out.println("Hello World!"+EX1.div1(45,3));    }
   catch (Exception e){    System.out.println("Invalid division....See the exception:
"+"\n"+e.getMessage());
   }}}
```

المخرجات:

```
 Division Okey:  3.0
Invalid division....See the exception:
invalid number
```

المختبر العاشر: برمجة الإدخال والإخراج

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

1. تعرف أساسيات الإدخال والإخراج في جافا.
2. كتابة برامج تطبيقية منوعة فيها عمليات القراءة والكتابة للـ Console.
3. تنفيذ برامج تطبيقية منوعة تتعامل مع ملفات الإدخال والإخراج.

ثانياً: عزيزي الطالب:

1. **مثال تطبيقي لتعريف ملف نصي والكتابة باستخدام مناهج أصناف الكتابة:**

```
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.File;
import java.io.Writer;
import java.io.FileNotFoundException;
import java.io.IOException;
public class WriteTextFileExample {
public static void main(String[] args) {
Writer writer = null;
 try {
String text = "This is a text file";
File file = new File("write.txt");
writer = new BufferedWriter(new FileWriter(file));
writer.write(text);
} catch (FileNotFoundException e) {
```

```
e.printStackTrace();
} catch (IOException e) {
e.printStackTrace();
} finally {
try {
if (writer != null) {
writer.close();}
} catch (IOException e) {
e.printStackTrace();}}}}
```

ينتج من تنفيذ هذا البرنامج إنشاء ملف write.txt وطباعة النص This is a text file في الملف باستخدام المنهج writer.write(text).

2. مثال تطبيقي لقراءة النص This is a text file الذي تم تخزينه في الملف النصي write.txt في التطبيق الأول :

```
import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.FileNotFoundException;
import java.io.IOException;
public class ReadTextFileExample {
public static void main(String[] args) {
File file = new File("write.txt");
StringBuffer contents = new StringBuffer();
BufferedReader reader = null;
 try {
```

```
reader = new BufferedReader(new FileReader(file));
String text = null;
 // repeat until all lines is read
while ((text = reader.readLine()) != null) {
contents.append(text)
.append(System.getProperty(
"line.separator"));}
} catch (FileNotFoundException e) {
e.printStackTrace();
} catch (IOException e) {
e.printStackTrace();
} finally {
try {
if (reader != null) {
reader.close();}
} catch (IOException e) {
e.printStackTrace();}}
 // show file contents here
System.out.println();
System.out.println(contents.toString());}}
```

3. مثال تطبيقي لقراءة اسم طالب وعلامته في 5 مقررات وحساب الوسط الحسابي(البرنامج يوضح الاستثناءات وبرمجة الإدخال باستخدام الصنف BufferedReader):

```
import java.io.BufferedReader;
```

```
import java.io.IOException;
import java.io.InputStreamReader;
public class ReadUserInput2 {
        public static void main(String[] args) throws IOException {
double grades[]=new double[5];
double grade;
String g=new String();
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("enter student name:");
g=br.readLine();
        System.out.print("give five grades marks: ");
                for(int i=0;i<grades.length;i++)
      try{              System.out.println("enter grade for student#"+(i+1)+" :");
            grade=Double.parseDouble(br.readLine().toString());
            grades[i]=grade;
                }catch(Exception e){
        System.out.println("error happened at grade #"+(i+1)+"!");
             i--;}
          double sum=0.0;
        System.out.println("You have entered the following grades:");
          for(int i=0;i<grades.length;i++)    {              System.out.println(grades[i]);
            sum+=grades[i];          }    System.out.println("name="+g);
          System.out.println("Average="+sum/grades.length);}}
```

المختبر الحادي عشر: Multithreading

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

1. **إنشاء وتعريف القنوات المتعددة**
2. **التعامل مع القنوات المتعددة والتعرف على مشكلة التزامن وكيفية حلها.**
3. **تمكين الطالب من تحويل صنف إلى صنف قناة.**

ثانياً:عزيزي الطالب:

1. **أطبع البرنامج التالي في احد مترجمات جافا وقم بتنفيذه، ولاحظ استخدام تعدد القنوات ومناهج thread.**

```
class xyz implements Runnable
{ public void run()
{ int i;
for (i=1;i<5;i++)
{ System.out.print(i);
try{Thread.sleep(1000);}catch(Exception e){}}}}
class Th1
{
public static void main(String ar[])
{ xyz k;Thread a,b;
k=new xyz();a=new Thread(k);b=new Thread(k);
a.start();b.start();
System.out.print("X");}}
```

المخرجات : X11223344 (or 1X1223344 or 11X223344).

.

2. **اكتب برنامجا بلغة جافا مستخدما مفهوم Multithreading تكون نتائجه كالتالي:**
1X23Y1425345

احد الحلول المقترحة:

```
 class xyz implements Runnable
{ public void run()
{ int i;
for (i=1;i<6;i++)
{ System.out.print(i);
try{Thread.sleep(1000);}catch(Exception e){}}}}
class pqr implements Runnable
{ public void run()
{ xyz t;Thread c;
System.out.print("X");
try{Thread.sleep(3000);}catch(Exception e){}
System.out.print("Y");
t=new xyz();c=new Thread(t);c.start();}}
class kapil{
public static void main(String ar[])
{ xyz k;pqr g;Thread a,b;
k=new xyz();a=new Thread(k);
g=new pqr();b=new Thread(g);
a.start();b.start();}}
```

المختبر الثاني عشر: استخدام الواجهات الرسومية والحزم swing + awt

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

1. **فتح نافذة داخل إطار رسومي.**
2. **التعرف على أبعاد الشاشة وتطبيق استخدامها.**
3. **استخدام نافذة رسومية للرسم والكتابة عليها.**
4. **استخدام الأشكال الهندسية (ثنائية البعد وثلاثية البعد)، والألوان وأنواع الخطوط وتطبيقها في واجهة رسومية.**

ثانيا: عزيزي الطالب:

1. **مثال بسيط لإنشاء واجهة رسومية (إطار رسومي):**

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class FrameTest {
  public static void main(String args[]){
    JFrame a=new JFrame("طروب  انتهى الفصل الثاني");
    a.setSize(600,600);
    a.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    a.setVisible(true);     }}
```

2. مثال تطبيقي حول استخدام الأزرار والاستجابة للأحداث (يمكن اعتبار البرنامج تطبيق على كافة مهارات جافا) :

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class Frame3 extends JFrame implements ActionListener{
  JLabel answer = new JLabel("");
  JPanel pane = new JPanel(); // create pane object
  JButton pressme = new JButton("Press Me");
  Frame3()   // the constructor  {
    super("Event Handler Demo"); setBounds(100,100,300,200);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    Container con = this.getContentPane(); // inherit main frame
    con.add(pane); pressme.setMnemonic('P'); // associate hotkey
```

```
  pressme.addActionListener(this);   // register button listener
  pane.add(answer); pane.add(pressme); pressme.requestFocus();
  setVisible(true); // make frame visible  }
// here is the basic event handler
public void actionPerformed(ActionEvent event)  {
  Object source = event.getSource();
  if (source == pressme)    {
    answer.setText("Button pressed!");
    JOptionPane.showMessageDialog(null,"I love java!","Message Dialog",
    JOptionPane.PLAIN_MESSAGE); setVisible(true);  // show something    }  }
public static void main(String args[]) {new Frame3();}}
```

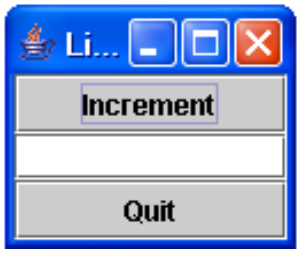
النتائج:

3. مثال تطبيقي حول الواجهة ActionListener الذي يوضح عملية إدخال عدد ما وبعد الضغط على زر زيادة يتم زيادة دورية إلى حين الضغط على زر خروج من النافذة:

```
import java.awt.BorderLayout;
  import java.awt.event.ActionEvent;
  import javax.swing.JButton;
```

```
 import javax.swing.JTextField;
import java.awt.Container;
 import java.awt.event.ActionListener;
import javax.swing.JFrame;
   public class ListeningExample implements ActionListener {
     private JButton incr;
     private JButton quit;
     private JTextField field;
       private ListeningExample(JButton i, JButton q, JTextField f) {
        incr = i; quit = q; field = f;       }
    public void actionPerformed(ActionEvent e) {
       if(e.getSource() == incr) {
          int i = Integer.parseInt(field.getText());
          field.setText("" + (i + 1));
       } else if(e.getSource() == quit) {
          System.exit(0);          }      }
      public static void main(String[] args) {
       JButton incrButton = new JButton("Increment");
       JButton quitButton = new JButton("Quit");
       JTextField numberField = new JTextField();
       ListeningExample listener = new ListeningExample(incrButton,
          quitButton, numberField);
       incrButton.addActionListener(listener);
       quitButton.addActionListener(listener);
         JFrame frame = new JFrame("Listening Example");
```

```
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container contents = frame.getContentPane();
        contents.add(incrButton, BorderLayout.NORTH);
        contents.add(numberField, BorderLayout.CENTER);
        contents.add(quitButton, BorderLayout.SOUTH);
        frame.pack();
        frame.setVisible(true);      }  }
```



❖ حاول عزيزي الطالب إجراء التعديلات على نافذة المخرجات(تغيير نوع الخط للنصوص وتغيير موقع زر **Quit** إلى يسار الشاشة).

4. مثال تطبيقي حول استخدام كائنات نصية **labels**:

```
import java.awt.*;
 import java.awt.event.*;
  // Java extension packages
 import javax.swing.*;
public class LabelTest extends JFrame {
private JLabel label1, label2, label3;
// set up GUI
public LabelTest(){
super( "Testing JLabel" );
// get content pane and set its layout
```

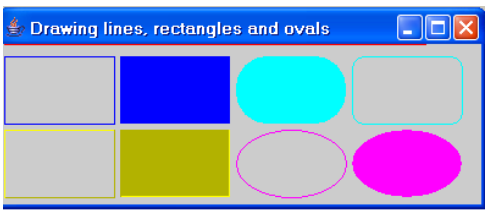
```
Container container = getContentPane();
container.setLayout( new FlowLayout() );
// JLabel constructor with a string argument
label1 = new JLabel( "نص تجريبي اول" );
label1.setToolTipText( "This is label1" );
container.add( label1 );
// JLabel constructor with string, Icon and
// alignment arguments
Icon bug = new ImageIcon( "bird.jpg" );
label2 = new JLabel( "نص تجريبي الثاني",
bug, SwingConstants.LEFT );
label2.setToolTipText( "This is label2" );
container.add( label2 );
// JLabel constructor no arguments
label3 = new JLabel();
label3.setText( "Label with icon and text at bottom" );
label3.setIcon( bug );
label3.setHorizontalTextPosition( SwingConstants.CENTER );
label3.setVerticalTextPosition( SwingConstants.BOTTOM );
label3.setToolTipText( "This is label3" );
container.add( label3 );
setSize( 275, 170 );
setVisible( true );}
// execute application
```

```
public static void main( String args[] ){
LabelTest application = new LabelTest();
application.setDefaultCloseOperation(
JFrame.EXIT_ON_CLOSE );}
} // end class LabelTest
```

5. مثال تطبيقي لرسم أشكال هندسية بتعبئة وبدون تعبئة، مع طباعة نص في النافذة:

```
import java.awt.*;
 import java.awt.event.*;
  // Java extension packages
 import javax.swing.*;
public class LinesRectsOvals extends JFrame {
// set window's title bar String and dimensions
public LinesRectsOvals(){
super( "Drawing lines, rectangles and ovals" );
setSize( 400, 165 );
setVisible( true );}
// display various lines, rectangles and ovals
public void paint( Graphics g ){
// call superclass's paint method
super.paint( g );
g.setColor( Color.red );
g.drawLine( 5, 30, 350, 30 );
g.setColor( Color.blue );
g.drawRect( 5, 40, 90, 55 );
```

```
g.fillRect( 100, 40, 90, 55 );

g.setColor( Color.cyan );

g.fillRoundRect( 195, 40, 90, 55, 50, 50 );

g.drawRoundRect( 290, 40, 90, 55, 20, 20 );

g.setColor( Color.yellow );

g.draw3DRect( 5, 100, 90, 55, true );

g.fill3DRect( 100, 100, 90, 55, false );

g.setColor( Color.magenta );

g.drawOval( 195, 100, 90, 55 );

g.fillOval( 290, 100, 90, 55 );

g.setColor(Color.green);

g.drawString(" 600 ,500 ,"انتهى الفصل الثاني");}

// execute application

public static void main( String args[] ){

LinesRectsOvals application = new LinesRectsOvals();

application.setDefaultCloseOperation(

JFrame.EXIT_ON_CLOSE );}

}  // end class LinesRectsOvals
```

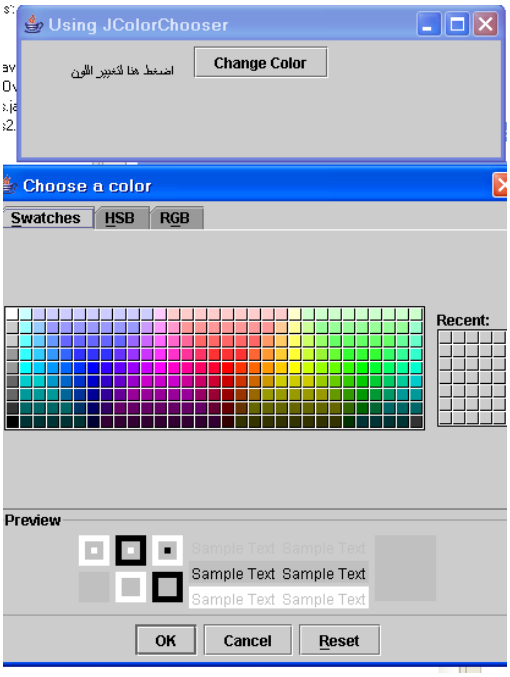


6. مثال تطبيقي لاستخدام الصنف JColorChooser ومناهجه:

```
import java.awt.*;
 import java.awt.event.*;
  // Java extension packages
 import javax.swing.*;
public class ShowColors2 extends JFrame {
private JButton changeColorButton;
private Color color = Color.lightGray;
private Container container;
// set up GUI
public ShowColors2(){
super( "Using JColorChooser" );
container = getContentPane();
container.setLayout( new FlowLayout() );
// set up changeColorButton and register its event handler
changeColorButton = new JButton( "Change Color" );
changeColorButton.addActionListener(
// anonymous inner class
new ActionListener() {
// display JColorChooser when user clicks button
public void actionPerformed( ActionEvent event ){
color = JColorChooser.showDialog(
ShowColors2.this, "Choose a color", color );
// set default color, if no color is returned
```

```
if ( color == null )
color = Color.lightGray;
// change content pane's background color
container.setBackground( color );}
}  // end anonymous inner class
); // end call to addActionListener
container.add( changeColorButton );
setSize( 400, 130 );
setVisible( true );}
public void paint( Graphics g ){
// call superclass's paint method
super.paint( g );
g.drawString("45,60 ,"اضغط هنا لتغيير اللون");}
// execute application
public static void main( String args[] ){
ShowColors2 application = new ShowColors2();
application.setDefaultCloseOperation(
JFrame.EXIT_ON_CLOSE );}
}  // end class ShowColors2
```

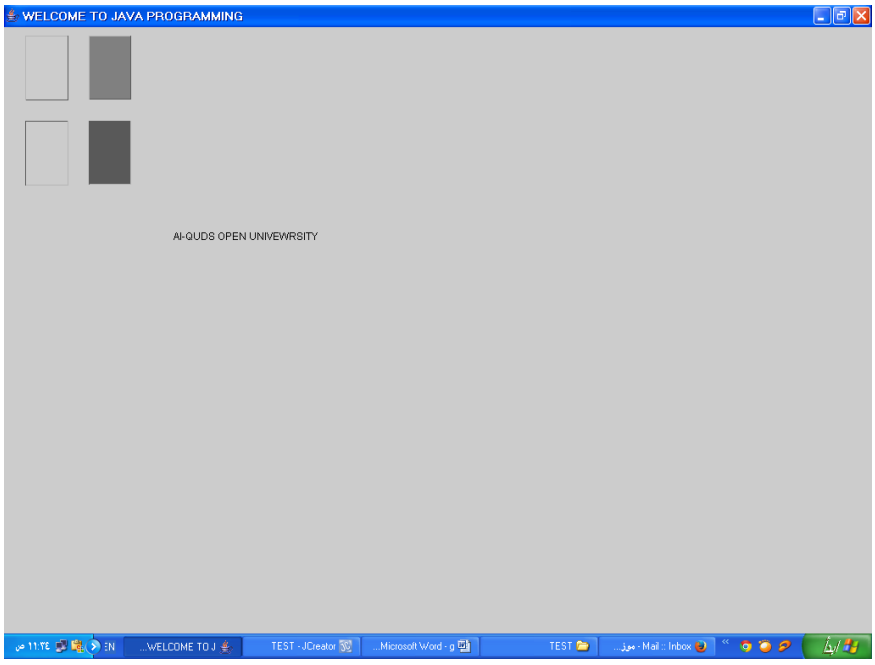
النتائج:



7. مثال تطبيقي بسيط على إنشاء JFrame ورسم أشكال هندسية داخله: عزيزي الطالب نفذ البرنامج وارصد النتائج، قم بتغيير أبعاد الكائنات الرسومية :

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
 public class TEST {
   public static void main(String[] args) {
   JFrame j=new JFrame();
   j.setBackground(Color.BLUE);
   j.setTitle("WELCOME TO JAVA PROGRAMMING");
   j.setVisible(true);
```

```
j.setSize(200,400);
 A panel = new A();
   Container ContentPane = j.getContentPane();
   ContentPane.add(panel);    }}
    class A extends JPanel {
      public void paintComponent (Graphics g) {
                super.paintComponent(g);
                g.drawString ("AI-QUDS OPEN UNIVEWRSITY", 200, 250);
                g.setColor (Color.gray);
                g.draw3DRect (25, 10, 50, 75, true);
                g.draw3DRect (25, 110, 50, 75, false);
                g.fill3DRect (100, 10, 50, 75, true);
                g.fill3DRect (100, 110, 50, 75, false);          }}
```



المختبر الثالث عشر: أمثلة تطبيقية منوعة

أولاً: أهداف المختبر:

تم إعداد هذا المختبر لتمكين الطلبة من:

1. تطبيق كافة الأساسيات ومهارات جافا التي درسها في المقرر.
2. استخدام عدة مهارات ومفاهيم أساسية في تطبيق برمجي كامل لحل مسالة أو مشكلة تواجهه في الحياة اليومية.

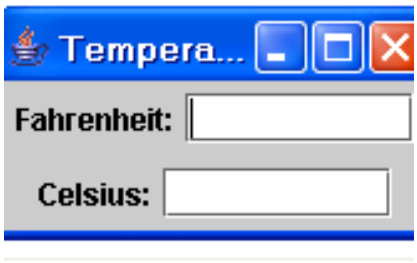
ثانيا: عزيزي الطالب:

1. اكتب برنامج اللازم لتحويل من الدرجة الفهرنهاتية إلى الدرجة المئوية باستخدام مهارات الواجهات الرسومية:

```
import java.awt.BorderLayout;
import java.awt.event.ActionEvent;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JTextField;
import java.awt.Container;
import java.awt.event.ActionListener;
import javax.swing.JFrame;
import javax.swing.JPanel;
public class TempConvert implements ActionListener {
   private JTextField fahr;
   private JTextField cels;
   private TempConvert(JTextField fahr, JTextField cels) {
     this.fahr = fahr; this.cels = cels;      }
```

```
public void actionPerformed(ActionEvent e) {
   if(e.getSource() == fahr) {
      double temp = Double.parseDouble(fahr.getText());
      double val = (temp - 32) / 1.8;
      cels.setText("" + Math.rint(val * 100) / 100);
   } else if(e.getSource() == cels) {
      double temp = Double.parseDouble(cels.getText());
      double val = 1.8 * temp + 32;
      fahr.setText("" + Math.rint(val * 100) / 100);          }       }
public static void main(String[] args) {
   JTextField fahrField = new JTextField(8);
   JTextField celsField = new JTextField(8);
   JPanel fahrPanel = new JPanel();
   fahrPanel.add(new JLabel("Fahrenheit:"));
   fahrPanel.add(fahrField);
   JPanel celsPanel = new JPanel();
   celsPanel.add(new JLabel("Celsius:"));
   celsPanel.add(celsField);
     TempConvert listener = new TempConvert(fahrField, celsField);
   fahrField.addActionListener(listener);
   celsField.addActionListener(listener);
   JFrame frame = new JFrame("Temperature Conversion");
   frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
   Container contents = frame.getContentPane();
   contents.add(fahrPanel, BorderLayout.NORTH);
```

```
        contents.add(celsPanel, BorderLayout.SOUTH);
        frame.pack();
        frame.setVisible(true);      }  }
```

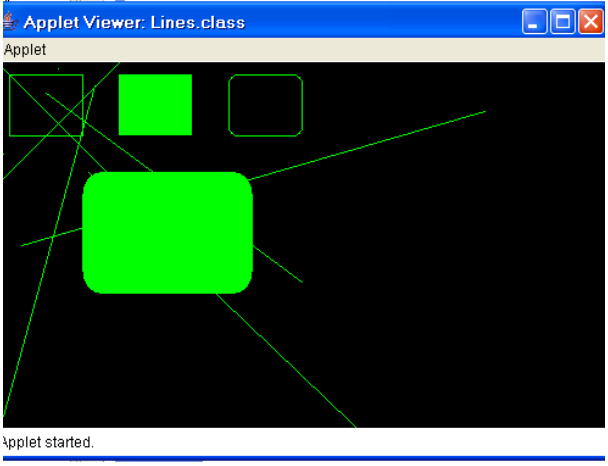


2. اكتب برنامج لإظهار النص hello world والزر ok في نافذة رسومية (JFrame) كما في الشكل التالي:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class HelloWorldGUI2 {
  private static class HelloWorldDisplay extends JPanel {
    public void paintComponent(Graphics g) {
      super.paintComponent(g);
      g.drawString( "Hello World!", 20, 30 );     }  }
  private static class ButtonHandler implements ActionListener {
```

```
public void actionPerformed(ActionEvent e) {
  System.exit(0);     }  }
public static void main(String[] args) {
   HelloWorldDisplay displayPanel = new HelloWorldDisplay();
JButton okButton = new JButton("OK");
ButtonHandler listener = new ButtonHandler();
okButton.addActionListener(listener);
JPanel content = new JPanel();
content.setLayout(new BorderLayout());
content.add(displayPanel, BorderLayout.CENTER);
content.add(okButton, BorderLayout.SOUTH);
JFrame window = new JFrame("GUI Test");
window.setContentPane(content);
window.setSize(250,100);
window.setLocation(100,100);
window.setVisible(true);   } }
```

3. اكتب برنامج لإنشاء Applet تظهر فيه خطوط ومتوازي مستطيلات (بعدة أبعاد تختارها أنت وبتعبئة وبدون تعبئة لمتوازي المستطيلات) حسب الشكل التالي:



☜ احد الحلول المقترحة:

```
import java.awt.*;
import java.applet.*;
public class Lines extends Applet {
        public void init()      {
                setBackground (Color.black);
                setForeground(Color.green);         }
        public void paint(Graphics g)        {
                g.drawLine(0, 0, 100, 100);
                g.drawLine(0, 100, 100, 0);
                g.drawLine(40, 25, 250, 180);
                g.drawLine(75, 90, 400, 400);
                g.drawLine(20, 150, 400, 40);
                g.drawLine(5, 290, 80, 19);
                g.drawLine (5, 75, 5, 75);
                g.drawLine (50, 5, 50, 5);
                g.drawRect(10, 10, 60, 50);
                g.fillRect(100, 10, 60, 50);
                g.drawRoundRect(190, 10, 60, 50, 15, 15);
                g.fillRoundRect(70, 90, 140, 100, 30, 40);
                g.drawRect(10, 10, 60, 50);
```

```
        g.fillRect(100, 10, 60, 50);
        g.drawRoundRect(190, 10, 60, 50, 15, 15);
        g.fillRoundRect(70, 90, 140, 100, 30, 40);          }}
```

يلاحظ الطالب المنهج paint والمنهج init بحيث يكون لديه القدرة على التمييز بين البرامج التطبيقية وبرامج Applet في لغة جافا.

4. اكتب برنامج بلغة جافا لإيجاد عدد مرات تكرار الحرف في نص Programming Language Java يتم إسناده داخل البرنامج إلى String، التعداد يشمل تكرار الحرف الكبير أو الصغير؟

احد الحلول المقترحة:

```
public class ahmad1 {
   public static void main(String args[]) {
      String input = "Programming Language Java";
     for (int j=65; j<=90; j++){
    int charCount = 0;
    char c=(char)j;
     for(int i =0 ; i<input.length(); i++){
        if((input.charAt(i) ==c) |(input.charAt(i) ==Character.toLowerCase(c))) {
           charCount++;          }       }
     System.out.println("count of character    "+ c + "  or "+ Character.toLowerCase(c)+ "
on String:" + charCount); } }}
```

المخرجات:

```
count of character    A  or a on String:5
count of character    B  or b on String:0
count of character    C  or c on String:0
```

count of character D or d on String:0

count of character E or e on String:1

count of character F or f on String:0

count of character G or g on String:4

count of character H or h on String:0

count of character I or i on String:1

count of character J or j on String:1

count of character K or k on String:0

count of character L or l on String:1

count of character M or m on String:2

count of character N or n on String:2

count of character O or o on String:1

count of character P or p on String:1

count of character Q or q on String:0

count of character R or r on String:2

count of character S or s on String:0

count of character T or t on String:0

count of character U or u on String:1

count of character V or v on String:1

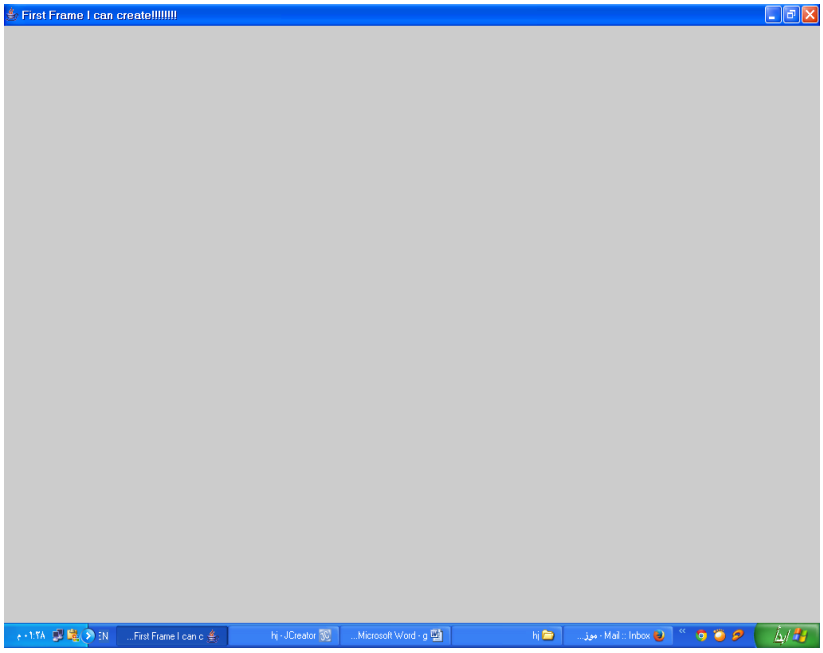
count of character W or w on String:0

count of character X or x on String:0

count of character Y or y on String:0

count of character Z or z on String:0

5. اكتب برنامج بلغة جافا لإظهار النافذة التالية:



احد الحلول المقترحة: (لاحظ تعريف الإطار داخل بنّاء الصنف)

```
import javax.swing.*;
public class FrameSimple{
        JFrame        frame1=new JFrame();
        FrameSimple ()       {
                frame1.setSize(400,700);//width and hieght
                frame1.setLocation(10,20);//x and y
                frame1.setVisible(true);
                frame1.setTitle("First Frame I can create!!!!!!!!");
                frame1.setResizable(true);   }
        public static void main(String argv[])       {
```

```
FrameSimple app=new FrameSimple ();   }}
```

6. مثال تطبيقي على استخدام مناهج الصنف Math:

```
public class Main {
  public static void main(String[] args) {
    int i = 7;
    int j = -9;
    double x = 72.3;
    double y = 0.34;
    System.out.println("i is " + i);
    System.out.println("j is " + j);
    System.out.println("x is " + x);
    System.out.println("y is " + y);
    // The absolute value of a number is equal to
    // the number if the number is positive or
    // zero and equal to the negative of the number
    // if the number is negative.
     System.out.println("|" + i + "| is " + Math.abs(i));
    System.out.println("|" + j + "| is " + Math.abs(j));
    System.out.println("|" + x + "| is " + Math.abs(x));
    System.out.println("|" + y + "| is " + Math.abs(y));
    // Truncating and Rounding functions
    // You can round off a floating point number
    // to the nearest integer with round()
     System.out.println(x + " is approximately " + Math.round(x));
```

```
System.out.println(y + " is approximately " + Math.round(y));
// The "ceiling" of a number is the
// smallest integer greater than or equal to
// the number. Every integer is its own
// ceiling.
System.out.println("The ceiling of " + i + " is " + Math.ceil(i));
System.out.println("The ceiling of " + j + " is " + Math.ceil(j));
System.out.println("The ceiling of " + x + " is " + Math.ceil(x));
System.out.println("The ceiling of " + y + " is " + Math.ceil(y));
// The "floor" of a number is the largest
// integer less than or equal to the number.
// Every integer is its own floor.
System.out.println("The floor of " + i + " is " + Math.floor(i));
System.out.println("The floor of " + j + " is " + Math.floor(j));
System.out.println("The floor of " + x + " is " + Math.floor(x));
System.out.println("The floor of " + y + " is " + Math.floor(y));
// Comparison operators
// min() returns the smaller of the two arguments you pass it
System.out.println("min(" + i + "," + j + ") is " + Math.min(i,j));
System.out.println("min(" + x + "," + y + ") is " + Math.min(x,y));
System.out.println("min(" + i + "," + x + ") is " + Math.min(i,x));
System.out.println("min(" + y + "," + j + ") is " + Math.min(y,j));

// There's a corresponding max() method
// that returns the larger of two numbers
```

```
System.out.println("max(" + i + "," + j + ") is " + Math.max(i,j));
System.out.println("max(" + x + "," + y + ") is " + Math.max(x,y));
System.out.println("max(" + i + "," + x + ") is " + Math.max(i,x));
System.out.println("max(" + y + "," + j + ") is " + Math.max(y,j));
// The Math library defines a couple
// of useful constants:
System.out.println("Pi is " + Math.PI);
System.out.println("e is " + Math.E);
// Trigonometric methods
// All arguments are given in radians
// Convert a 45 degree angle to radians
double angle = 45.0 * 2.0 * Math.PI/360.0;
System.out.println("cos(" + angle + ") is " + Math.cos(angle));
System.out.println("sin(" + angle + ") is " + Math.sin(angle));
    // Inverse Trigonometric methods
// All values are returned as radians
  double value = 0.707;
System.out.println("acos(" + value + ") is " + Math.acos(value));
System.out.println("asin(" + value + ") is " + Math.asin(value));
System.out.println("atan(" + value + ") is " + Math.atan(value));
// Exponential and Logarithmic Methods
  // exp(a) returns e (2.71828...) raised
// to the power of a.
System.out.println("exp(1.0) is "  + Math.exp(1.0));
System.out.println("exp(10.0) is " + Math.exp(10.0));
```

```
System.out.println("exp(0.0) is "  +  Math.exp(0.0));
// log(a) returns  the natural
// logarithm (base e) of a.
System.out.println("log(1.0) is "    + Math.log(1.0));
System.out.println("log(10.0) is "   + Math.log(10.0));
System.out.println("log(Math.E) is " + Math.log(Math.E));
// pow(x, y) returns the x raised
// to the yth power.
System.out.println("pow(2.0, 2.0) is "  + Math.pow(2.0,2.0));
System.out.println("pow(10.0, 3.5) is " + Math.pow(10.0,3.5));
System.out.println("pow(8, -1) is "     + Math.pow(8,-1));
// sqrt(x) returns the square root of x.
for (i=0; i < 10; i++) {
  System.out.println(
   "The square root of " + i + " is " + Math.sqrt(i));    }
// Finally there's one Random method
// that returns a pseudo-random number
// between 0.0 and 1.0;
System.out.println("Here's one random number: " + Math.random());
System.out.println("Here's another random number: " + Math.random());  }}
```

المخرجات:

j is -9

x is 72.3

y is 0.34

|7| is 7

|-9| is 9

|72.3| is 72.3

|0.34| is 0.34

72.3 is approximately 72

0.34 is approximately 0

The ceiling of 7 is 7.0

The ceiling of -9 is -9.0

The ceiling of 72.3 is 73.0

The ceiling of 0.34 is 1.0

The floor of 7 is 7.0

The floor of -9 is -9.0

The floor of 72.3 is 72.0

The floor of 0.34 is 0.0

min(7,-9) is -9

min(72.3,0.34) is 0.34

min(7,72.3) is 7.0

min(0.34,-9) is -9.0

max(7,-9) is 7

max(72.3,0.34) is 72.3

max(7,72.3) is 72.3

max(0.34,-9) is 0.34

Pi is 3.141592653589793

e is 2.718281828459045

cos(0.7853981633974483) is 0.7071067811865476

sin(0.7853981633974483) is 0.7071067811865475

acos(0.707) is 0.7855491633997437

asin(0.707) is 0.785247163395153

atan(0.707) is 0.6154085176292563

exp(1.0) is 2.7182818284590455

exp(10.0) is 22026.465794806718

exp(0.0) is 1.0

log(1.0) is 0.0

log(10.0) is 2.302585092994046

log(Math.E) is 1.0

pow(2.0, 2.0) is 4.0

pow(10.0, 3.5) is 3162.2776601683795

pow(8, -1) is 0.125

The square root of 0 is 0.0

The square root of 1 is 1.0

The square root of 2 is 1.4142135623730951

The square root of 3 is 1.7320508075688772

The square root of 4 is 2.0

The square root of 5 is 2.23606797749979

The square root of 6 is 2.449489742783178

The square root of 7 is 2.6457513110645907

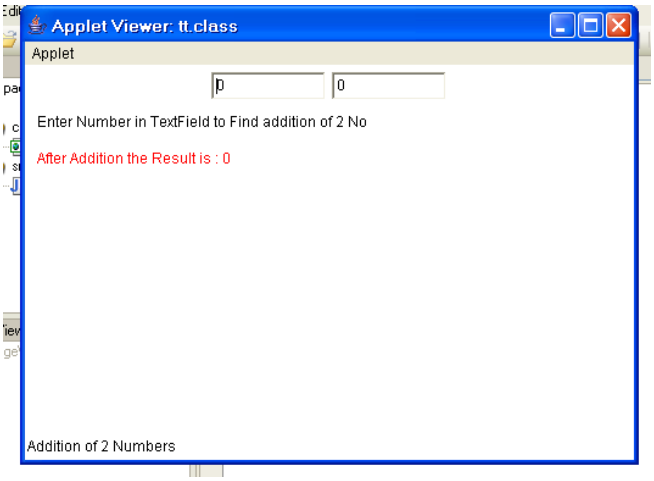
The square root of 8 is 2.8284271247461903

The square root of 9 is 3.0

Here's one random number: 0.1508387269122793

Here's another random number: 0.5343659672214046

7. اكتب برنامج بلغة جافا لاظهار النافذة التالية والتي تقوم باستقبال رقمين ثم اظهار ناتج الجمع في نافذة Applet:



احد الحلول المقترحة:

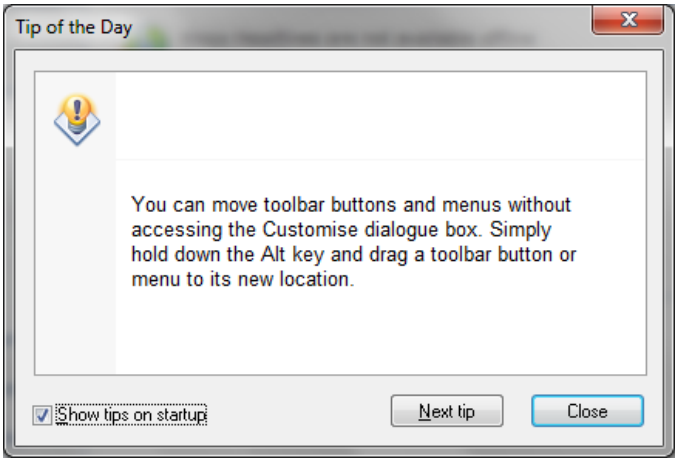
```
import java.awt.*;
import java.applet.*;
public class tt extends Applet{
    TextField T1,T2;
    public void init() {
       T1 = new TextField(10);
       T2 = new TextField(10);
       add(T1);
       add(T2);
       T1.setText("0");
       T2.setText("0");      }
     public void paint(Graphics g) {
         int a, b, result;
         String str;
```

```
    g.drawString("Enter Number in TextField to Find addition of 2 No ",10,50);
    g.setColor(Color.red);
    str=T1.getText();
    a=Integer.parseInt(str);
    str=T2.getText();
    b=Integer.parseInt(str);
    result=a+b;
    g.drawString("After Addition the Result is : "+result,10,80);
    showStatus("Addition of 2 Numbers");       }
  public boolean action(Event e, Object o){
     repaint();
     return true;       }}
```

ملحق 1

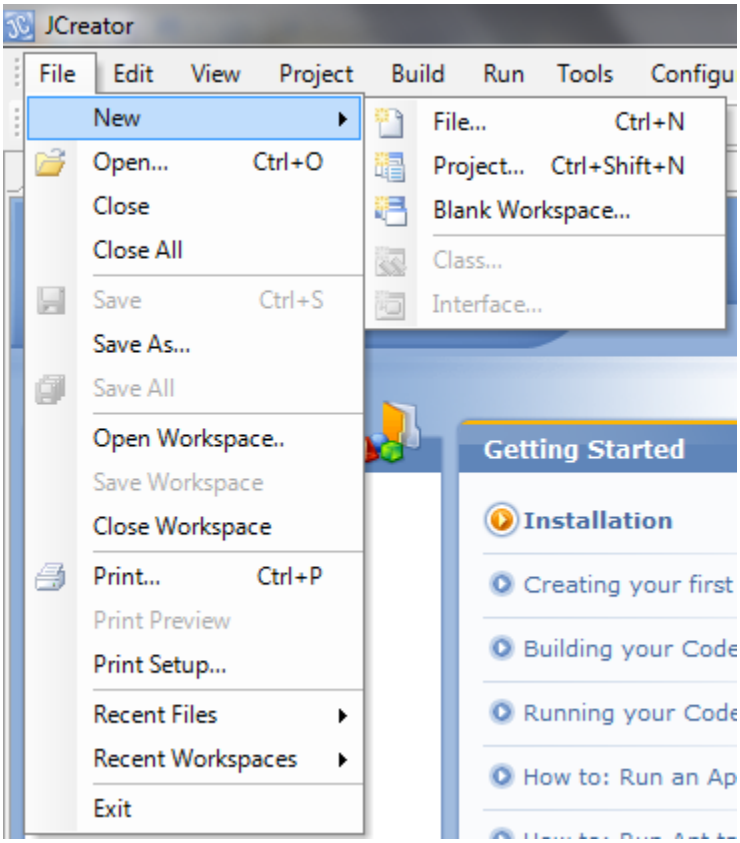
إرشادات استخدام محرر البرمجة Jcreator

أولاً: صندوق التعليمات المُساعدة للمستخدم:



- يظهر مع كل مرة يتم فيها تشغيل المحرر, و يتم فيها ذكر معلومة واحدة كل مرة تساعد المستخدم (المبرمج) على التعامل بسهولة أكبر مع المحرر.
- يمكن معرفة المعلومات التالية بالضغط على زر **next tip** أو إغلاقه من خلال **close**.
- لعدم إظهار الصندوق في المرات التالية قم بحذف الإشارة في صندوق الاختيار.

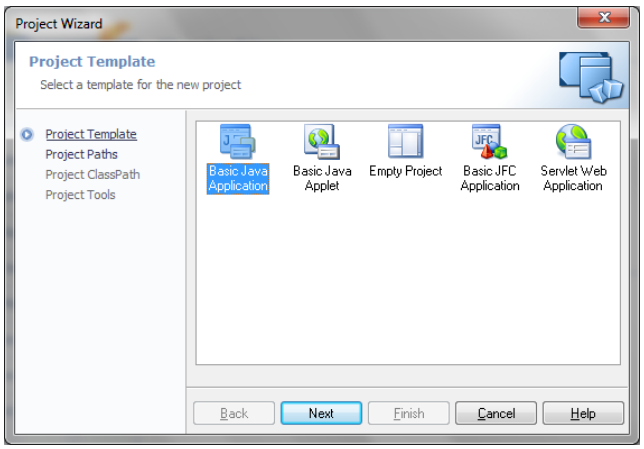
لإنشاء برنامج جديد:



- من القائمة ملف (**file**)
- نختار جديد (**New**)
- نختار مشروع (**Project**)
- ثم تتبع الخطوات الموضحة في الشرائح القادمة

لفتح مشروع سابق نستخدم الخيار فتح (**Open**).

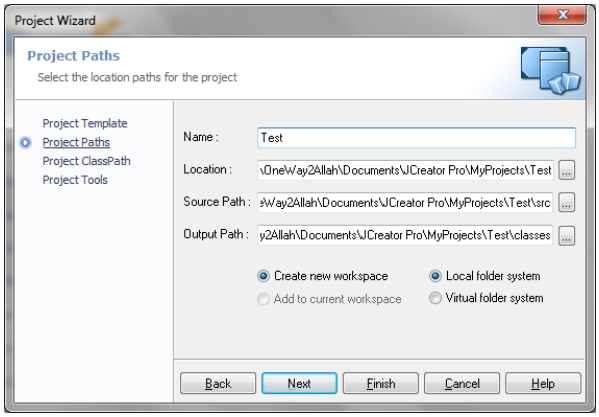
❖ لإنشاء برنامج جديد:



☜ نختار النوع Basic Java Application

☜ ثم نضغط على التالي (next)

☜ ثم نكتب اسم المشروع في الخانة الأولى.

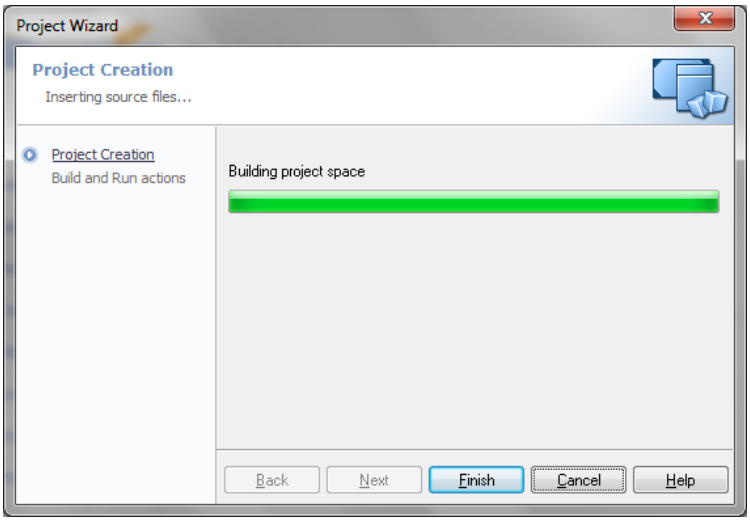


☜ ثم نضغط انتهاء (Finish)

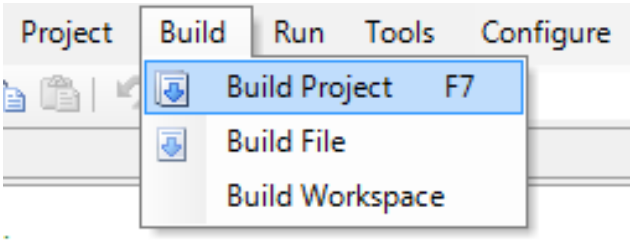
❖ لإنشاء برنامج جديد:

☜ نتابع،،،،،،، سيتطلب إنشاء المشروع بعض ثوان.

☜ اضغط بعد ذلك انتهاء مرة أخرى.



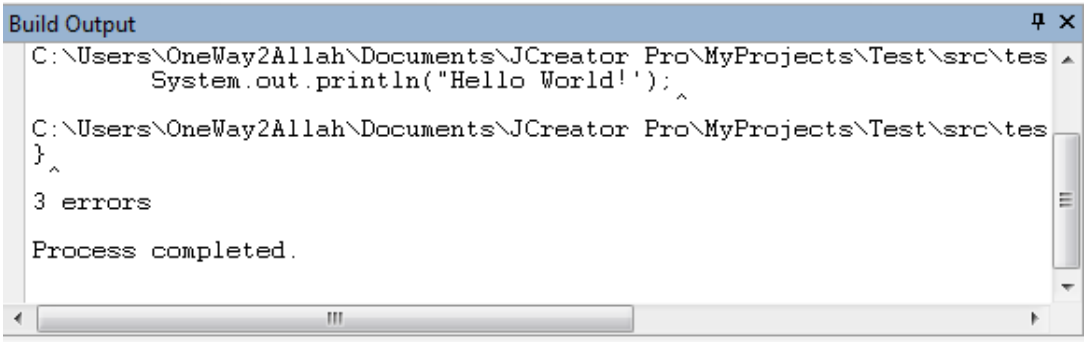
❖ لفحص الكود البرمجي من الأخطاء:

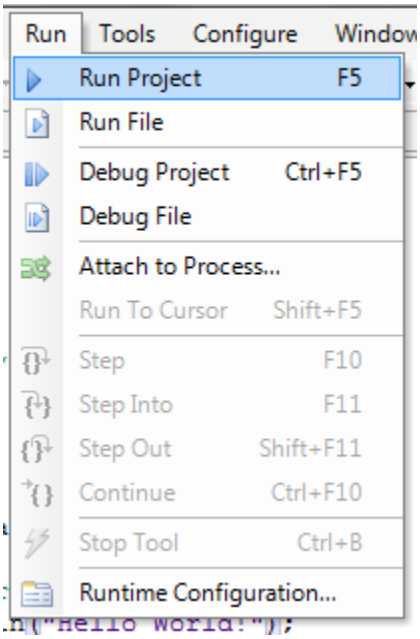


☜ نختار القائمة بناء (Build)

☜ ثم نختار الخيار (Build Project) أو نضغط المفتاح F7 من لوحة المفاتيح.

☜ حيث يتم فحص المشروع و في حال وجد أخطاءً برمجية يقوم بإظهارها في صندوق الأخطاء في أسفل الشاشة.



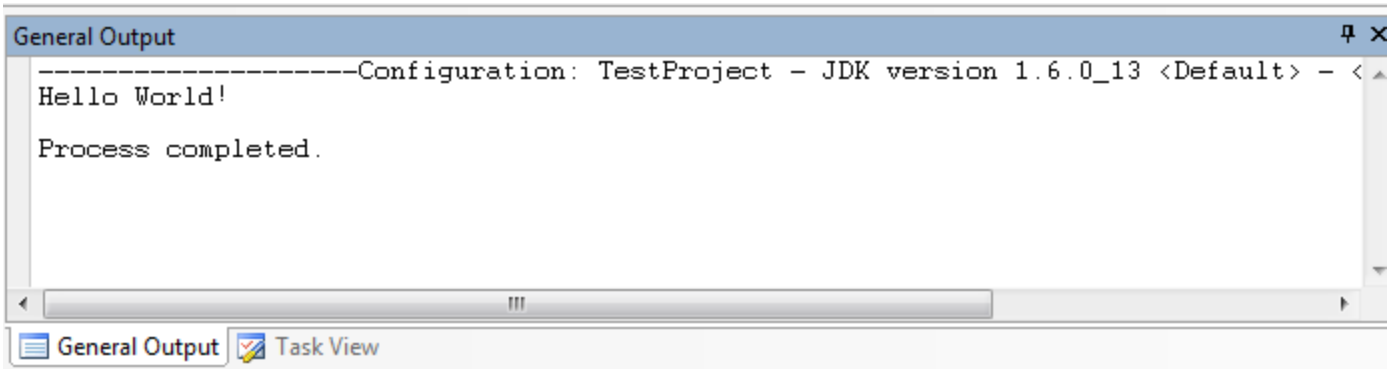


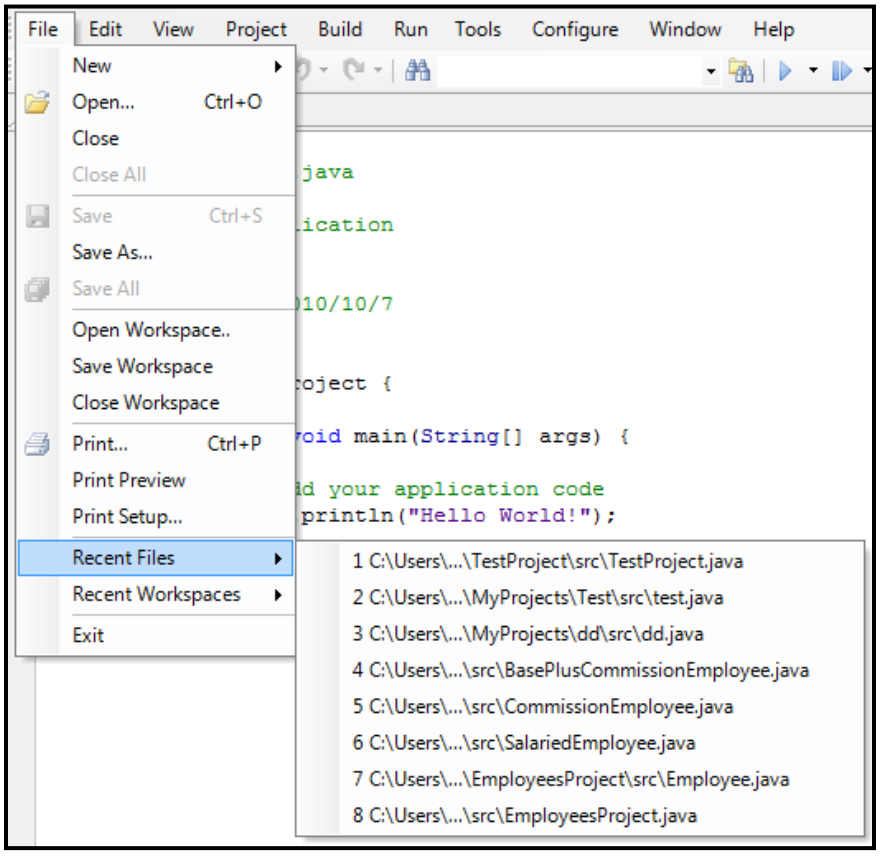
❖ لتنفيذ البرنامج و مشاهدة الناتج:

☜ نختار القائمة Run

☜ ثم نختار الخيار Run Project

❖ في حال كان البرنامج بلا أخطاء و له مخرجات ستظهر في صندوق المخرجات في أسفل ساحة العمل. و يمكن تثبيت النافذة بالطريقة ذاتها لأي نافذة أخرى.





❖ **لفتح ملف من الملفات التي تم التفاعل معها مؤخرا.**

☜ **من القائمة ملف (File)**

☜ **نختار الخيار (Recent files)**

☜ **ثم نختار الملف المقصود منهم.**

ملحق 2

إرشادات استخدام أداة التطوير NetBeans

لغة جافا هي لغة متعددة الأغراض ومتعددة المنصات تصلح لعدد كبير من التطبيقات .ومترجم جافا يقوم بإنتاج ملفات في شكل byte code وهو يختلف عن الملفات التنفيذية التي تنتج عن لغات البرمجة الأخرى مثل سي وباسكال .وتحتاج البرامج المكتوبة بلغة جافا إلى منصة في أنظمة التشغيل المختلفة لتتمكن برامجها من العمل في هذه الأنظمة .وهذه المنصة تسُمى آلة جافا الافتراضية Java Run-time.أو JVMأو إختصارا بلJava Virtual Machine .

تتوفر هذه المنصة في عدد كبير من أنظمة التشغيل، وقبل تشغيل برنامج جافا لابد من التأكد من وجودها . وكل نظام تشغيل يحتاج لآلة افتراضية خاصة به .مثلا نظام ويندوز 32 بت يحتاج لآلة افتراضية مخصصة لويندوز 32 بت، وويندوز 64 بت يحتاج لآلة افتراضية 64 بت .وهكذا

مثال لاسم ملف لتثبيت آلة جافا الافتراضية لنظام ويندوز 64 بت:

Jdk-6u16-windows-x64.exe

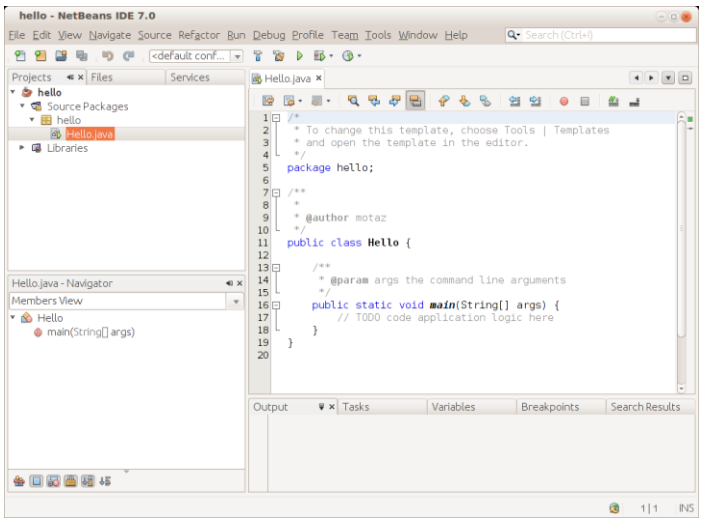
وهو يمُثل نسخة جافا 1.6 أو ما يسُمى جافا6 .

وهذه اسم حزمة تحتوي على الآلة الافتراضية لجافا 7 لنظام أوبونتو 32 بت:

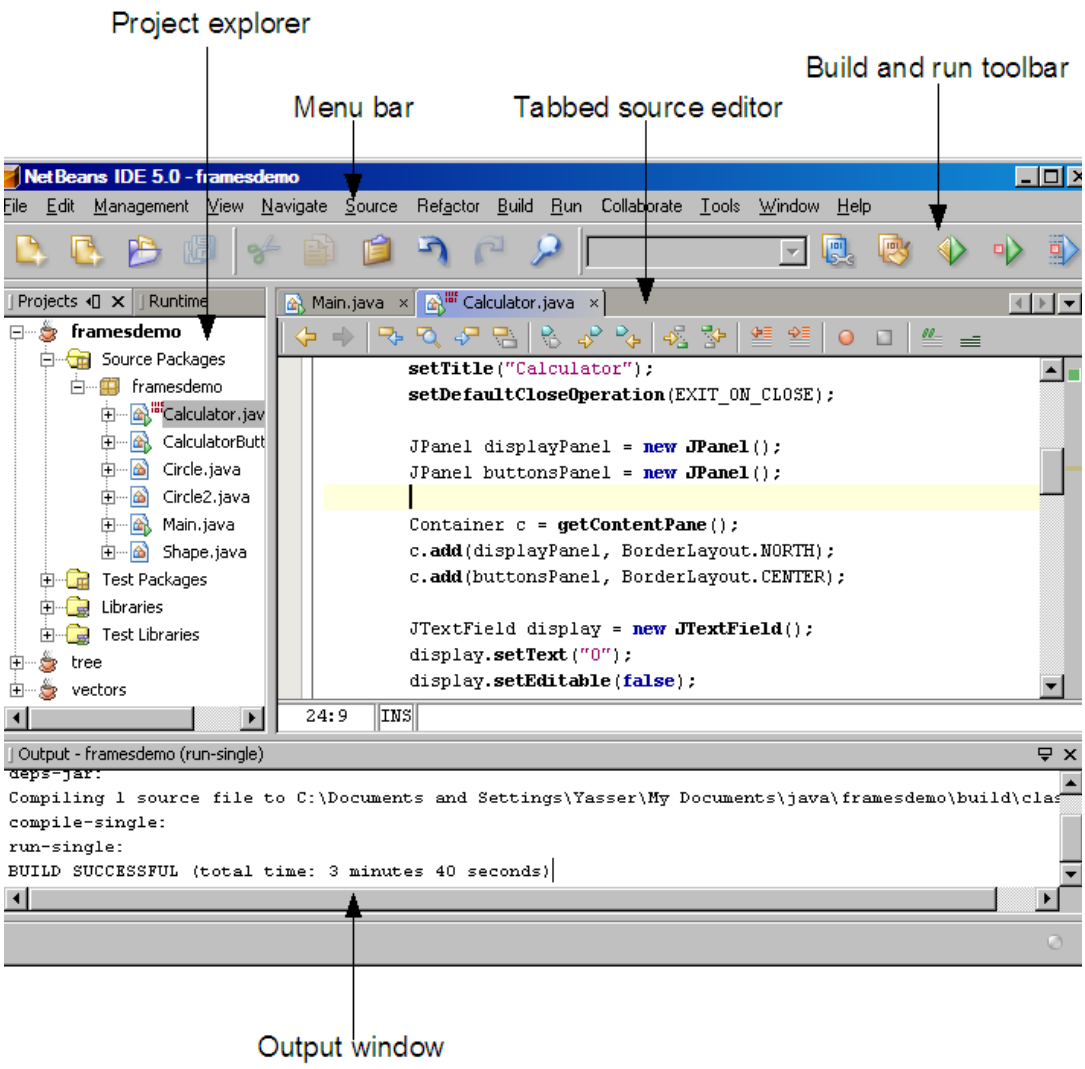
Openjdk-7-jre

يعتبر Netbeans من أفضل أدوات التطوير للغة جافا، وقد تمت كتابتها باستخدام لغة جافا نفسها بواسطة شركة أوراكل صاحبة تلك اللغة.

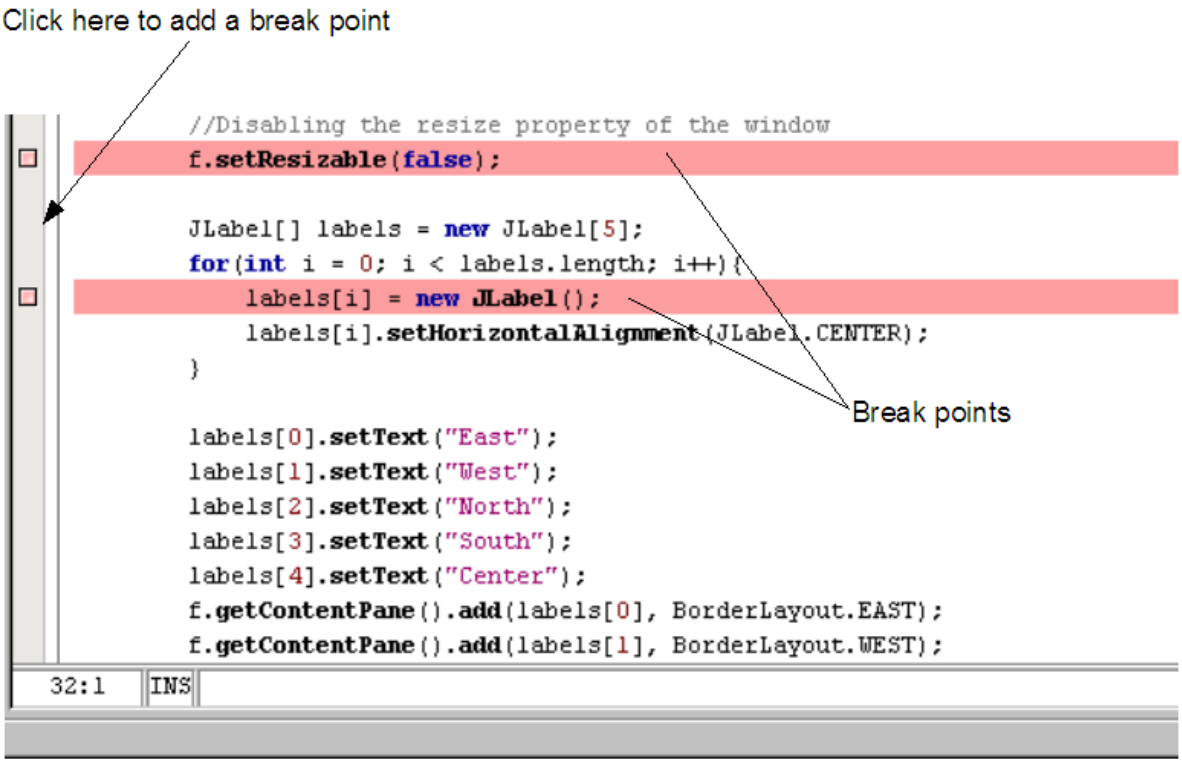
يمُكن استخدام هذه الأداة لتطوير برامج بلغات برمجة أخرى غير الجافا مثل برامج php و c++، توجد أداة تطوير أخرى مشهورة و هيEclipse ، وهي أخف وأسرع من أداة التطويرNetBeans ، وفيها إمكانية لبرمجة الأندرويد.



أول برنامج في لغة جافا في نافذة Netbeans



المكونات الرئيسية لنافذة NetBeans IDE



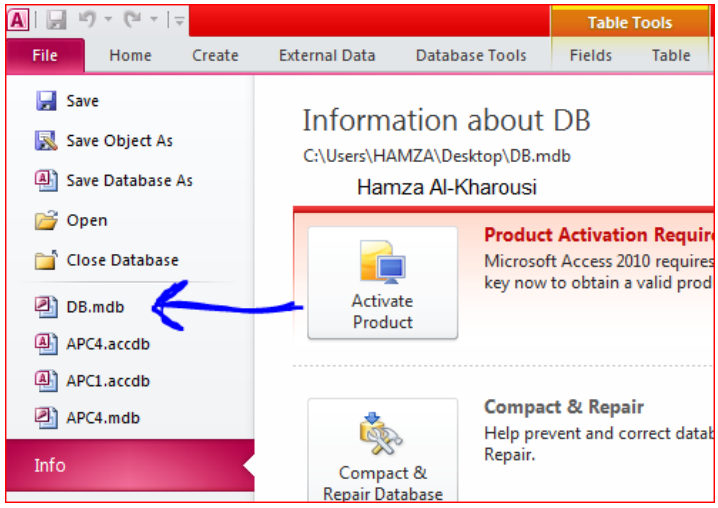
إضافة فواصل إلى برنامجك NetBeans IDE

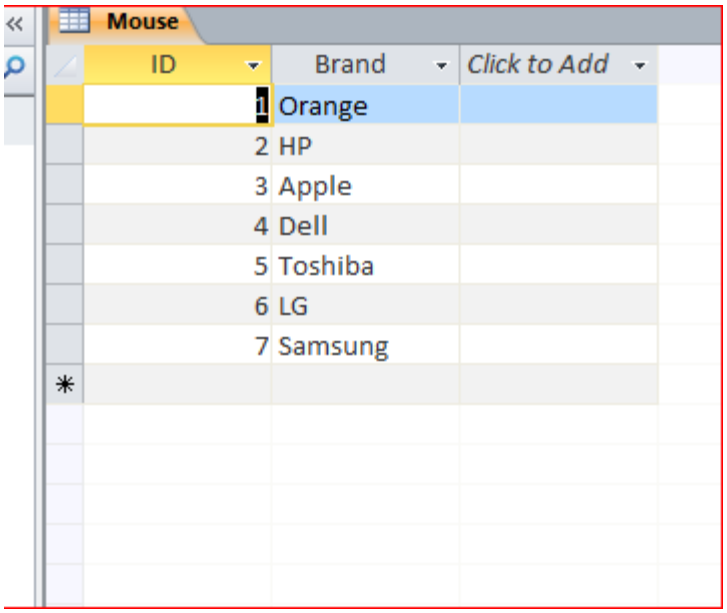
ملحق 3

إرشادات ربط برنامج جافا بقاعدة البيانات

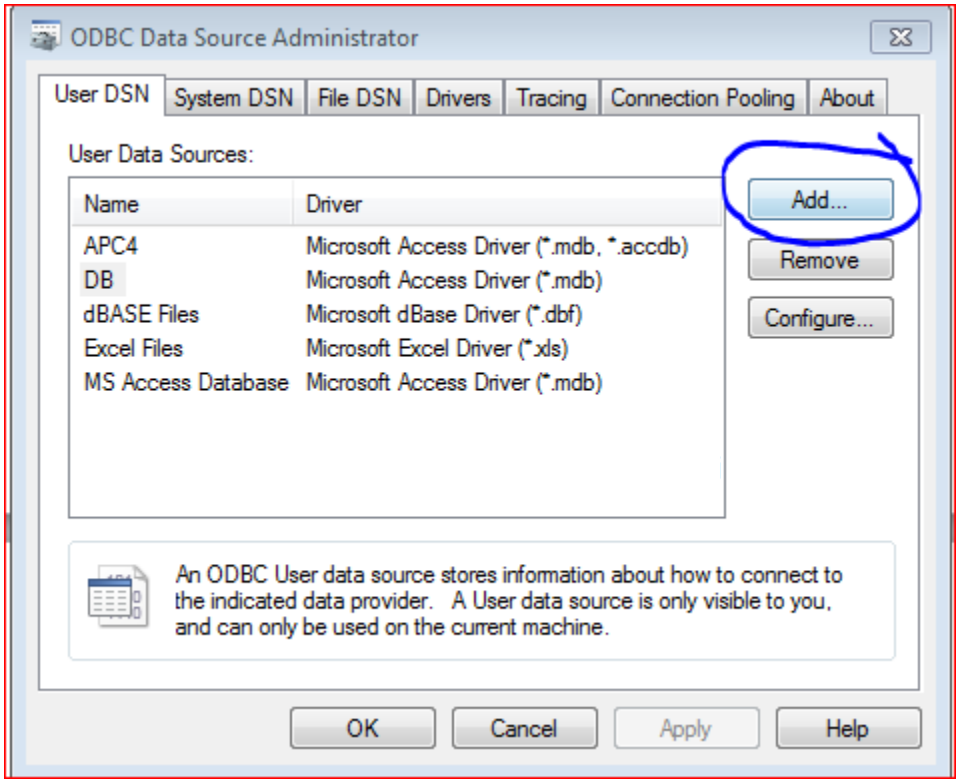
نختار ربط برنامج جافا بقاعدة البيانات Access، وذلك لتوفرها في برامج الأوفيس لدى الجميع، ويمكن أن يكون العمل في برنامج Net Beans أو أي برنامج آخر بالإضافة إلى برنامج الأكسس.

1. نفرض أن لدينا قاعدة بيانات باسم DB وفيها جدول باسم Mouses ، كما في الشكل التالي:

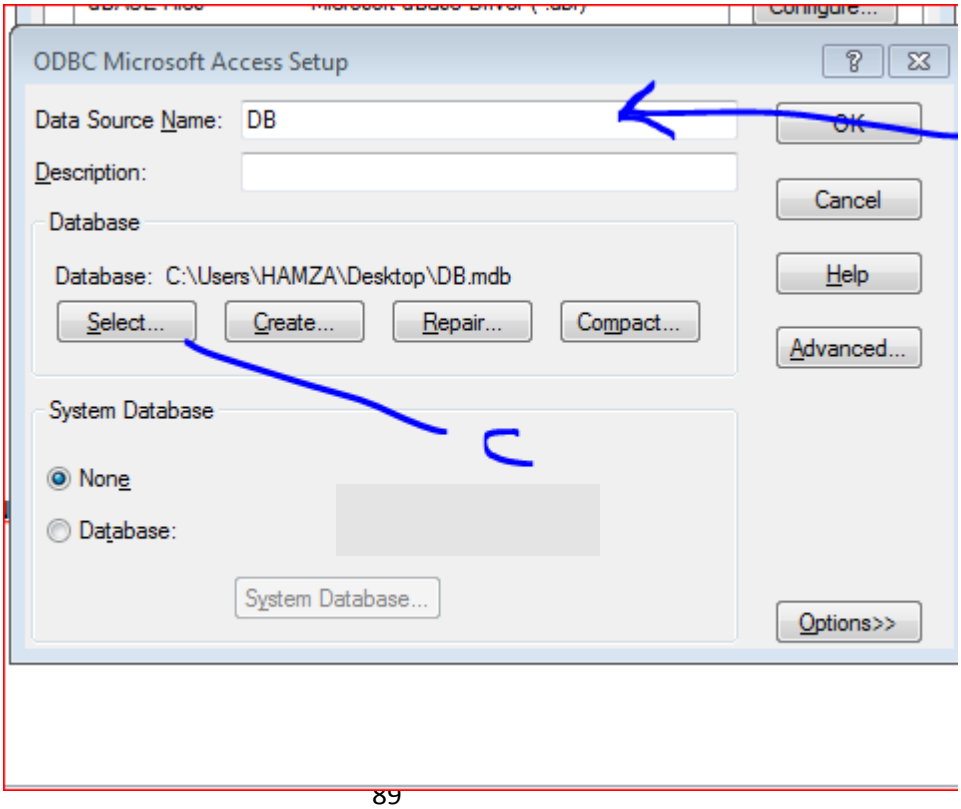




2. لآن نفتح لوحة التحكم Control Panel من زر أبدا، بعد ذلك نذهب إلى System and Security، ثم نختار Administrative tools، ونفتح Data Source (ODBC) :



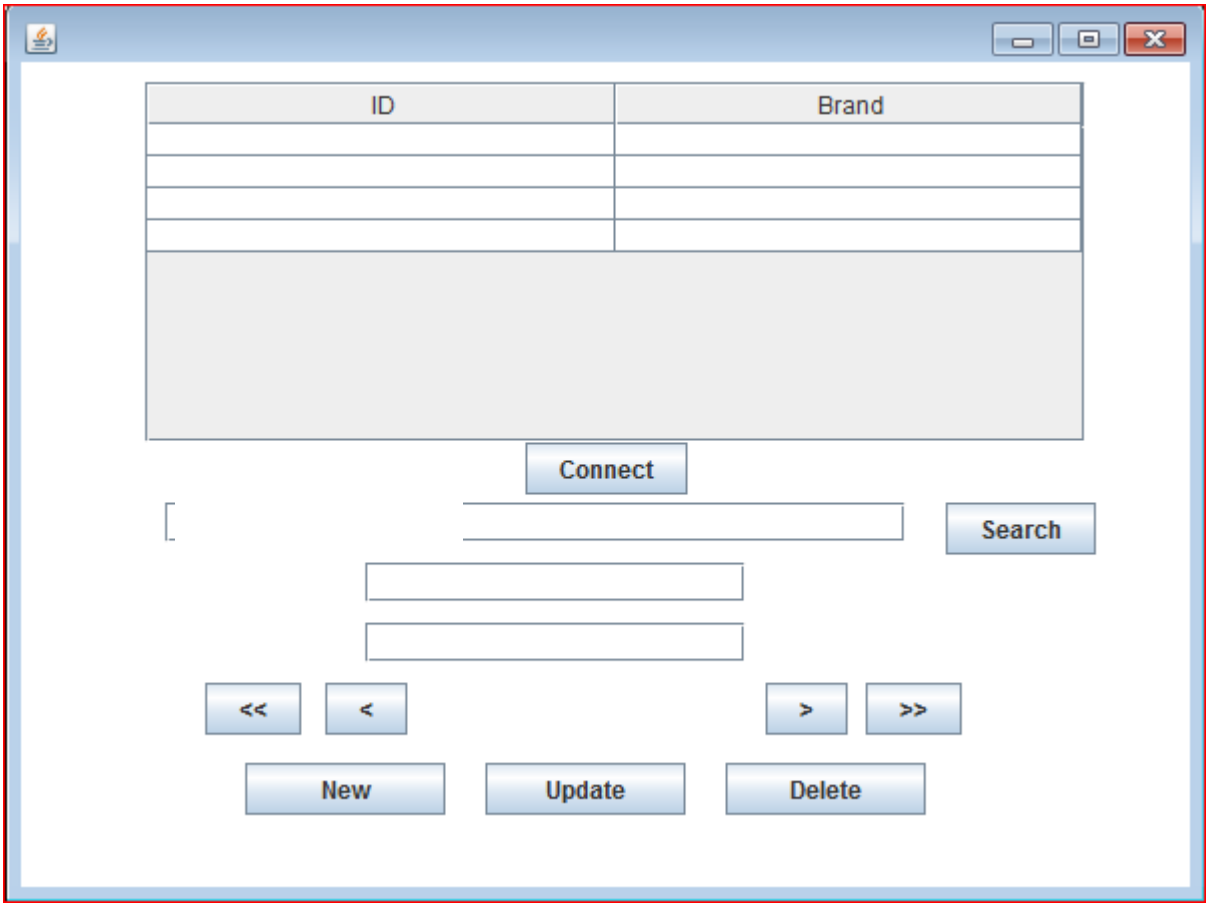
3. **نضغط على زر add لكي نضيف قاعدة البيانات الي النظام ODBC ويتم التعرف عليها من خلال NetBeans، بعد الضغط على زر add، نقوم باختيار Microsoft Access Driver (*.mdb, *accdb).**



4. **نقوم بكتابة اسم قاعدة البيانات والذي سيكون كرابط لقاعدة البيانات (الاسم يكون محفوظ في ODBC ولا يشترط أن يكون نفس اسم قاعدة البيانات الأصلي)، نقوم بتحديد مكان قاعدة البيانات،بعد ذلك نضغط ok، الآن تم إضافة قاعدة البيانات إلى ODBC.**
5. **نذهب إلى البرنامج الذي نريد توصيله بقاعدة البيانات،**

اولاً: وسنعمل على طريقة استيراد جدول وإظهاره كاملا. فيما يلي الخطوات التفصيلية:

نقوم بعمل وتصميم الإطار التالي :



1. **نتقل إلى وضعSource ، ونقوم باستدعاء المكتبة الخاصة بالتعامل مع SQL، Import java.sql*;، بعد ذلك نقوم بإضافة الدالة (Method) التالية :**

```
19      private Connection con;
20      private Statement sql;
21      private String Search;
22      private ResultSet rs;
23
24
25  private void connDB(){//Hamza Al-Kharousi
26          try {
27                  Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
28                    con=DriverManager.getConnection("jdbc:odbc:DB");
29                    sql=con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
30                    rs=sql.executeQuery("SELECT * FROM MOUSE" + Search );
31                    JOptionPane.showMessageDialog(null, "Connected");
32                    DefaultTableModel dtm= new DefaultTableModel();
33                    jTable1.setModel(dtm);
34                   dtm.addColumn("ID");                    Hamza Al-Kharousi
35                   dtm.addColumn("Brand");
36                   if(rs.next()){
37                       jTextField1.setText(rs.getString("ID"));
38                       jTextField2.setText(rs.getString("Brand"));
39                       rs.first();
40                       do{
41                           Object ary[]={rs.getInt("ID"), rs.getString("Brand")};
42                           dtm.addRow(ary);
43
44                       }while(rs.next());
45                       rs.first();
46                   }
47                   else{
48                       jTextField1.setText("");
49                       jTextField2.setText("");
50                       jTable1.removeAll();
51
52              }
53               }
54           catch (Exception ex) {
55                   JOptionPane.showMessageDialog(null, ex.getMessage());
56                   System.exit(0);
57               }
58  }
```

2. وفيما يلي شرح لجمل البرنامج: نقوم بتعريف المتغير con ليقوم بربط قاعدة البيانات التي انشأناها، نقوم بتعريف المتغير sql ليقوم بتنفيذ جمل الاستعلام، نقوم بتعريف المتغير search ، نعرف المتغير rs لحفظ نتائج الاستعلام (البيانات)، نقوم بإنشاء الدالة connDB، نستخدم try – catch للتحذير من وجود أي خطأ، Class هي مكتبة خاصة بلغة الجافا و forName هي خاصية تابعة في الكلاس نستخدمها لتقوم باستدعاء JBDC لتتم عملية ربط الجافا بقاعدة البيانات، الاتصال بقاعدة البيانات عن طريق الرابط DB الذي انشأناه، إنشاء جمل الاستعلام وتطبيقها داخل المشروع، تنفيذ جمل الاستعلام المطلوبة، تقوم بإنشاء نافذة لكي تتأكد أن الاتصال قد تم بنجاح، نعرف مخزن للجدول وحيث أن الجدول لا يتم تعبئته إلا عن طريق مصفوفة والمصفوفة تأخذ البيانات من DB وتقوم بتعبئة الجدول، نجعل الجدول يأخذ بياناته من dtm، نضيف عمودين ف dtm والذي سينقل البيانات إلى الجدول ونسمي الأول ID والثاني Brand، بعد جلب البيانات من DB وحفظها في rs ، إذا كان هنالك سطر تالي نفذ الآتي، يقوم بالبدء من الصف الأول، يقوم بإنشاء تكرار لإدخال البيانات في dtm، يقوم بالرجوع إلى الصف الأول، إذا كانت DB خالية فسيتم إبقاء الحقول المحددة خالية، في حال وجود خطأ , سيتم إظهار رسالة مبينة الخطأ، ثم أخيراً الخروج من البرنامج.

3. كود الزر Connect : عند الضغط عليه يتم استدعاء الدالة connDB وبدأ الاتصال بقاعدة البيانات:

```
private void btnconnectActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Search="";
     connDB();

}
```

4. كـود الـزر **Search:** : فـي حالـة أن المسـتخدم يريـد البحـث فـي الجـدول يقـوم بالكتابـة فـي الحقـل **JTextField3** وتنفيذ المتغير **Search**، أما إذا كان الحقل فارغ فإن المتغير **Search** يكون فارغا.

```
private void btnSearchActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if(jTextField3.getText().equals("")){
        Search="";
        connDB();
    }
    else{
        //Don't forget to leave space before where
       Search = " where Brand like '%"+ jTextField3.getText()+"%'";
        connDB();
    }
}
```

5. كود الزر> : لعرض الصف السابق.

```
private void btnPreviousActionPerformed(java.awt.event.ActionEvent evt)
    // TODO add your handling code here:
    try{
     if(rs.previous()){

         jTextField1.setText(rs.getString("ID"));
         jTextField2.setText(rs.getString("Brand"));

     }
     }
        catch(Exception ex){
         JOptionPane.showMessageDialog(null, ex.getMessage());
     }

}
```

6. كود الزر>> لعرض آخر صف في الجدول

```
private void btnLastActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
          try{
       if(rs.last()){
           jTextField1.setText(rs.getString("ID"));
           jTextField2.setText(rs.getString("Brand"));

       }
    }
    catch(Exception ex){
        JOptionPane.showMessageDialog(null, ex.getMessage());
    }
}
```

7. **كود زر >: لعرض الصف السابق**

```
private void btnNextActionPerformed(java.awt.event.ActionEvent evt) {
     // TODO add your handling code here:
    try{
     if(!(rs.isLast())&& rs.next()){
          jTextField1.setText(rs.getString("ID"));
          jTextField2.setText(rs.getString("Brand"));


     }
    }
    catch(Exception ex){
      JOptionPane.showMessageDialog(null, ex.getMessage());

    }
}
```

8. **كود زر >>: لعرض أول عمود في الجدول**

```
    private void btnFirstActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
try{
        if(rs.first()){

            jTextField1.setText(rs.getString("ID"));
            jTextField2.setText(rs.getString("Brand"));

        }
        }
           catch(Exception ex){
            JOptionPane.showMessageDialog(null, ex.getMessage());
        }

    }
```

Hamza Al-Kharousi

9. كود زر **New**: لإضافة عمود جديد:

```
    private void btnNewActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
if (btnNew.getText().equals("New")){
    jTextField1.setEnabled(true);
    jTextField2.setEnabled(true);
    jTextField1.setText("");
    jTextField2.setText("");
    btnNew.setText("Save");
}
else{ if(!(jTextField1.getText().equals(""))&& !(jTextField2.getText().equals(""))){
    try{
        String s ="insert into mouse values(?,?)";
        PreparedStatement ps= con.prepareStatement(s);
        ps.setInt(1,Integer.parseInt(jTextField1.getText()));
        ps.setString(2, jTextField2.getText());
        int t = ps.executeUpdate();
        if(t>0){
            JOptionPane.showMessageDialog(null, "Saved....");
            connDB();
            jTextField1.setEnabled(false);
            jTextField2.setEnabled(false);
        }
         btnNew.setText("New");
    }
    catch(Exception ex){
        JOptionPane.showMessageDialog(null, ex.getMessage());


        }


}
```

10. كود زر **Update**: لعمل تغير في بيانات صف من الجدول

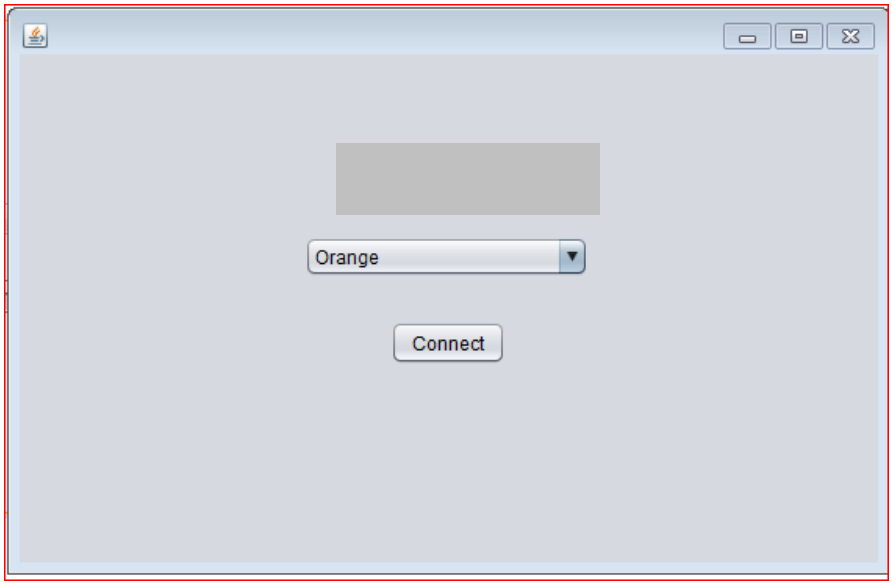
```
private void btnUpdateActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if (btnUpdate.getText().equals("Update")){
        jTextField1.setEnabled(true);
        jTextField2.setEnabled(true);
        btnUpdate.setText("Save");
    }
    else{
        try{
            String s = "Upd                              "where ID = ?";
            PreparedStatement ps = con.prepareStatement(s);
            ps.setString(1, jTextField2.getText());
            ps.setInt(2,Integer.parseInt(jTextField1.getText()));
           int t = ps.executeUpdate();
            if(t>0){
                JOptionPane.showMessageDialog(null, "Updated...");
                connDB();

            }

        }
        catch(Exception ex){
    JOptionPane.showMessageDialog(null, ex.getMessage());
        }

    }
}
```

11. **كود زر Delete: لمسح أي صف من صفوف الجدول**

```
    private void btnDeleteActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        if(jTextField1.getText().equals("")){
            JOptionPane.showConfirmDialog(null, "Can't Delete Field");
        }
        else{
            try{
            String s = "Delete from Mouse where ID = ?";
            PreparedStatement ps =con.prepareStatement(s);
            ps.setInt(1,Integer.parseInt(jTextField1.getText()));
            int t =ps.executeUpdate();
            if(t>0){
                JOptionPane.showMessageDialog(null, "Delleted....");
            connDB();

            }
            }
        catch(Exception ex){
          JOptionPane.showMessageDialog(null, ex.getMessage());

        }

        }
    }
```

ثانيا : طريقة استيراد عمود ووضعه محتواه في كومبوبوكس

نقوم بتجهيز الفريم التالي



ستكون الدالة connDB بهذه الطريقة:

```
23        public Connection con;
24         public Statement sql;
25          public ResultSet rs;
26       private void connDB(){ //* @author HAMZA
27
28          try {
29
30              Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
31              con=DriverManager.getConnection("jdbc:odbc:DB");
32              sql=con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
33
34              rs = sql.executeQuery("SELECT Brand FROM Mouse");
35
36
37             JOptionPane.showMessageDialog(null, "Database has been Connected");
38
39              rs.first();
40
41
42                  //---------------------------------
43                do{
44                    MouseComboBox.addItem(rs.getString("BRAND"));
45    }while(  rs.next() );
46
47
48              }
49           catch (Exception ex) {
50                    JOptionPane.showMessageDialog(null, ex.getMessage());
51                    System.exit(0);
52              }
53
54
55      }
```

يتم في هذه الدالة تغيير قيمة المتغير rs حيث سيقوم بجلب فقط عمود واحد فقط وهو BRAND

وتم استخدام do – while تكرار لتعبئة الكومبوبوكس.

مثال تطبيقي لربط قاعدة البيانات:

```
import java.sql.*;
public class SelectQuery {
  private Connection con=null;
  private Statement selectStatement;
  private ResultSet selectResult;
    public void connectToDB() {
    try{
      System.out.println("Loading the driver ...");
      Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
      System.out.println("Connecting to database ...");
      con = DriverManager.getConnection("jdbc:odbc:DB");
      if(con!=null)
        System.out.println("Connected to database ...");     }
    catch(SQLException sqle) {
      System.out.println(sqle.getMessage());     }
    catch(ClassNotFoundException cnfe){
      System.out.println(cnfe.getMessage());     }    }
  public void closeDB(){
      try {
        if(con != null)
          con.close();        }
```

```
      catch(SQLException sqle) {
```

```
        System.out.println(sqle.getMessage());        }   }
public void processBooksQuery(){
   String  query = "select * from students ";
   try{
      selectStatement = con.createStatement();
      System.out.println("Sending Query to database Table ...\n");
      selectResult = selectStatement.executeQuery(query);
      System.out.println("The Query  :  "+query+"\n");


      System.out.println("Query results\n");
      while(selectResult.next()) {
         System.out.println(selectResult.getString("Names")+
               "\t"+selectResult.getString("Number")+
               "\t"+selectResult.getString("Score"));        }
      selectResult.close();      }
   catch(SQLException sqle){
      System.out.println(sqle.getMessage());      }   }
public static void main(String[] args){
   SelectQuery sq = new SelectQuery();
   sq.connectToDB();
   sq.processBooksQuery();
   sq.closeDB();   }  }
```

مثال تطبيقي آخر على قواعد البيانات:

```
Purpose: A Program to execute select query using JDBC

import java.sql.*;

class SelectFromPer {

public static void main(String argv[]) {

try {

// Load the JDBC-ODBC bridge
Class.forName ("sun.jdbc.odbc.JdbcOdbcDriver");

// specify the ODBC data source's URL
String url = "jdbc:odbc:SSPer";

// connect
Connection con = DriverManager.getConnection(url,"North","Ken");

// create and execute a SELECT
Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery
("SELECT Surname,FirstName,Category FROM Per");

System.out.println("Class is SelectFromPer\n");
// traverse through results
System.out.println("Found row:");

while (rs.next()) {

// get current row values
String Surname = rs.getString(1);
String FirstName = rs.getString(2);
int Category = rs.getInt(3);

// print values
System.out.print (" Surname=" + Surname);
System.out.print (" FirstName=" + FirstName);
System.out.print (" Category=" + Category);
System.out.print(" \n");
}
```

```
// close statement and connection
stmt.close();
con.close();
} catch (java.lang.Exception ex) {

ex.printStackTrace();
}

}

}
```

ملحق 4

إصدارات وأدوات لغة وبيئة جافا

JDK Versions

1. **JDK 1.02 (1995)**
2. **JDK 1.1 (1996)**
3. **JDK 1.2 (1998)**
4. **JDK 1.3 (2000)**
5. **JDK 1.4 (2002)**
6. **JDK 1.5 (2004) a. k. a. JDK 5 or Java 5.**

JDK Editions

1. **Java Standard Edition (J2SE)**

J2SE can be used to develop client-side standalone applications or applets.

2. **Java Enterprise Edition (J2EE)**

J2EE can be used to develop server-side applications such as Java servlets and Java ServerPages.

3. **Java Micro Edition (J2ME).**

J2ME can be used to develop applications for mobile devices such as cell phones.

Java IDE Tools

1. **Borland JBuilder**
2. **NetBeans Open Source by Sun**
3. **Sun ONE Studio by Sun MicroSystems**
4. **Eclipse Open Source by IBM**

جدول المحتويات

الرقم	الموضوع	رقم الصفحة
.1	مقدمة في لغة جافا	2
.2	المختبر الأول	16
.3	المختبر الثاني	19
.4	المختبر الثالث	23
.5	المختبر الرابع	28
.6	المختبر الخامس	31
.7	المختبر السادس	34
.8	المختبر السابع	36
.9	المختبر الثامن	39
.10	المختبر التاسع	42
.11	المختبر العاشر	44
.12	المختبر الحادي عشر	48
.13	المختبر الثاني عشر	50
.14	المختبر الثالث عشر	62
.15	ملحق1: إرشادات استخدام محرر البرمجة Jcreator	78

.16	ملحق2: إرشادات استخدام محرر البرمجة NetBeans	83
.17	ملحق3: إرشادات ربط برنامج جافا بقاعدة البيانات	87
.18	ملحق 4: إصدارات وأدوات لغة وبيئة جافا	100
	جدول المحتويات	102