Agenda

---

1. while loop

2. do ...while loop

3. switch case

4. Introduction OOP

---

1. while loop

---

format:

Initialization;

while(Condition)

{

Statements;

Increment/Decrement

}

Example:

---

public class MyWhileLoop

{

public static void main(String args[])

{

int i=1;

```
while(i<=10)
{
System.out.println(i+". "+"Hello CS2!!!");
i++;
}
}
}
```

Output:

---

1. Hello CS2!!!
2. Hello CS2!!!
3. Hello CS2!!!

.....

10. Hello CS2!!!

Exercise:

Write a Java Program which uses JOPtionPane class to prompt user to enter a Number less than 15 then uses while loop to display its multiplication table.

Answer:

```
import javax.swing.*;
public class MyExerciseWhile
```

```
{
public static void main(String args[])
{
JOptionPane p=new JOptionPane();
String s=p.showInpuDialog("Enter a Number less Than 15");
int n,mult;
n=Integer.parseInt(s);
if(n<15)
{
int i=1;
while(i<=n)
{
mult=n*i;
System.out.println(n*i+"="+mult);
i++;
}
}
}
}
```

---

2. do ... while loop

Format:

```
initialization;
```

```
do
{
Statements;
Increment/Decrement;
}
while(Condition);
```

Example

```
public class MyDoWhile
{
public static void main(String args[])
{
int i=10;
do
{
System.out.println("Java Do While Loop!");
i--;
}
while(i>=1);
}
}
```

Exercise:

Write a Java Program which uses JOptionPane and do while loop concepts.

The program prompts user to enter a number N between 10 and 20.

If the number is even it displays "Hello Java" N-times otherwise

it displays "I like Java" N-times.

Answer:

=========

```
import javax.swing.*;
public class MyExerciseDoWhile {
public static void main(String args[])
{
JOptionPane op=new JOptionPane();
String s=op.showInputDialog("Number between 10 and 20");
int i,n;
n=Integer.parseInt(s);
if(n>=10&&n<=20)
{
if(n%2==0)
{
i=1;
do
{
System.out.println("Hello Java");
i++;
```

```
}
while(i<=n);
}
else
{
i=1;
do
{
System.out.println("I like Java");
i++;
}
while(i<=n);
}
}
else
{
System.out.println("Number Not Allowed");
}
}
}
```

---

3. Switch Case

-> This is not a loop

-> It uses many cases to test a condition

-> It uses break after each case

Format

Initialization;

Switch(TestValue)

{

Case Value1:

{

Statement;

Break;

}

Case Value2:

{

Statements;

Break;

}

.....

Case ValueN:

{

Statements;

Break;

}

Default:

{

```
Statements;
}
```

Example:

```
public class MySwitchExample
{
public static void main(String args[])
{
int marks=70;
switch(marks)
{
case 60:
{
System.out.println("Satisfaction");
break;
}
case 50:
{
System.out.println("Pass");
break;
}
case 40:
{
System.out.println("Fail");
```

```
break;
}
case 70:
{
System.out.println("Distinction");
break;
}
default:
{
System.out.println("Great Distinction");
}
}
}
}
```

Exercise:

Write a Java Program which prompts user
to enter two numbers and operator (+,-,*,/,%). The program
uses switch case to test the entered operator. Depending on the operator,
it computes the corresponding operation and displays the results.

Example:

10

20

*

output: 10*20=20

Agenda

---

1. Connecting Java to MS Access

2. Connecting Java to Mysql

---

1. Connecting Java to MS Access

-> We need JDBC (Java Database Connectivity) driver and ODBC (Open Database Connectivity)

connector. ODBC is used by all Microsoft Database Management Systems

-> We must import sql package.

-> To avoid exceptions we use try and catch exception handling technique.

Steps to connect Java to Database:

-> We must load driver into memory

-> We must create a connection to database using URL(Uniform Resource Locator) or path

-> We must create statement that will work with sql

-> We must execute statement to manipulate database

-> We must close connection

Example1:

```
import java.sql.*;

public class MyJavaAccess {

public static void main(String args[]){

String url;

String username,password;

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

url="jdbc:odbc:UR_CS_Java";

username="";

password="";

Connection con= DriverManager.getConnection(url,username,password);

String q1="insert into Student(ID,FName,LName,Tel,Class)
values(1,'Peter','Kamanzi',0789967123,'CS3')";

Statement st=con.createStatement();

st.execute(q1);

String q2="select * from Student";

st.execute(q2);

ResultSet rs=st.getResultSet();

while(rs.next())

{

String fname,lname;

fname=rs.getString("FName");

lname=rs.getString("LName");

int tel=rs.getInt("Tel");
```

```
System.out.println(fname+" "+lname+" "+tel);
}
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
```

---

2. Connecting to Mysql

```
import java.sql.*;
public class MyJavaMysql {
public static void main(String args[])
{
String username,password;
String url;
try
{
username="root";
password="";
url="jdbc:mysql://localhost:3306/CS2";
String driver="com.mysql.jdbc.Driver";
```

```
Class.forName(driver).newInstance();
Connection c=DriverManager.getConnection(url,username,password);
Statement st=c.createStatement();
String q1="insert into Student(Fname,Lname,Tel,Class)values('Paul','Ineza',0789867755,'CS2')";
st.execute(q1);
String q2="select * from Student";
st.execute(q2);
ResultSet rs=st.getResultSet();
while(rs.next())
{
System.out.println(rs.getString("fname")+" "+rs.getString("lname"));
}
}
catch(Exception e)
{
e.printStackTrace();
}
}
})
```

Agenda

---

1. Excise on Interface
2. Introduction to Java Applets
3. Introduction to Java Swing

---

1. Exercise on Interface:

Write a Java program which uses the concepts of Interface and Class.

The program has an interface called Exercise which has two

methods getNumber() and displaySquare(), getNumber is used

to prompt user to enter an odd number, and displaySquare computes square of the

entered number displays it on screen. The class that implements Exercise is called

InferaceImplement.

Answer:

```
public interface Exercise
{
public void getNumber();
public void displaySquare();
}
```

save this as Exercise.java

```
import javax.swing.*;
public class InterfaceImplement implements Exercise
{
int a,sq;
String s;
public void getNumber()
{
```

```
JOptionPane p=new JOptionPane();
s=p.showInputDialog("Enter a Number");
a=Integer.parseInt(s);
}
public void DisplaySquare()
{
if(a%2!=0)
{
sq=a*a;
System.out.println(sq);
}
else
{
System.out.println("Only Odd Numbers are used");
}
}
}
class MainClass
{
public static void  main(String args[])
{
InterfaceImplement i=new InterfaceImplement();
i.getNumber();
i.displaySquare();
}
```

}

Save this as InterfaceImplement.java

---

2. Introduction to Java Applet

Applets are program code that can be interpreted by web browsers.

->Applet program does not contain main method

->Applet program inherits Applet Class

->Applet is embedded in html using applet tag

=> Applet program has an important method called paint which used to display on screen.

=>Applet has class called Graphics that has many method for drawing on screen

=>All layout and events used in applet are found in AWT Package

=> There are two ways to run Applet: Using a web browser supported by Java or using AppletViewer tool foung JDK.

Example:

```
import java.awt.*;
import java.applet.*;
public class MyAppletProgram extends Applet
{
public void paint(Graphics g)
{
```

```
g.drawString("Hello Applet", 200,300);
}
}
```

save this as MyAppletProgram.java

Create a HTML file:

```
<html>
<head>
<title> My First Applet</title>
</head>
<body>
<applet code="MyAppletProgram.class" height=200 width=300>
</applet>
</body>
</html>
```

save this as applet.html

Process of running Applet:

---

1. compile applet program to create .class file
2. run html file containing applet tag.

Agenda 15-04-2016

---

1. Applet Programming

2. Swing Programming

3. Java Database Connectivity

---

1. Applet Programming

-> Applet was developed to work with graphics (especially 2D Graphics)

->All Applet classes are found in java.applet package

-> Applet programs do not have main method

-> Applet was also developed for Internet Programming

-> Applet Programs run in web browser or using appletviewer tool of JDK

Steps to run Applet

-> Write Applet Program and save it with .java extension

-> compile Applet Program to produce a .class file

-> Embed Applet in HTML using <applet> tag

->Run a .hmtl file using web browser.

Example:

```
import java.applet.*;
import java.awt.*;
/*
 <applet code="MyApplet1.class" height=200 width=200>
 </applet>
 */
```

```
public class MyApplet1 extends Applet {
public void paint(Graphics g)
{
g.setColor(Color.red);
g.drawLine(50,130,220,130);
g.setColor(Color.blue);
g.drawString("Hello CS 2 Students of Group1",50,150);
g.setColor(Color.red);
g.drawLine(50,160,220,160);
g.setColor(Color.green);
g.drawRect(50,170,172,170);
}
}
```

save this as MyApplet1.java

Your HTML code should look like the following:

```
<html>
<head>
<title> Applet Programming</title>
</head>
<body>
<applet code="MyApplet1.class" height=200 width=300>
</applet>
</body>
</html>
```

save this as Test.html

=>Test.html and MyApplet1.java must be in same folder.

=>To run your applet you double click on Test.html

---

2. Java Swing

-> Swing API was developed to help programmers to design faster and good
Graphical User Interface for small and large software applications.

-> Swing classes were written based on applet classes, for this we
add J in front of applet classes.

-> Swing Interfaces use layouts from AWT

-> Swing programs are said to be Look and Feel, to implement this
for Look: we use Swing classes and Layout Classes
for Feel: we use Event Listeners found in java.awt.event package

-> Swing programs are executed in main method

Example1:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class MySwing1 extends JApplet
{

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

```
JFrame f=new JFrame("Java Swing Program");
f.setTitle("CS2 Swing Programming");
f.setLayout(new FlowLayout());
f.setVisible(true);
f.setSize(400,500);


JMenuBar bar=new JMenuBar();
f.add(bar);


JLabel L1=new JLabel("Type Your Name:");
f.add(L1);


JTextField t1=new JTextField(10);
f.add(t1);


JButton b1=new JButton("Reset");
f.add(b1);
```

```
JMenu file=new JMenu("File");
bar.add(file);


JMenuItem fitem1=new JMenuItem("New");
JMenuItem fitem2=new JMenuItem("Close");
file.add(fitem1);
file.add(fitem2);

JMenu edit=new JMenu("Edit");
bar.add(edit);


JMenuItem eitem1=new JMenuItem("Copy");
JMenuItem eitem2=new JMenuItem("Paste");
edit.add(eitem1);
edit.add(eitem2);



}
}
```

(((basic netbeans JFrame tutorial)))

Example2:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class MySwing2 extends JApplet implements ActionListener
{
JButton b1;
JTextField t1;
public MySwing2()
{
JFrame f=new JFrame("Java Swing Program");
JLabel L1=new JLabel("Type Your Name:");
t1=new JTextField(10);
b1=new JButton("Reset");
b1.addActionListener(this);
JMenuBar bar=new JMenuBar();
JMenu file=new JMenu("File");
JMenu edit=new JMenu("Edit");
JMenuItem fitem1=new JMenuItem("New");
JMenuItem fitem2=new JMenuItem("Close");
JMenuItem eitem1=new JMenuItem("Copy");
JMenuItem eitem2=new JMenuItem("Paste");
file.add(fitem1);
```

```
file.add(fitem2);
edit.add(eitem1);
edit.add(eitem2);
bar.add(file);
bar.add(edit);
f.add(bar);
f.add(L1);
f.add(t1);
f.add(b1);
f.setTitle("CS2 Swing Programming");
f.setLayout(new FlowLayout());
f.setVisible(true);
f.setSize(400,500);
}
public void actionPerformed(ActionEvent a)
{
if(a.getActionCommand()=="Reset")
{
t1.setText("");
}
}
public static void main(String args[])
{
MySwing2 ms=new MySwing2();
}
```

}

Java Database Connectivity (JDBC)

->JDBC is a driver connecting java to almost all relational databases

-> We have to import java.sql package

=> Stepts to run Java Database Application

1. Load Driver in memory

2. Open Connection to Database

3. Create Statement

4. Create ResultSet

5. Close Connection

Java and MS Access

->We use JDBC ODBC Bridge.

-> Create DataSource

Example:

```
import java.sql.*;
public class JavaAccess1 {
public static void main(String args[])
{
```

```
String driver="sun.jdbc.odbc.JdbcOdbcDriver";

String url="jdbc:odbc:CS2DSN";

try

{

Class.forName(driver);

Connection con=DriverManager.getConnection(url,"","");

Statement st=con.createStatement();

String q="insert into Student (FName,LName,Class,Tel,Email)
values('Eric','Abimana','CS2',0788867492,'ea@gmail.com')";

String q2="select * from Student";

st.executeUpdate(q);


ResultSet rs=st.executeQuery(q2);

while(rs.next())

{

System.out.println(rs.getString("FName")+" "+rs.getString("Class")+" "+rs.getInt("Tel"));

}

}

catch(Exception e)

{

e.printStackTrace();

}

}

}
```