1. **Write any program to check multiple conditions using if statement.**

**class Donation1**

**{**

**public static void main(String args[])**

**{**

**int age=16;**

**int weight=54;**

**if(age>=18)**

**{**

**if (weight>50)**

**{**

**System.out.println("You are eligible to donate blood");**

**}**

**else**

**{**

**System.out.println("You are not eligible to donate blood");**

**}}**

**}}**

1. **Write a program to display pyramids of stars/pattern using increment/decrement.**

**Program:::**

**class star**

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

**{**

**Int I,j,row=6;**

**for (i=0:i<row;i++)**

**{**

**for (j=0;j<i;j++)**

**{ System.out.println(“\*”);**

**}**

**System.out.println()**

**}}}**

1. **Write a program to display number 1 to 50 using do-while loop.**

**Class whileex**

**{**

**Public static void main (String args[])**

**{ int i=1;**

**do**

**{ System.out.println(i);**

**i++;**

**}while (i<=50)**

**}}**

1. **Write a program to convert variable of basic data type and show result of explicit type casting.**

**public class castex**

**{**

**public static void main(String args[])**

**{**

**double d = 166.66;**

**long l = (long)d;**

**int i = (int)l;**

**System.out.println("Before conversion: "+d);**

**System.out.println("After conversion into long type: "+l);**

**System.out.println("After conversion into int type: "+i);**

**}**

**}**

1. **Write a program to create a vector with seven elements as (10,30,50,20,40,10,20).Remove element at 3rd and 4th position. Insert new element at 3rd position. Display the original and current size of vector.**

**import java.util.\*;**

**class ex**

**{**

**public static void main (String args[])**

**{Vector v = new Vector (7);**

**v.add(10);**

**v.add(30);**

**v.add(50);**

**v.add(20);**

**v.add(40);**

**v.add(10);**

**v.add(20);**

**System.out.println("the size of the vector is::"+v.size());**

**v.remove(2);**

**v.remove(3);**

**v.insertElementAt(500,2);**

**v.insertElementAt(200,3);**

**System.out.println("the size of the vector is::"+v.size());**

**System.out.println(v);**

**}}**

1. **Write a program a program to demonstrate Autoboxing and unboxing using wrapper class**

**public class WrapperExample3{**

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

**byte b=10;**

**short s=20;**

**int i=30;**

**long l=40;**

**float f=50.0F;**

**double d=60.0D;**

**char c='a';**

**boolean b2=true;**

**Byte byteobj=b;**

**Short shortobj=s;**

**Integer intobj=i;**

**Long longobj=l;**

**Float floatobj=f;**

**Double doubleobj=d;**

**Character charobj=c;**

**Boolean boolobj=b2;**

**System.out.println("---Printing object values---");**

**System.out.println("Byte object: "+byteobj);**

**System.out.println("Short object: "+shortobj);**

**System.out.println("Integer object: "+intobj);**

**System.out.println("Long object: "+longobj);**

**System.out.println("Float object: "+floatobj);**

**System.out.println("Double object: "+doubleobj);**

**System.out.println("Character object: "+charobj);**

**System.out.println("Boolean object: "+boolobj);**

**byte bytevalue=byteobj;**

**short shortvalue=shortobj;**

**int intvalue=intobj;**

**long longvalue=longobj;**

**float floatvalue=floatobj;**

**double doublevalue=doubleobj;**

**char charvalue=charobj;**

**boolean boolvalue=boolobj;**

**System.out.println("---Printing primitive values---");**

**System.out.println("byte value: "+bytevalue);**

**System.out.println("short value: "+shortvalue);**

**System.out.println("int value: "+intvalue);**

**System.out.println("long value: "+longvalue);**

**System.out.println("float value: "+floatvalue);**

**System.out.println("double value: "+doublevalue);**

**System.out.println("char value: "+charvalue);**

**System.out.println("boolean value: "+boolvalue);**

**}}**

1. **Demonstrate the use of overriding method display() using super & sub classes.**

**class Vehicle{**

**void run(){System.out.println("Vehicle is running");}**

**}**

**class Bike2 extends Vehicle{**

**void run(){System.out.println("Bike is running safely");}**

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

**Bike2 obj = new Bike2();**

**obj.run();**

**}**

**}**

1. **Develop a program to implement the multilevel inheritance.**

**class Animal**

**{**

**void eat(){**

**System.out.println("eating...");}**

**}**

**class Dog extends Animal**

**{**

**void bark()**

**{**

**System.out.println("barking...");**

**}**

**}**

**class BabyDog extends Dog**

**{**

**void weep()**

**{**

**System.out.println("weeping...");**

**}**

**}**

**class TestInheritance2**

**{**

**public static void main(String args[])**

**{**

**BabyDog d=new BabyDog();**

**d.weep();**

**d.bark();**

**d.eat();**

**}}**

1. **Design a package containing a class which defines a method to find area of rectangle. Import it in java application to calculate area of rectangle.**
2. **Create three threads and run these threads according to set priority.**

class A extends Thread

{

public void run()

{

System.out.println("Thread A started ");

for(int i=1; i<=5; i++)

{

System.out.println("From thread A: i=" +i);

}

System.out.println("Exit from A");

}

}

class B extends Thread

{

public void run()

{

System.out.println("Thread B started ");

for(int j=1; j<=5; j++)

{

System.out.println("From thread B:j=" +j);

}

System.out.println("Exit from B");

}

}

class C extends Thread

{

public void run()

{

System.out.println("Thread C started ");

for(int k=1; k<=5; k++)

{

System.out.println("From thread C: k=" +k);

}

System.out.println("Exit from C");

}

}

class ThreadPriorityDemo

{

public static void main(String args[])

{

A objA= new A( );

B objB= new B( );

C objC= new C( );

objC.setPriority(Thread.MAX\_PRIORITY);

objB.setPriority(objA.getPriority() +1);

objA.setPriority(Thread.MIN\_PRIORITY);

System.out.println("Start Thread A");

objA.start();

System.out.println("Start Thread B");

objB.start();

System.out.println("Start Thread C");

objC.start();

System.out.println("End of main Thread");

}

}

1. **Write a program to accept password from user and throw “Authentication Failure” exception if password is incorrect.**

**import java.io.BufferedReader;**

**import java.io.IOException;**

**import java.io.InputStreamReader;**

**class AuthenticationException extends Exception**

**{**

**public AuthenticationException(String message)**

**{ super(message);**

**}}**

**public class AuthenticationExcDemo {**

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

**InputStreamReader isr = new InputStreamReader(System.in);**

**BufferedReader br = new BufferedReader(isr);**

**String pwd;**

**try {**

**System.out.print("Enter password :: ");**

**pwd = br.readLine();**

**if(!pwd.equals("123"))**

**throw new AuthenticationException("Incorrect password\nType correct password");**

**else**

**System.out.println("Welcome User !!!");**

**}**

**catch (IOException e) {**

**e.printStackTrace();**

**}**

**catch (AuthenticationException a) {**

**a.printStackTrace();**

**}**

**System.out.println("BYE BYE");**

**}}**

1. **Develop a basic applet to display “Welcome to the World of Applet”.**

**import java.applet.\*;**

**import java.awt.\*;**

**Public class Appletdemo extends Applet**

**{ public void paint(Graphics g)**

**{ g.drawString(“welcome to applet program”);**

**}**

**}**

**<html>**

**<applet code = “Appletdemo.class” width=500 height=500>**

**</applet>**

**</html>**

1. **Develop an applet for drawing a human face.**

**import java.awt.\*;  
import java.applet.\*;  
public class Humanface extends Applet  
{  
public void paint (Graphics g)  
{  
g.drawOval(40, 40, 120, 150);  
g.drawOval(57, 75, 30, 20);  
g.drawOval(110, 75, 30, 20);  
g.fillOval(68, 81, 10, 10);  
g.fillOval(121, 81, 10, 10);  
g.drawOval(85, 100, 30, 30);  
g.fillArc(60, 125, 80, 40, 180, 180);  
g.drawOval(25, 92, 15, 30);  
g.drawOval(160, 92, 15, 30);  
}}**

**<html>**

**<applet code= “Humanface” width=500 height=500>**

**</applet>**

**</html>**

1. **Develop a program to draw any two of the following shapes: i)Cone ii)Cylinder iii)Cube**

import java.awt.\*;  
import java.applet.\*;  
public class Shapes extends Applet  
{  
public void paint(Graphics g)  
{  
/\*Cylinder\*/  
g.drawString("(a).Cylinder",10,110);  
g.drawOval(10,10,50,10);  
g.drawOval(10,80,50,10);  
g.drawLine(10,15,10,85);  
g.drawLine(60,15,60,85);  
/\*Cube\*/  
g.drawString("(b).Cube",95,110);  
g.drawRect(80,10,50,50);  
g.drawRect(95,25,50,50);  
g.drawLine(80,10,95,25);  
g.drawLine(130,10,145,25);  
g.drawLine(80,60,95,75);  
g.drawLine(130,60,145,75);  
}

<html>

<applet code= “shapes.class” width=500 height=500>

</applet></html>

1. **Demonstrate the use of stream classes for reading and writing bytes/Characters.**