

slip no 1 Q.1 : Write a java program to scroll the text from left to right and vice versa continuously.

```
import java.applet.Applet;

import java.awt.*;

import java.awt.Graphics;

/* <APLET CODE=slip1A.class WIDTH=400 HEIGHT=200> </APPLET>*/

public class slip1A extends Applet implements Runnable

{

String msg="Welcome to java programming.....";

Thread t=null;

public void init()

{

setBackground(Color.red);

setForeground(Color.black);

t=new Thread(this);

t.start();

}

public void run()

{

char ch;

for(;;)

{

try
```

```

{
repaint();

Thread.sleep(400);

ch=msg.charAt(0);

msg=msg.substring(1,msg.length());

msg+=ch;

}

catch(InterruptedException e)

{}

}

}

public void paint(Graphics g)

{

g.drawString(msg,10,10);

}

}

```

slip no 1 Q.2 : Write a socket program in java for chatting application.(Use Swing)

first file : client.java

```

import java.net.*;

import java.io.*;

class client

{

```

```

public static void main(String[] args) throws Exception
{
    Socket s=new Socket("localhost",500);
    System.out.println("Client send request to server");
    DataOutputStream dos=new DataOutputStream(s.getOutputStream());
    DataInputStream dis=new DataInputStream(s.getInputStream());
    InputStreamReader ir=new InputStreamReader(System.in);
    BufferedReader br=new BufferedReader(ir);
    while(true)
    {
        System.out.println("client:");
        dos.writeUTF(br.readLine());
        System.out.println("server:");
        System.out.println(dis.readUTF());
    }
}
}

```

second file server.java

```

import java.net.*;
import java.io.*;
class server
{

```

```

public static void main(String[] args) throws Exception
{
    ServerSocket ss=new ServerSocket(500);
    System.out.println("Wait for Client!");
    Socket s=ss.accept();
    System.out.println("Server Accepted Request");
    DataOutputStream dos=new DataOutputStream(s.getOutputStream());
    DataInputStream dis=new DataInputStream(s.getInputStream());
    InputStreamReader ir=new InputStreamReader(System.in);
    BufferedReader br=new BufferedReader(ir);
    while(true)
    {
        System.out.println("server:");
        System.out.println(dis.readUTF());
        System.out.println("client:");
        System.out.println(br.readLine());
    }
}
}

```

slip no 2 Q.2 : Write a java program in multithreading using applet for drawing flag.

```

import java.awt.*;
import java.applet.*;

```

```
public class slip2B extends Applet
{
    public void paint(Graphics g)
    {
        g.setColor(Color.black);
        g.fillRect(50,20,5,300);
        g.setColor(Color.black);
        g.drawRect(50,18,3,300);
        g.setColor(Color.orange);
        g.fillRect(55,20,120,30);
        g.setColor(Color.black);
        g.drawRect(55,20,118,28);
        g.setColor(Color.green);
        g.fillRect(55,80,119,30);
        g.setColor(Color.black);
        g.drawRect(55,80,117,28);
        g.setColor(Color.black);
        g.drawOval(100,50,30,30);
        g.setColor(Color.blue)
    }
}
```

slip no 3 Q.2 : Write a java program using applet for bouncing ball, for each bounce color of ball should change randomly.

```
import java.applet.*;

import java.awt.*;

import java.awt.event.*;

public class slip3B extends Applet implements MouseListener, Runnable

{

    Thread t=null;

    int x1=10, x2=10, x3=10, x4=10;

    int y1=300, y2=300, y3=300, y4=300;

    int flagx1,flagy1,flagx2,flagy2;

    int flagx3,flagy3,flagx4,flagy4;


    public void init()

    {

        addMouseListener(this);

    }

    public void mouseExited(MouseEvent me) {}

    public void mouseReleased(MouseEvent me) {}

    public void mouseEntered(MouseEvent me) {}

    public void mousePressed(MouseEvent me) {}

    public void mouseClicked(MouseEvent me) {}


    public void start()

    {

        t=new Thread(this);
```

```
t.start();  
}  
  
public void run()  
{  
    for(;;)  
    {  
        try  
        {  
            repaint();  
            if(y1<=50)  
            flagx1=0;  
            else if(y1>=300)  
            flagx1=1;  
            if(x1<=10)  
            flagy1=0;  
            else if(x1>=400)  
            flagy1=1;  
            if(y2<=50)  
            flagx2=0;  
            else if(y2>=300)  
            flagx2=1;  
            if(x2<=10)  
            flagy2=0;
```

```
else if(x2>=400)

flagy2=1;

if(y3<=50)

flagx3=0;

else if(y3>=300)

flagx3=1;

if(x3<=10)

flagy3=0;

else if(x3>=400)

flagy3=1;

if(y4<=50)

flagx4=0;

else if(y4>=300)

flagx4=1;

if(x4<=10)

flagy4=0;

else if(x4>=400)

flagy4=1;

        Thread.sleep(10);

    }catch(InterruptedException e){}

}

}

public void paint(Graphics g)

{
```



```
g.drawRect(10,50,410,270);
```

```
    g.setColor(Color.blue);
```

```
    g.fillOval(x1,y1,20,20);
```

```
if(flagx1==1)
```

```
    y1-=2;
```

```
else if(flagx1==0)
```

```
    y1+=2;
```

```
if(flagy1==0)
```

```
    x1+=4;
```

```
else if(flagy1==1)
```

```
    x1-=4;
```

```
    g.setColor(Color.red);
```

```
    g.fillOval(x2,y2,20,20);
```

```
if(flagx2==1)
```

```
    y2-=4;
```

```
else if(flagx2==0)
```

```
    y2+=4;
```

```
if(flagy2==0)
```

```
    x2+=3;
```

```
else if(flagy2==1)
```

```
    x2-=3;
```

```
    g.setColor(Color.yellow);
```

```
    g.fillOval(x3,y3,20,20);
```

```
if(flagx3==1)

y3-=6;

else if(flagx3==0)

y3+=6;

if(flagy3==0)

x3+=2;

else if(flagy3==1)

x3-=2;

g.setColor(Color.magenta);

    g.fillOval(x4,y4,20,20);

if(flagx4==1)

y4-=5;

else if(flagx4==0)

y4+=5;

if(flagy4==0)

x4+=1;

else if(flagy4==1)

x4-=1;

}

}
```

-

slip no 5 Q.1 : Write a JSP program to calculate sum of first and last digit of a given number. Display sum in Red Color with font size 18.

Note : save file with name extension .html

```

<html>
<body>
<form method=get action="Number.jsp">
Enter Any Number : <Input type=text name=num><br><br>
<input type=submit value=Calculate>
</form>
</body>
</html>

```

Number.jsp

```

<html>
<body>
<%! int n,rem,r; %>
<% n=Integer.parseInt(request.getParameter("num"));
    if(n<10)
    {
        out.println("Sum of first and last digit is  ");
%><font size=18 color=red><%= n %></font>
<%
    }
    else
    {
        rem=n%10;
        do{
            r=n%10;
            n=n/10;
        }while(n>0);
        n=rem+r;
        out.println("Sum of first and last digit is  ");
%><font size=18 color=red><%= n %></font>
<%
    }
%>
</body>
</html>

```

slip 5 Q.2 : Write a java program in multithreading using applet for Traffic signal.

```

import java.applet.Applet;
import java.awt.*;
public class slip5B extends Applet implements Runnable {
    int i,j;

```

```

Thread t;
int timeOfSignal,signalColor;

public void init() {
    timeOfSignal=90;signalColor=0;
    t=new Thread(this);
    t.start();
}
public void run() {
    try{
        if (signalColor==0){
            for (j=0;j<90;j++){
                timeOfSignal=timeOfSignal-1;
                Thread.sleep(500);
                repaint();
            }
            signalColor=1;
            timeOfSignal=15;
        }
        if(signalColor==1){
            for (j=0;j<15;j++){
                timeOfSignal=timeOfSignal-1;
                Thread.sleep(500);
                repaint();
            }
            signalColor=2;
            timeOfSignal=90;
        }
        if(signalColor==2){
            for (j=0;j<90;j++){
                timeOfSignal=timeOfSignal-1;
                Thread.sleep(500);
                repaint();
            }
            signalColor=0;
            timeOfSignal=90;
        }
        if (i==0) {
            run();
        }
    }
    catch(Exception e) {
    }
}

```

```

public void paint(Graphics g) {
    g.drawOval(100,100,100,100);
    g.drawOval(100,225,100,100);
    g.drawOval(100,350,100,100);
    g.setFont(new Font("",Font.PLAIN,24));
    g.drawString(String.valueOf(timeOfSignal),400,150);

    if(signalColor==0&&!(j==90)){
        g.setColor(Color.red);
        g.fillOval(100,100,100,100);
        g.drawOval(100,100,100,100);
        g.drawString("stop",400,430);
    }
    if(signalColor==1&&!(j==15)) {
        g.setColor(Color.yellow);
        g.fillOval(100, 225, 100, 100);
        g.drawOval(100, 225, 100, 100);
        g.drawString("slow", 400, 430);
    }
    if (signalColor==2&&!(j==90)) {
        g.setColor(Color.green);
        g.fillOval(100,350,100,100);
        g.drawOval(100,350,100,100);
        g.drawString("go",400,430);
    }
}
}
/*<applet code= "Signal.class" height="600" width="600"></applet>*/

```

slip no 6 Q.1 : Write a java program to blink image on the Frame continuously

```

import java.awt.Color;
import java.awt.Graphics;
import java.awt.Image;
import java.io.File;

import javax.imageio.ImageIO;
import javax.swing.ImageIcon;
import javax.swing.JFrame;

public class slip6A extends JFrame {
    public static final long serialVersionUID = 1L;

    Graphics dbg;

```

```

Image dbImage;
static Image block;
static Block block1 = new Block();
static Image player1;
static Player player = new Player(193, 143);

public Game() {
    Image playerIcon = new ImageIcon("res/play.png").getImage();
    setSize(500, 400);
    setTitle("Game");
    setDefaultCloseOperation(EXIT_ON_CLOSE);
    setIconImage(playerIcon);
    setLocationRelativeTo(null);
    setVisible(true);
    addKeyListener(new InputHandler());
    setBackground(Color.BLACK);
    setResizable(false);
}

public static void main(String[] args) {
    new Game();
    Thread p = new Thread(player);
    p.start();
}

@SuppressWarnings("static-access")
public void paint(Graphics g) {
    try {
        dbImage = ImageIO.read(new File("res/img.png"));
    } catch (Exception e) {
        e.printStackTrace();
    }
    try {
        player1 = ImageIO.read(new File("res/img.png"));
    } catch (Exception e) {
        e.printStackTrace();
    }
    try {
        block = ImageIO.read(new File("res/img.png"));
    } catch (Exception e) {
        e.printStackTrace();
    }
    dbg = dbImage.getGraphics();
    draw(dbg);
    g.drawImage(dbImage, 0, 0, this);
}

```

```

        g.drawImage(player1, player.x, player.y, this);
        g.drawImage(block, block1.x, block1.y, this);
    }

    public void draw(Graphics g) {
        repaint();
    }
}

```

slip no 7 Q.2 : Write a Multithreading program in java to display the number's between 1 to 100 continuously in a TextField by clicking on button. (use Runnable Interface).

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
public class slip7B extends JFrame implements ActionListener
{
    Container cc;
    JButton b1,b2;
    JTextField t1;
    slip7B()
    {
        setVisible(true);
        setSize(1024,768);
        cc=getContentPane();
        setLayout(null);
        t1=new JTextField(500);
        cc.add(t1);
        t1.setBounds(10,10,1000,30);
        b1=new JButton("start");
        cc.add(b1);
        b1.setBounds(20,50,100,40);
        b1.addActionListener(this);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
    }
    public void actionPerformed(ActionEvent e)
    {
        if(e.getSource()==b1)
        {
            new Mythread();
        }
    }
}

```

```

}
class Mythread extends Thread
{
    Mythread()
    {
        start();
    }
    public void run()
    {
        for(int i=1;i<=100;i++)
        {
            try {
                Thread.sleep(1000);
            }
            catch (InterruptedException e) {
            }
            t1.setText(t1.getText()+""+i+"\n");
        }
    }
}
//System.out.println()
}
}
public static void main(String arg[])
{
    new slip7B().show();
}
}

```

slip no 8 Q.2 : Write a java program in multithreading using applet for Digital watch.

```

import java.applet.Applet;
import java.awt.*;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
public class slip8B extends Applet implements Runnable
{

    Thread t1 = null;
    int hours = 0, minutes = 0, seconds = 0;
    String time = "";
    public void init() {
        setBackground( Color.green);
    }
}

```



```

public void start() {
    t1 = new Thread( this );
    t1.start();
}

public void run() {
    try {
        while (true) {
            Calendar cal = Calendar.getInstance();
            hours = cal.get( Calendar.HOUR_OF_DAY );
            if ( hours > 12 ) hours -= 12;
            minutes = cal.get( Calendar.MINUTE );
            seconds = cal.get( Calendar.SECOND );
            SimpleDateFormat formatter = new SimpleDateFormat("hh:mm:ss");
            Date d = cal.getTime();
            time = formatter.format( d );
            repaint();
            Thread.sleep( 1000 );
        }
    }
    catch (Exception ignored) {}
}

public void paint( Graphics g ) {
    g.setColor( Color.blue );
    g.setFont(new Font("",Font.PLAIN,100));
    g.drawString( time, 100, 150 );
}
}
/*<applet code= "slip8B.class" height="300" width="600"></applet>*/

```

slip no 9 Q.1 : Write a Java Program to create a Emp (ENo, EName, Sal) table and insert record into it. (Use PreparedStatement Interface)

```

import java.io.*;
import java.sql.*;
public class slip9A
{
    static Connection cn;
    static Statement st;
    public static void main(String args[])
    {
        try
        {
            int tno,sal;

```

```

String tname,desg;
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
cn=DriverManager.getConnection("jdbc:odbc:dsn","","");
st=cn.createStatement();
String str="create table employee(eno number,ename varchar(20),sal
number,Desg varchar(20))";
st.executeUpdate(str);
System.out.println("Table Created");
System.out.println("Enter eno");
tno=Integer.parseInt(br.readLine());
System.out.println("Enter ename");
tname=br.readLine();
System.out.println("Enter Sal");
sal=Integer.parseInt(br.readLine());
System.out.println("Enter Desg");
desg=br.readLine();
st.executeUpdate("insert into employee
values("+eno+", "+ename+", "+sal+", "+desg+"");
System.out.println("Record added successfully");
cn.close();
}catch(Exception e)
{
    System.out.println(e);
}
}
}

```

slip no 11 Q.1 : Write a java program to display IPAddress and name of client machine

```

import java.net.*;
import java.io.*;
import java.util.*;
import java.net.InetAddress;
public class slip11A
{
    public static void main(String args[]) throws Exception{
        InetAddress my_localhost = InetAddress.getLocalHost();
        System.out.println("The IP Address of client is : " +
(my_localhost.getHostAddress()).trim());
        String my_system_address = "";
        try{
            URL my_url = new URL("http://bot.whatismyipaddress.com");
            BufferedReader my_br = new BufferedReader(new

```

```

        InputStreamReader(my_url.openStream()));
        my_system_address = my_br.readLine().trim();
    }
    catch (Exception e){
        my_system_address = "Cannot Execute Properly";
    }
}
}

```

slip no 13 Q.1 : Write a java program to display name of currently executing Thread in multithreading.

```

class slip13A extends Thread{
    @Override
    public void run() {
        System.out.println("Current thread is "+getName());
    }
    public static void main(String[] args) {
        slip13A thread1 = new slip13A();
        thread1.setName("first thread");
        slip13A thread2 = new slip13A();
        thread2.setName("second thread");
        slip13A thread3 = new slip13A();
        thread3.setName("third thread");
        slip13A thread4 = new slip13A();
        thread4.setName("fourth thread");
        slip13A thread5 = new slip13A();
        thread5.setName("fifth thread");

        thread1.run();
        thread2.run();
        thread3.run();
        thread4.run();
        thread5.run();
    }
}

```

slip no 14 Q. : Write a Java program to display given extension files from a specific directory on server machine.

```

import java.io.*;
public class slip14B

```

```

{
    public static void main(String[] args) throws IOException {
        String userFileExtension = "";

        BufferedReader bufferedInputStream = new BufferedReader(
            new InputStreamReader(System.in));
        System.out.println("GIVE A FILE LOCATION TO FIND SPECIFIC TYPE OF FILE");
        String userFileLocation = bufferedInputStream.readLine();
        File fileLocation = new File(userFileLocation);
        if (fileLocation.exists()) {
            System.out.println("\nGIVE A FILE EXTENSION TO FIND IN GIVEN DIRECTORY");
            userFileExtension = bufferedInputStream.readLine();
            String finalUserFileExtension = userFileExtension;

            String[] filenamesInDIR = fileLocation.list(new FilenameFilter() {
                @Override
                public boolean accept(File dir, String name) {
                    if (name.toLowerCase().endsWith("." + finalUserFileExtension)) {
                        return true;
                    } else {
                        return false;
                    }
                }
            });
            assert filenamesInDIR != null;
            if(filenamesInDIR.length!=0){
                for (String fileName:filenamesInDIR){
                    System.out.println(fileName);
                }
            }else {
                System.out.println("THERE IS NO FILE OF THAT EXTENSION");
            }
        }else {
            System.out.println("THIS FILE NOT EXIST IF YOU WANT TO RECHECK THEN TYPE 'Y' OR 'N'.");
            String recheckOption = bufferedInputStream.readLine();
            if("Y".equals(recheckOption.toUpperCase())){
                main(args);
            }
        }
    }
}

```

slip no 15 Q.1 : Write a java program to display each alphabet after 2 seconds between 'a' to 'z'.

```
import java.io.*;
class slip15A
{
    public static void main(String[] args) {

        char c;

        for(c = 'A'; c <= 'Z'; ++c)
            System.out.print(c + " ");
    }
}
```

slip no 17 Q. 1 : Write a java program to accept a String from user and display each vowel from a String after 3 seconds.

```
import java.util.HashSet;
import java.util.Scanner;
import java.util.Set;

public class slip17A
{
    public static void main(String args[])
    {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter an String : ");
        String str = scanner.next();

        Set<Character> set=new HashSet<Character>();
        for (int i = 0; i < str.length(); i++) {

            char c=str.charAt(i);
            if(isVowel(c))
            {
                set.add(c);
            }
        }

        System.out.println("Vowels are:");
    }
}
```

```

    for (Character c:set) {
        System.out.print(" "+c);
    }

    scanner.close();
}

public static boolean isVowel(char character)
{
    if(character=='a' || character=='A' || character=='e' || character=='E' ||
        character=='i' || character=='I' || character=='o' || character=='O' ||
        character=='u' || character=='U'){
        return true;
    }else{
        return false;
    }
}
}

```

slip no 18 Q.1 : Write a java program to calculate factorial of a number. (Use sleep () method).

```

import java.util.Scanner;

public class slip18A
{
    public static void main(String[] args) {

        int fact = 1;
        int i = 1;

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter a number whose factorial is to be found: ");
        int num = sc.nextInt();

        //counting the factorial using while loop
    }
}

```

```

while( i <= num ){
    fact = fact * i;
    i++; //increment i by 1
}

System.out.println("\nFactorial of " + num + " is: " + fact);
}
}

```

slip no 20 Q.2 : Write a java program in multithreading using applet for drawing temple.

```

import java.applet.Applet;
import java.awt.*;
public class slip20B extends Applet
{
    public void paint(Graphics g)
    {
        g.drawRect(100,150,90,120);
        g.drawRect(130,230,20,40);
        g.drawLine(150,100,100,150);
        g.drawLine(150,100,190,150);
        g.drawLine(150,50,150,100);
        g.drawRect(150,50,20,20);
    }
}

```

slip no 21 Q.1 : Write a java program to display name and priority of a Thread.

```

class slip21A extends Thread
{
    @Override
    public void run() {
        System.out.println("Current thread Name "+this.getName());
        System.out.println("Current thread priority "+this.getPriority());
        System.out.println();
    }
    public static void main(String[] args) {
        slip21A thread1 = new slip21A();
        slip21A thread2 = new slip21A();
    }
}

```

```

thread2.setName("groot");
thread2.setPriority(6);
slip21A thread3 = new slip21A();
thread3.setName("Amar");
thread3.setPriority(2);

thread1.run();
thread2.run();
thread3.run();
}
}

```

slip no 22 Q.1 : Write a java program to display Date and Time of Server machine on client machine

```

import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;

public class slip22A
{
    public static void main(String[] args) {
        LocalDateTime current = LocalDateTime.now();

        DateTimeFormatter formatter = DateTimeFormatter.ofPattern("yyyy-MM-dd
HH:mm:ss.SSS");
        String formatted = current.format(formatter);

        System.out.println("Current Date and Time is: " + formatted);
    }
}

```

slip no 23 Q.2 : Write a SERVLET application to accept username and password, search them into database, if found then display appropriate message on the browser otherwise display error message.

note : save file name with extension .html

```

<html>
<body>
<form method=post action="http://localhost:4141/Program/servlet/UserPass">

```



```
User Name :<input type=text name=user><br><br>
Password :<input type=text name=pass><br><br>
<input type=submit value="Login">
</form>
</body>
</html>
```

UserPass.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class UserPass extends HttpServlet
{
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws IOException, ServletException
    {
        PrintWriter out = response.getWriter();
        try{
            String us=request.getParameter("user");
            String pa=request.getParameter("pass");
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            Connection cn=DriverManager.getConnection("jdbc:odbc:dsn2","","");
            Statement st=cn.createStatement();
            ResultSet rs=st.executeQuery("select * from UserPass");
            while(rs.next())
            {
                if(us.equals(rs.getString("user"))&&pa.equals(rs.getString("pass")))
                    out.println("Valid user");
                else
                    out.println("Invalid user");
            }
        }catch(Exception e)
        {
            out.println(e);
        }
    }

    public void doPost(HttpServletRequest request, HttpServletResponse response)
        throws IOException, ServletException
    {
        doGet(request, response);
    }
}
```

Web.xml file(servlet entry)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app>
<servlet>
    <servlet-name>UserPass</servlet-name>
    <servlet-class>UserPass</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>UserPass</servlet-name>
    <url-pattern>/servlet/UserPass</url-pattern>
</servlet-mapping>
</web-app>
```

slip no 24 Q.1 : Create a JSP page to accept a number from a user and display it in words: Example: 123 – One Two Three. The output should be in red color

note : save file name with extension html

```
<html>
<body>
<form method=get action="NumberWord.jsp">
Enter Any Number : <input type=text name=num><br><br>
<input type=submit value="Display">
</form>
<body>
</html>
```

NumberWord.jsp

```
<html>
<body>
<font color=red>
<%! int i,n;
    String s1;
%>
<% s1=request.getParameter("num");
    n=s1.length();
    i=0;
    do
    {
        char ch=s1.charAt(i);
        switch(ch)
        {
```

```

        case '0': out.println("Zero ");break;
        case '1': out.println("One ");break;
        case '2': out.println("Two ");break;
        case '3': out.println("Three ");break;
        case '4': out.println("Four ");break;
        case '5': out.println("Five ");break;
        case '6': out.println("Six ");break;
        case '7': out.println("Seven ");break;
        case '8': out.println("Eight ");break;
        case '9': out.println("Nine ");break;
    }
    i++;
}while(i<n);
%>
</font>
</body>
</html>

```

slip no 26 Q.2 : Write a SERVLET program to Design an HTML page containing 4 option buttons (Painting, Drawing, Singing and swimming) and 2 buttons reset and submit. When the user clicks submit, the server responds by adding cookie containing the selected hobby and sends the HTML page to the client. Program should not allow duplicate cookies to be written.

```

import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class slip26B extends HttpServlet
{
    public void doPost(HttpServletRequest req,HttpServletResponse resp)throws
ServletException,IOException
    {
        resp.setContentType("text/html");
        PrintWriter out = resp.getWriter();
        String s=req.getParameter("hobby");

        Cookie c=new Cookie("hobby",s);
        out.println("You have selected:"+c.getValue());

    }
    public void doGet(HttpServletRequest req,HttpServletResponse resp)throws
ServletException,IOException

```

```

{
    resp.setContentType("text/html");
    PrintWriter out = resp.getWriter();
    String s=req.getParameter("hobby");

    Cookie c=new Cookie("hobby",null);
    out.println("Reset....");

}

```

slip no 28 Q.1 : Write a java program for the implementation of synchronization

```

import system.io.*;
class slip28A
{
    synchronized void printTable(int n)
    {
        for(int i=1;i<=5;i++){
            System.out.println(n*i);
            try{
                Thread.sleep(400);
            }catch(Exception e){System.out.println(e);}
        }
    }
}

class MyThread1 extends Thread{
    slip28A t;
    MyThread1(slip28A t){
        this.t=t;
    }
    public void run(){
        t.printTable(5);
    }
}

class MyThread2 extends Thread
{
    slip28A t;
    MyThread2(slip28A t){
        this.t=t;
    }
}

```

```

public void run(){
t.printTable(100);
}
}

public class slip28AA
{
public static void main(String args[]){
slip28A obj = new slip28A();
MyThread1 t1=new MyThread1(obj);
MyThread2 t2=new MyThread2(obj);
t1.start();
t2.start();
}
}

```

slip no 29 Q.1 : Write a java program using multithreading for the following: 1. Display all the odd numbers between 1 to n. 2. Display all the prime numbers between 1 to n

```

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

class slip29A extends Thread
{
String name;
Thread t;
static BufferedReader br;
slip29A(String fname){
name = fname;
t=new Thread(this.name);
t.start();
}
@Override
public void run() {
if (name.equals("odd"))
{
try{
System.out.println("Enter a number to find odd number ");
int n = Integer.parseInt(br.readLine());
System.out.print("ODD NUMBERS ");
for (int i=1; i<=n; i+=2)

```

```

        {
            System.out.print(i+" ");
        }
    }catch(Exception e){
        System.out.println(e);
    }
}
else if (name.equals("prime"))
{
    try{
        System.out.println("Enter a number to find prime number ");
        int n = Integer.parseInt(br.readLine());
        if(n==0||n==1){
            System.out.println(n+" is not prime number");
        }
        else{
            for (int num = 2; num <= n; num++) {
                boolean isPrime = true;
                for (int i=2; i <= num/2; i++){
                    if ( num % i == 0) {
                        isPrime = false;
                        break;
                    }
                }
                if ( isPrime == true ){
                    System.out.print(num + " ");
                }
            }
        }
    }catch(Exception e){
        System.out.println(e);
    }
}
}

public static void main(String[] args) throws IOException {
    br = new BufferedReader(new InputStreamReader(System.in));
    slip29A odd = new slip29A("odd");
    slip29A prime = new slip29A("prime");

    odd.run();
    System.out.println();
    prime.run();
    br.close();
}
}

```

slip 30 Q.2 : Create Table Employee(Eno, Ename, Designation, Salary). Create Android Application for performing the following operation on the table. (Using SQLite Database) i) Insert New Employee Details. [25 M] ii) Display all the Employee detail

```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
public class slip30B extends Applet implements ActionListener
{
    int x, y, z;
    int t1,t2;
    Button b1, b2;
    String msg=" ";
    void slep()
    {
        try
        {
            Thread.sleep(100);
        }
        catch(Exception ex)
        {
        }
    }
    public void init()
    {
        t1=0;
        t2=1;
        x=20;
        y=60;
        setLayout(new FlowLayout(FlowLayout.CENTER));
        Label l=new Label("webeduclick");
        b1=new Button("Forward");
        add(b1);
        b2=new Button("Stop");
        add(b2);
        b1.addActionListener(this);
        b2.addActionListener(this);
    }
    public void start()
    {
    }
    public void actionPerformed(ActionEvent e)
```

```

{
String s=e.getActionCommand();
if(s.equals("Forward"))
{
msg="Forward";
repaint();
}
else if(s.equals("Stop"))
{
msg=" ";
repaint();
}
}
public void paint(Graphics g)
{
setBackground(Color.black);
z=getWidth();
Color c1=new Color(20,160,200);
Color c2=new Color(200,60,200);
g.setColor(c1);
g.drawLine(0,y+75,z,y+75);
g.setColor(Color.red);
g.fillRect(x,y+20,100,40,5,5);
g.fillArc(x+90,y+20,20,40,270,180);
g.setColor(Color.BLUE);
g.fillRect(x+10,y,70,25,10,10);
g.setColor(Color.white);
g.fillRect(x+20,y+5,20,25);
g.fillRect(x+50,y+5,20,25);
g.setColor(Color.black);
g.fillRect(x+55,y+10,10,20,10,10);
g.fillOval(x+10,y+50,25,25);
g.fillOval(x+60,y+50,25,25);
g.setColor(Color.white);
g.fillOval(x+15,y+55,10,10);
g.fillOval(x+65,y+55,10,10);
x=x+10;
slep();
if(msg.equals("Forward"))
{
if(x+120<z)
{
x=x+1;
showStatus("Press Forward for Starting Car");
repaint(); } } }

```


