Name: Aarna Bafna
Batch: S11
Roll No.: 11
Subject: JAVA

Java Assignments – FINAL

1. Salary Calculation

Program

```
import java.util.Scanner; public class Assignment2
{
public static void calcProcess(int sal,String name, int no)
{
double DA, HRA, PF, gross, net; int CCA=240, PT=100;

HRA=(0.3*sal);
DA=(0.7*sal);
PF=(0.1*sal);
gross = (sal + HRA + DA + CCA + PF + PT);
net=(gross-(CCA+PT+PF)); System.out.println();
System.out.println("The DA is : " + DA); System.out.println("The
HRA is : " + HRA); System.out.println("The CCA is : " + CCA);
System.out.println("The PF is : " + PF); System.out.println("The PT
is : " + PT);
System.out.println("The gross and net salary of "
+ name + " with employee number : " + no + " is : ");
System.out.println("Gross = " + gross); System.out.println("Net = "
+ net);
}

public static void main (String args[])
{
Scanner scanner = new Scanner(System.in); boolean quit = false;
while(!quit)
{
int n; System.out.println();
System.out.println("Please select an option from below :");
System.out.println("1. Enter employee details\n2.
Quit");
System.out.println("Enter the option selected : ");
n=scanner.nextInt();
System.out.println();

switch(n)
{
case 1 : int empno, basic; String empname;
Scanner sc = new Scanner(System.in); System.out.println("Enter
employee name : ");
empname=sc.nextLine();
System.out.println("Enter employee number :”);
        empno=sc.nextInt();
        System.out.println("Enter the basic salary of " + empname +
        " : ");
        basic=sc.nextInt(); calcProcess(basic,empname,empno);
        break;
```

Output

```
Please select an option from below :
1. Enter employee details
2. Quit
Enter the option selected :
1
Enter employee name :
Aarna
Enter employee number :
11
Enter the basic salary of Aarna :
100000
The DA is : 70000.0
The HRA is : 30000.0
The CCA is : 240
The PF is : 10000.0
The PT is : 100
The gross and net salary of Aarna with
employee number : 11 is :
Gross = 210340.0
Net = 200000.0
Please select an option from below :
1. Enter employee details
2. QuitEnter the option selected: 2
```

```
        case 2 : quit=true; System.out.println("Program Quit
        Successfully.");
        break;

        default : System.out.println("Invalid choice.");
        }
        }
        }
}
```

2. Basic Programs Using Switch Case

Program

```
import java.util.Scanner;
public class Assignment
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      System.out.println("1. Even or Odd");
      System.out.println("2. Factorial of a number");
      System.out.println("3. Prime Number or not");
      System.out.println("4. Swapping numbers");
      System.out.println("5. Reversing the number");
      System.out.println("6. Armstrong Number or not");
      System.out.println("7. Palindrome");
      System.out.println("8. Fibonacci series");

      System.out.println("Enter option number : ");
      int number = sc.nextInt();

      switch(number)
      {
         case 1 : System.out.println("Even or Odd");
                  EvenOrOdd();
         break;

         case 2 : System.out.println("Factorial of a number ");
                  factorial();
         break;

         case 3 :  System.out.println("Prime Number or not ");
                   primeNo();
         break;

         case 4 : System.out.println("Swapping numbers ");
                  swapNos();
         break;

         case 5 : System.out.println("Reversing the number ");
                  reverseNo();
         break;

         case 6 : System.out.println("Armstrong Number or not ");
                  armstrongNo();
         break;

         case 7 : System.out.println("Palindrome or not");
                  palindromeNo();
         break;

         case 8 : System.out.println("Fibonacci series");
                  fibonacciSeries();
         break;

         default : System.out.println("Invalid choice");
      }
   }
//Even Odd
```

Output

```
Command Prompt

C:\Javapractice>javac SwitchCase.java

C:\Javapractice>java SwitchCase
1. Even or Odd
2. Factorial of a number
3. Prime Number or not
4. Swapping numbers
5. Reversing the number
6. Armstrong Number or not
7. Palindrome
8. Fibonacci series
Enter option number :
1
Even or Odd

Enter a positive number:
23
Number 23 is odd
```

```
Command Prompt

C:\Javapractice>javac SwitchCase.java

C:\Javapractice>java SwitchCase
1. Even or Odd
2. Factorial of a number
3. Prime Number or not
4. Swapping numbers
5. Reversing the number
6. Armstrong Number or not
7. Palindrome
8. Fibonacci series
Enter option number :
2
Factorial of a number

Enter a positive number:
3
Factorial of number 3 is 6
```

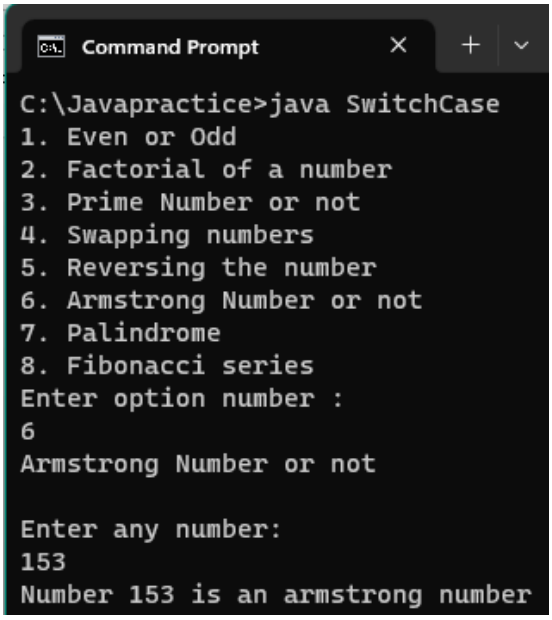
```
    public static void EvenOrOdd()
    {
        System.out.println("\nEnter a positive number: ");
        Scanner sc = new Scanner(System.in);
        int a = sc.nextInt();
        if(a % 2 == 0)
        {
            System.out.println("Number " + a + " is even");
        }
        else
        {
            System.out.println("Number " + a + " is odd");
        }
    }

//Factorial
    public static void factorial()
    {
        System.out.println("\nEnter a positive number: ");
        int fact = 1;
        Scanner sc = new Scanner(System.in);
        int a = sc.nextInt();
        for(int i = 1; i<= a; i++)
        {
            fact = fact*i;
        }
       System.out.println("Factorial of number " + a + " is " + fact);
    }

//Prime Number
    public static void primeNo()
    {
        System.out.println("\nEnter a positive number: ");
        int count = 0;
        Scanner sc = new Scanner(System.in);
        int a = sc.nextInt();
        for(int i = 1; i<= a; i++)
            {
               if(a % i == 0)
               {
                  count++;
               }
            }
           if(count==2)
           {
              System.out.println("Number " + a + " is a prime
number");
           }
           else
           {
              System.out.println("Number " + a + " is not a
primenumber");
           }

    }

//Swapping numbers
    public static void swapNos()
    {
        System.out.println("\nEnter two positive number");
```

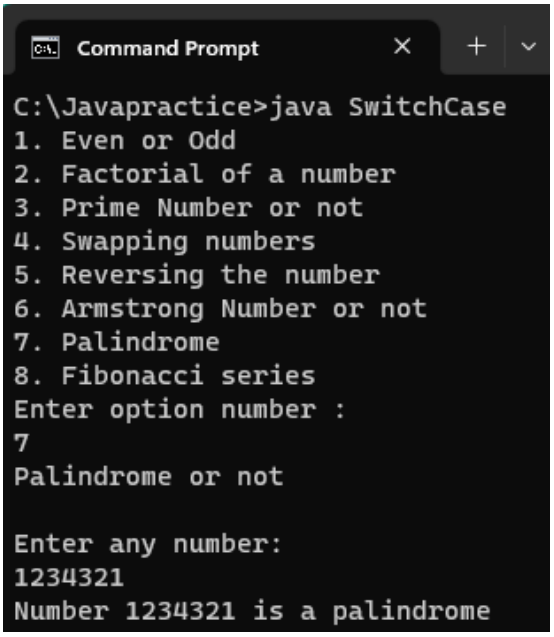
```
Command Prompt

C:\Javapractice>javac SwitchCase.java

C:\Javapractice>java SwitchCase
1. Even or Odd
2. Factorial of a number
3. Prime Number or not
4. Swapping numbers
5. Reversing the number
6. Armstrong Number or not
7. Palindrome
8. Fibonacci series
Enter option number :
3
Prime Number or not

Enter a positive number:
3
Number 3 is a prime number
```

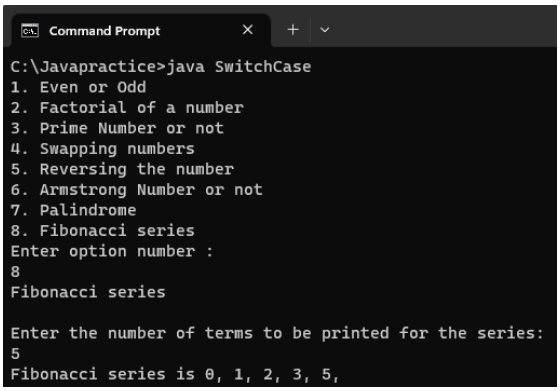
```
Command Prompt

C:\Javapractice>javac SwitchCase.java

C:\Javapractice>java SwitchCase
1. Even or Odd
2. Factorial of a number
3. Prime Number or not
4. Swapping numbers
5. Reversing the number
6. Armstrong Number or not
7. Palindrome
8. Fibonacci series
Enter option number :
4
Swapping numbers

Enter two positive number
12
45
Numbers before swapping are 12 and 45
Numbers after swapping are 45 and 12
```

```
Command Prompt

C:\Javapractice>javac SwitchCase.java

C:\Javapractice>java SwitchCase
1. Even or Odd
2. Factorial of a number
3. Prime Number or not
4. Swapping numbers
5. Reversing the number
6. Armstrong Number or not
7. Palindrome
8. Fibonacci series
Enter option number :
5
Reversing the number

Enter any number:
1234
Number 1234 after reversing becomes 4321
```

```
        int temp;
        Scanner sc = new Scanner(System.in);
        int a = sc.nextInt();
        int b = sc.nextInt();
        System.out.println("Numbers before swapping are " + a + "
and " + b);
        temp = a;
        a = b;
        b = temp;
        System.out.println("Numbers after swapping are " + a + " and
"+ b);
    }

//Reversing number
    public static void reverseNo()
    {
        System.out.println("\nEnter any number: ");
        int rem = 0;
        int reverse = 0;
        Scanner sc = new Scanner(System.in);
        int a = sc.nextInt();
        int m = a;
        while(m > 0)
        {
            rem = m % 10;
            reverse = reverse*10 + rem;
            m = m/10;
        }
        System.out.println("Number " + a + " after reversing becomes
"+ reverse);
    }

//Armstrong number
    public static void armstrongNo()
    {
        System.out.println("\nEnter any number: ");
        int rem = 0;
        int sum = 0;
        Scanner sc = new Scanner(System.in);
        int a = sc.nextInt();
        int m = a;
        while(m > 0)
        {
            rem = m % 10;
            sum = sum + rem*rem*rem;
            m = m/10;
        }
        if(sum == a)
        {
            System.out.println("Number " + a + " is an armstrong
number");
        }
        else
        {
            System.out.println("Number " + a + " is not an armstrong
number");
        }
    }

//Palindrome
```







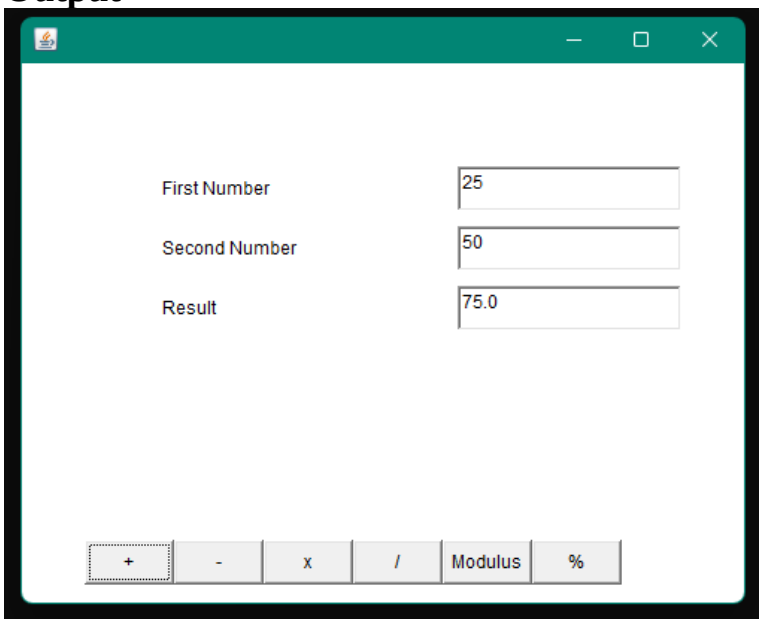
```
   public static void palindromeNo()
   {
      System.out.println("\nEnter any number: ");
      int rem = 0;
      int reverse = 0;
      Scanner sc = new Scanner(System.in);
      int a = sc.nextInt();
      int m = a;
      while(m > 0)
      {
         rem = m % 10;
         reverse = reverse*10 + rem;
         m = m/10;
      }
      if(a == reverse)
      {
         System.out.println("Number " + a + " is a palindrome");
      }
      else
      {
         System.out.println("Number " + a + " is not a palindrome");
      }
   }

//Fibonacci
   public static void fibonacciSeries()
   {
      int t1, t2;
      t1 = 0;
      t2 = 1;
      int nextTerm = t1 + t2;
      System.out.println("\nEnter the number of terms to be printed
for the series: ");
      Scanner sc = new Scanner(System.in);
      int a = sc.nextInt();
      System.out.print("Fibonacci series is " + t1 + ", " + t2 + ", ");
      for(int i = 3; i<=a ; i++)
      {
        t1 = t2;
        t2 = nextTerm;
        nextTerm = t1 + t2;
        System.out.printf("%d, ", nextTerm);
      }

   }
}
}
```

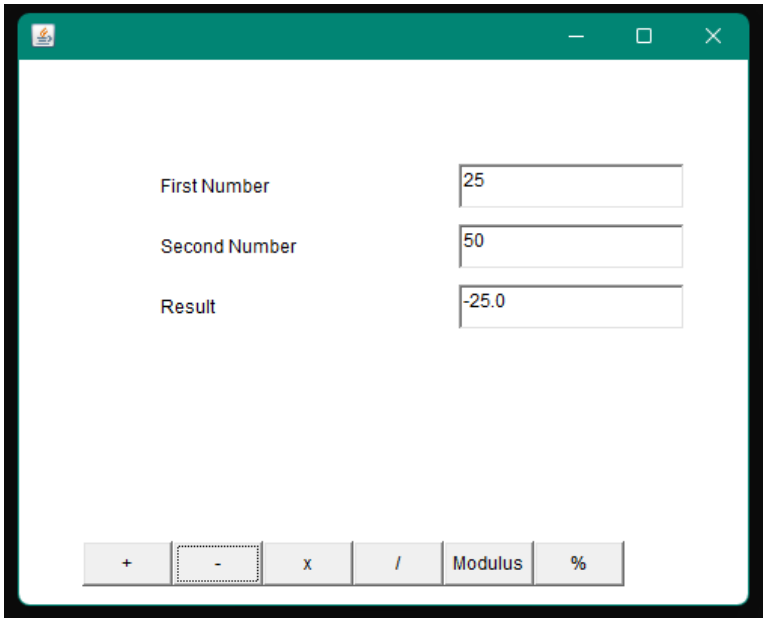
3. Calculator

Program

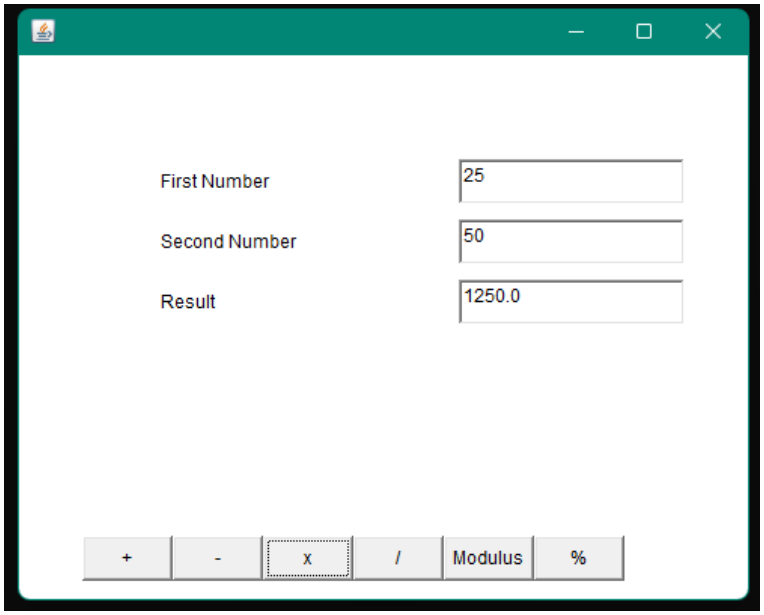
```
import java.awt.*;
import java.awt.event.*;
class calculator implements ActionListener
{
   Frame f=new Frame();
   Label l1=new Label("First Number");
   Label l2=new Label("Second Number");
   Label l3=new Label("Result");

   TextField t1=new TextField();
   TextField t2=new TextField();
   TextField t3=new TextField();

         Button sum=new Button("+");
   Button sub=new Button("-");
   Button mul=new Button("x");
   Button divi=new Button("/");
   Button percent=new Button("%");
   Button modul=new Button("Modulus");

   calculator()
   {
      l1.setBounds(100,100,150,30);
      l2.setBounds(100,140,150,30);
      l3.setBounds(100,180,150,30);

      t1.setBounds(300,100,150,30);
      t2.setBounds(300,140,150,30);
      t3.setBounds(300,180,150,30);

      sum.setBounds(50,350,60,30);
      sub.setBounds(110,350,60,30);
      mul.setBounds(170,350,60,30);
      divi.setBounds(230,350,60,30);
      modul.setBounds(290,350,60,30);
      percent.setBounds(350,350,60,30);

      f.add(l1);
      f.add(l2);
      f.add(l3);

      f.add(t1);
      f.add(t2);
      f.add(t3);

      f.add(sum);
      f.add(sub);
      f.add(mul);
      f.add(divi);
      f.add(modul);
      f.add(percent);

      sum.addActionListener(this);
      sub.addActionListener(this);
      mul.addActionListener(this);
      divi.addActionListener(this);
```

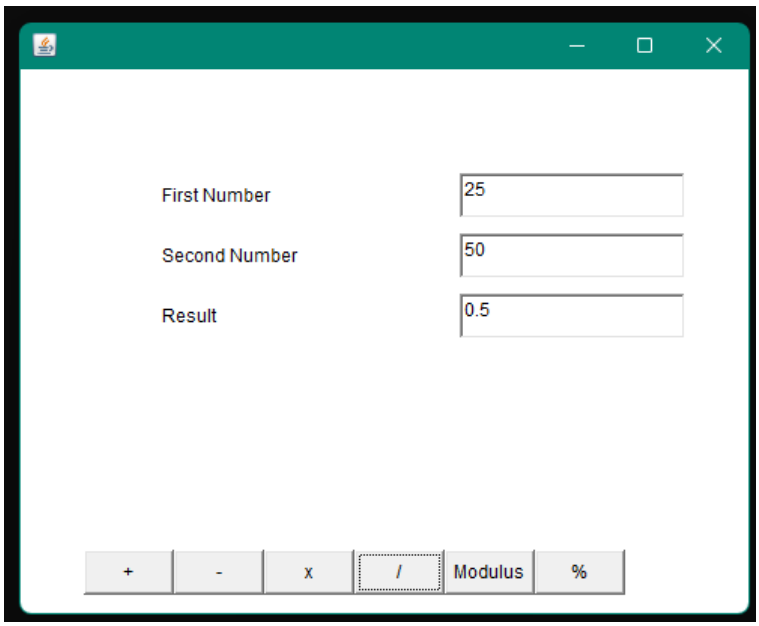
Output

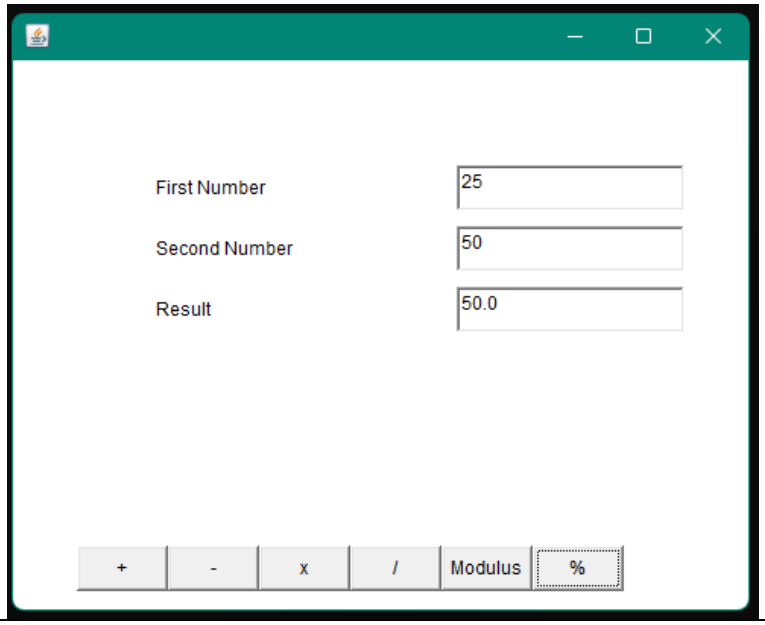




```
        modul.addActionListener(this);
        percent.addActionListener(this);

        f.setLayout(null);
        f.setVisible(true);
        f.setSize(500,400);
    }

    public void  actionPerformed(ActionEvent e)
    {
        float n1=Integer.parseInt(t1.getText());
        float n2=Integer.parseInt(t2.getText());

        if(e.getSource()==sum)
        {
            t3.setText(String.valueOf(n1+n2));
        }
        if(e.getSource()==sub)
        {
            t3.setText(String.valueOf(n1-n2));
        }
        if(e.getSource()==mul)
        {
            t3.setText(String.valueOf(n1*n2));
        }
        if(e.getSource()==divi)
        {
            t3.setText(String.valueOf(n1/n2));
        }
        if(e.getSource()==modul)
        {
            t3.setText(String.valueOf(n1%n2));
        }
        if(e.getSource()==percent)
        {
            t3.setText(String.valueOf((n1/n2)*100));
        }
    }

    public static void main(String...s)
    {
        new calculator();
    }
}
```







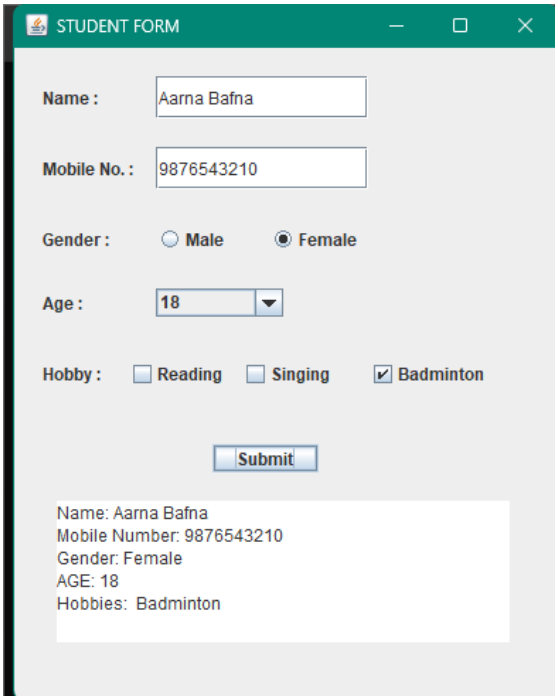
****

4. Student Form

Program

```
import javax.swing.*;
import java.awt.event.*;
 class StudentForm extends JFrame implements ActionListener{
   JLabel l1,l2,l3,l4,l5;
   JTextField tx1,tx2;
         JRadioButton rb1,rb2;
         JComboBox cb;
         JCheckBox cb1,cb2,cb3;
         JButton b;
         JTextArea area;

         StudentForm()
         {
                  JFrame f=new JFrame("STUDENT FORM");
            JLabel l1=new JLabel("Name :");
     l1.setBounds(20,20, 80,30);
                  tx1 = new JTextField();
     tx1.setBounds(100,20, 150,30);
                  f.add(l1);          f.add(tx1);


                  JLabel l2=new JLabel("Mobile No. :");
     l2.setBounds(20,70, 80,30);
                  tx2 = new JTextField();
     tx2.setBounds(100,70, 150,30);
           f.add(l2);      f.add(tx2);

                  JLabel l3=new JLabel("Gender :");
     l3.setBounds(20,120, 80,30);
                  rb1=new JRadioButton("Male");
     rb1.setBounds(100,120,60,30);
     rb2=new JRadioButton("Female");
     rb2.setBounds(180,120,100,30);

     ButtonGroup bg=new ButtonGroup();
     bg.add(rb1);bg.add(rb2);
     f.add(rb1);f.add(rb2);f.add(l3);

     JLabel l4=new JLabel("Age :");
     l4.setBounds(20,165, 80,30);

                  String age[]={"18","19","20","21","22"};
     cb=new JComboBox(age);
     cb.setBounds(100, 170,90,20);
     f.add(l4); f.add(cb);

           JLabel l5=new JLabel("Hobby :");
     l5.setBounds(20,215, 50,30);
                  f.add(l5);
                  cb1=new JCheckBox("Reading");
     cb1.setBounds(80,220,80,20);
     cb2=new JCheckBox("Singing");
     cb2.setBounds(160,220,80,20);
     cb3=new JCheckBox("Badminton");
     cb3.setBounds(250,220,100,20);
                  f.add(cb1);f.add(cb2);f.add(cb3);
```

Output



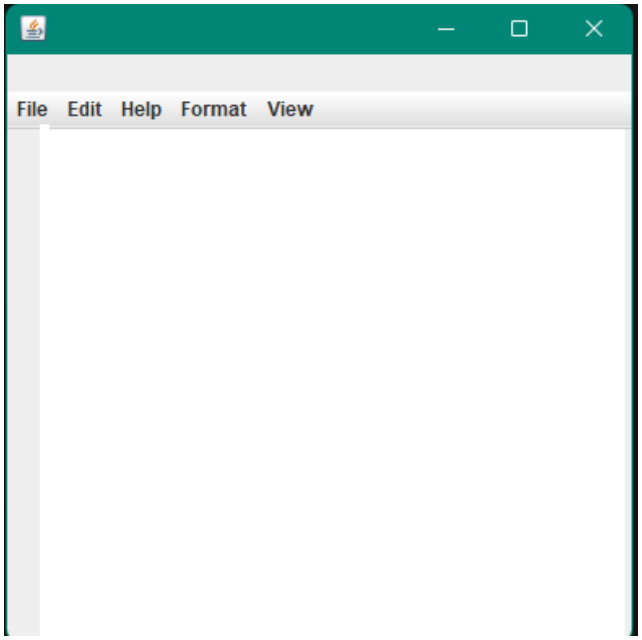
```
                JButton b=new JButton("Submit");
      b.setBounds(140,280,75,20);
            f.add(b);

            area=new JTextArea();
      area.setBounds(30,320, 320,100);
      f.add(area);
                //-------------------------------------------------------------
---
                b.addActionListener(this);
                setDefaultCloseOperation(EXIT_ON_CLOSE);

                f.setSize(400,500);
                f.setLayout(null);
      f.setVisible(true);
          }

          public void actionPerformed(ActionEvent e)
          {
                String name =  tx1.getText();
                String mobile =  tx2.getText();
                String gender =  rb1.isSelected() ?
"Male":rb2.isSelected()?"Female":"-";
                String age =
cb.getItemAt(cb.getSelectedIndex()).toString();
                String hobby="";
          if(cb1.isSelected())
          {
      hobby="Reading";
          }

          if(cb2.isSelected())
          {
      hobby=hobby+" "+"Singing";
          }

          if(cb3.isSelected())
          {
      hobby=hobby+" "+"Badminton";;
    }

          area.setText("Name: "+name+"\n"+"Mobile Number:
"+mobile+"\n"+"Gender: "+gender+"\n"+"AGE:
"+age+"\n"+"Hobbies: "+hobby);

}

          public static void main(String[] args)
          {
                new StudentForm();
          }
}
```

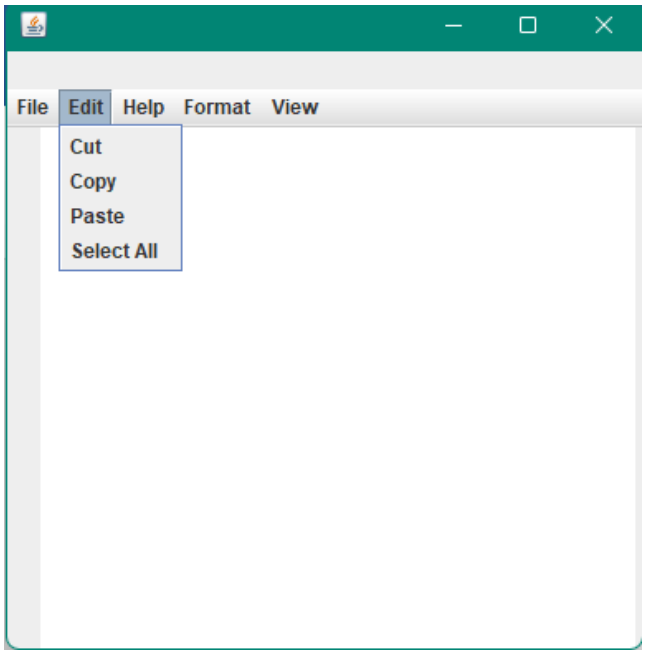
5. Menu Bar

Program

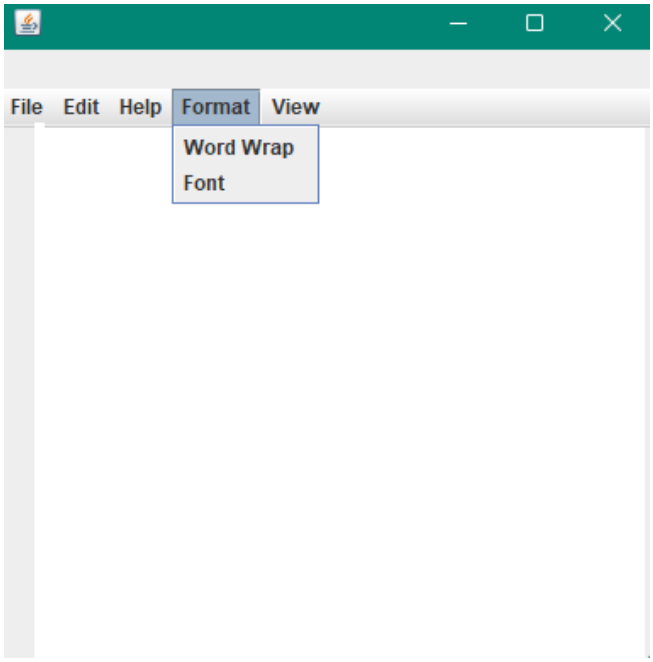
```
import javax.swing.*;
import java.awt.event.*;
public class WorkingNotepad implements ActionListener
{
	JFrame f;
	JMenuBar mb;
	JMenu file,edit,help,view,format;
	JMenuItem
cut,copy,paste,selectAll,N,Nw,O,S,Sa,Ps,P,E,ww,font,sb,Z;
	JTextArea ta;
	WorkingNotepad()
	{
		f=new JFrame();
		mb= new JMenuBar();
		f.setJMenuBar(mb);
		f.add(mb);
		file=new JMenu("File");
		edit=new JMenu("Edit");
		help=new JMenu("Help");
		format=new JMenu("Format");
		view=new JMenu ("View");
		mb.add(file);
		mb.add(edit);
		mb.add(help);
		mb.add(format);
		mb.add(view);

		{
			//FILE OPTIONS
		N=new JMenuItem("New");
		Nw=new JMenuItem("New Window");
		O=new JMenuItem("Open");
		S=new JMenuItem("Save");
		Sa=new JMenuItem("Save As");
		Ps=new JMenuItem("Page Setup");
		P=new JMenuItem("Print");
		E=new JMenuItem("Exit");
		file.add(N);
		file.add(Nw);
		file.add(O);
		file.add(S);
		file.add(Sa);
		file.add(Ps);
		file.add(P);
		file.add(E);
		}

		{
			//FORMAT OPTIONS
		ww = new JMenuItem("Word Wrap");
		font= new JMenuItem("Font");
		format.add(ww);
		format.add(font);
		}

		{
```

Output





```
                        //VIEW OPTIONS
                sb= new JMenuItem("Status Bar");
                Z = new JMenuItem("Zoom");
                view.add(Z);
                view.add(sb);
                }

                {
                        //EDIT OPTIONS
                cut=new JMenuItem("Cut");
                copy=new JMenuItem("Copy");
                paste=new JMenuItem("Paste");
                selectAll=new JMenuItem("Select All");
                edit.add(cut);
                edit.add(copy);
                edit.add(paste);
                edit.add(selectAll);

                cut.addActionListener(this);
                copy.addActionListener(this);
                paste.addActionListener(this);
                selectAll.addActionListener(this);
                }
                ta=new JTextArea();
                ta.setBounds(20,20,360,320);
                f.add(ta);

                f.setLayout(null);
                f.setSize(400,400);
                f.setVisible(true);
        }
        public void actionPerformed(ActionEvent e)
        {
                if(e.getSource()==cut)
                        ta.cut();
                if(e.getSource()==copy)
                        ta.copy();
                if(e.getSource()==paste)
                        ta.paste();
                if(e.getSource()==selectAll)
                        ta.selectAll();
        }
        public static void main(String[] args)
        {
                new WorkingNotepad();
        }
}
```



6. Method Overloading

Program

```
class Book
{
public static void main(String[] args)
{
Book book = new Book();
book.display("How to win friends and influence people",
"Dale Carnegie", 1936, 1, 550.0);
Reference_Book ref = new Reference_Book();
ref.display("Applied Physics Part II", "I.A. Shaikh", 2000, 2,
329.0);
Magazine mag = new Magazine();
mag.display("Vogue", "Anonymous", 2004, 5, 250.5);
}
protected String bookName;
protected String authorName;
protected int publishedYear;
protected int editionNo;
protected double cost;

public void display(String bookName, String authorName, int
publishedYear, int editionNo, double cost)
{
System.out.println("The name of book is " + bookName);
System.out.println("The author of book is " + authorName);
System.out.println("The year of publishing is " +
publishedYear);
System.out.println("The edition number is " + editionNo);
System.out.println("The cost of book is " + cost); }
}
class Reference_Book extends Book
{
public void display(String bookName, String authorName, int
publishedYear, int editionNo, double cost)
{
System.out.println();
System.out.println("This is a reference book");
System.out.println("The name of book is " + bookName);
System.out.println("The author of book is " + authorName);
System.out.println("The year of publishing is " +
publishedYear);
System.out.println("The edition number is " + editionNo);
System.out.println("The cost of book is " + cost);
}
}
class Magazine extends Book
{
public void display(String bookName, String authorName, int
publishedYear, int editionNo, double cost)
{
System.out.println();
System.out.println("This is a magazine");
```

Output

```
The name of book is How to win friends and influence people
The author of book is Dale Carnegie
The year of publishing is 1936
The edition number is 1
The cost of book is 550.0

This is a reference book
The name of book is Applied Physics Part II
The author of book is I.A. Shaikh
The year of publishing is 2000
The edition number is 2
The cost of book is 329.0

This is a magazine
The name of book is Vogue
The author of book is Anonymous
The year of publishing is 2004
The edition number is 5
The cost of book is 250.5
```

```
System.out.println("The name of book is " + bookName);
System.out.println("The author of book is " + authorName)
System.out.println("The year of publishing is " +
publishedYear);
System.out.println("The edition number is " + editionNo);
System.out.println("The cost of book is " + cost);
}
}
```

****

7. Constructor

Program

```
public class Operations
{
 private int x;
 private int y;
 private int z;

//Constructor with no parameters passed
public Operations()
{
   x=5;
   y=7;
   z=10;
   int sum=x+y+z;
   System.out.println("Sum of numbers " +x+ ", " +y+ ", " +z+
"in constructor 0 is " +sum);
}

//Constructor with one parameter passed
public Operations(int x)
{
   this.x=x;
   y=22;
   z=12;
   int sum=x+y+z;
   System.out.println("Sum of numbers " +x+ ", " +y+ ", " +z+ "
in constructor 1 is " +sum);
}

//Constructor with two parameters passed
public Operations(int x, int y)
{
   this.x=x;
   this.y=y;
   z=12;
   int sum=x+y+z;
   System.out.println("Sum of numbers " +x+ ", " +y+ ", " +z+ "
in constructor 2 is " +sum);
}

//Constructor with three parameters passed
public Operations(int x, int y, int z)
{
   this.x=x;
   this.y=y;
   this.z=z;
   int sum=x+y+z;
   System.out.println("Sum of numbers " +x+ ", " +y+ ", " +z+ "
in constructor 3 is " +sum);
}
}
public class MainClass
{
   public static void main(String[] args)
   {
      Operations op0 = new Operations();
      Operations op1 = new Operations(5);
```

Output

```
Sum of numbers 5, 7, 10 in constructor 0 is 22
Sum of numbers 5, 22, 12 in constructor 1 is 39
Sum of numbers 7, 3, 12 in constructor 2 is 22
Sum of numbers 2, 9, 23 in constructor 3 is 3
```

```
        Operations op2 = new Operations(7, 3);
        Operations op3 = new Operations(2, 9, 23);
    }
}
```

****

8 . Inheritance

Program	Output

//Single Inheritance
class Animal
{
void eat()
{
System.out.println("eating...");
}
}
 	
class Dog extends Animal
{
void bark()
{
System.out.println("barking...");
}
}
 	
class singleInherit
{
public static void main(String args[])
{
Dog d=new Dog();
d.bark();
d.eat();
}
}
	C:\Javapractice>javac singleInherit.java
	
	C:\Javapractice>java singleInherit
	barking...
	eating...

//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 multilevelInheritance
{
	C:\Javapractice>javac multilevelInheritance.java
	
	C:\Javapractice>java multilevelInheritance
	weeping...
	barking...
	eating...

```
        public static void main(String args[])
        {
                BabyDog d=new BabyDog();
                d.weep();
                d.bark();
                d.eat();
        }
}
```

```
//Hierarchial Inheritance
class Animal
{
        void eat()
        {
                System.out.println("eating...");
        }
}

class Dog extends Animal
{
        void bark()
        {
                System.out.println("barking...");
        }
}

class Cat extends Animal
{
        void meow()
        {
                System.out.println("meowing...");
        }
}

class hierarchialInherit
{
        public static void main(String args[])
        {
                Cat c=new Cat();
           c.meow();
     c.eat();
        }
}
```

```
C:\Javapractice>javac hierarchialInherit.java

C:\Javapractice>java hierarchialInherit
meowing...
eating...
```

****

9. Interface

Program

```
interface Vehicle
{
  void tyres();
  void speed();
  void cost();
  void brand();
}

class Bicycle implements Vehicle
{
   public void tyres()
  {
     System.out.println("Bicycle has two tyres.");
   }
   public void speed()
   {
      System.out.println("Its speed is less compared to
bike and car.");
    }
    public void cost()
   {
      System.out.println("Its cheaper than bikes and
cars.");
    }
    public void brand()
    {
      System.out.println("Hero is a famous brand of
bicycles.");
      System.out.println();
    }
}
class Bike implements Vehicle
{
  public void tyres()
  {
     System.out.println("Bikes also have two tyres.");
   }
   public void speed()
   {
     System.out.println("Its speed is less compared to car
but more than bicycle.");
    }
   public void cost()
   {
      System.out.println("Its more expensive than bicycle
but cheaper than car.");
    }
    public void brand()
    {
       System.out.println("Yamaha is a famous brand of
bike."); System.out.println();
     }

}

class Car implements Vehicle
```

Output

```
C:\Javapractice>javac Int.java

C:\Javapractice>java Int
Bicycle has two tyres.
Its speed is less compared to bike and car.
Its cheaper than bikes and cars.
Hero is a famous brand of bicycles.

Bikes also have two tyres.
Its speed is less compared to car but more than bicycle.
Its more expensive than bicycle but cheaper than car.
Yamaha is a famous brand of bike.

Cars have four tyres.
Its faster than bicycle and bike.
Its most expensive amongst the three.
Maruti Suzuki is a famous brand of car.
```

```
{
   public void tyres()
  {
     System.out.println("Cars have four tyres.");
  }
   public void speed()
  {
      System.out.println("Its faster than bicycle and bike.");
   }
   public void cost()
   {
       System.out.println("Its most expensive amongst the
three.");
    }
    public void brand()
   {
       System.out.println("Maruti Suzuki is a famous brand
of car.");
       System.out.println();
    }
}

class Int
{
    public static void main(String[] args)
    {
        Bicycle bicycle = new Bicycle();
        bicycle.tyres();
        bicycle.speed();
        bicycle.cost();
        bicycle.brand();
        Bike bike = new Bike();
        bike.tyres();
        bike.speed();
        bike.cost();
        bike.brand();
        Car car = new Car();
        car.tyres();
        car.speed();
        car.cost();
        car.brand();
     }
}
```

****

10. Exception Handling

Program	Output

//Exception example 1
public class Exceptions
{
  public static void main(String[] args)
  {
  int a = 5;
  int b = 0;
  int c = a/b;
  System.out.println(c);
  }
}
	C:\Javapractice>javac Exceptions1.java
	
	C:\Javapractice>java Exceptions1
	Exception in thread "main" java.lang.ArithmeticException: / by zero
	  at Exceptions1.main(Exceptions1.java:1)

//Exceptional Example 2
public class Exceptions2
{
  public static void main(String[] args)
  {
  try
  {
  int a = 5;
  int b = 0;
  int c = a/b;
  System.out.println(c);
  }
  catch (ArithmeticException e)
  {
  System.out.println(e);
  System.out.println("Oopss, there is an error!:(");
  }
  }
}
	C:\Javapractice>javac Exceptions2.java
	
	C:\Javapractice>java Exceptions2
	java.lang.ArithmeticException: / by zero
	Oopss, there is an error!:(

//Exception example 3
public class Exceptions3
{
  public static void main(String[] args)
  {
  try
  {
  int[] arr = new int[10];
  arr[12] = 15;
  int a = 5;
  int b = 0;
  int c = a/b;
  System.out.println(c);
  }
  catch (ArithmeticException e)
  {
  System.out.println(e);
  System.out.println("Oopss, there is an arithmetic exception error!:(");
  }
  catch (ArrayIndexOutOfBoundsException e)
  {
	C:\Javapractice>javac Exceptions3.java
	
	C:\Javapractice>java Exceptions3
	java.lang.ArrayIndexOutOfBoundsException: Index 12 out of bounds for
	Oopss, there is an array out of bound exception error!:(

```
                        System.out.println(e);
                        System.out.println("Oopss, there is
an array out of bound exception error!:(");
                }
        }
}
```

```
//Exception example4
public class Exceptions4
{
        public static void main(String[] args)
        {
                try
                {
                        int[] arr = new int[10];
                        arr[12] = 15;
                }
                catch (ArrayIndexOutOfBoundsException e)
                {
                        System.out.println(e);
                        System.out.println("Oopss, there is
an array out of bound exception error!:(");
                }
                try
                {
                        int a = 5;
                        int b = 0;
                        int c = a/b;
                        System.out.println(c);
                }
                catch (ArithmeticException e)
                {
                        System.out.println(e);
                        System.out.println("Oopss, there is
an arithmetic exception error!:(");
                }
        }
}
```

```
C:\Javapractice>javac Exceptions4.java

C:\Javapractice>java Exceptions4
java.lang.ArrayIndexOutOfBoundsException: Index 12 out of bounds for length 10
Oopss, there is an array out of bound exception error!:(
java.lang.ArithmeticException: / by zero
Oopss, there is an arithmetic exception error!:(
```

```
//Exception example 5
public class Exceptions5
{
        public static void main(String[] args)
        {
        try
        {
                int[] arr = new int[10];
                arr[12] = 15;
        }
        catch (ArrayIndexOutOfBoundsException e)
        {
                System.out.println(e);
                System.out.println("Oopss, there is an array
out of bound exception error!:(");
        }
        try
        {
                int a = 5;
                int b = 0;
```

```
C:\Javapractice>javac Exceptions5.java

C:\Javapractice>java Exceptions5
java.lang.ArrayIndexOutOfBoundsException: Index 12 out of bounds for length 10
Oopss, there is an array out of bound exception error!:(

java.lang.ArithmeticException: / by zero
Oopss, there is an arithmetic exception error!:(

Hehehe, I'm executed without any error!:)
```

```
                int c = a/b;
                System.out.println(c);
        }
        catch (ArithmeticException e)
        {
                System.out.println();
                System.out.println(e);
                System.out.println("Oopss, there is an
arithmetic exception error!:(");
        }
        finally
        {
                System.out.println();
                System.out.println("Hehehe, I'm executed
without any error!:)");
        }
}
}
```

```
//Exception example 6
import java.util.Scanner;
public class Exceptions6
{
        public static void main(String[] args)
        {
                validate();
                System.out.println("Kia ora mate, have a
nice day!:)");
        }
        public static void validate()
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter your age: ");
                int age = sc.nextInt();
                if(age<18)
                try
                {
                        throw new
ArithmeticException("You are not eligible to vote!");
                }
                catch (ArithmeticException e)
                {
                        System.out.println(e);
                        System.out.println("Sorry, you're
too young to vote!:(");
                }
        }
}
```

```
C:\Javapractice>javac Exceptions6.java

C:\Javapractice>java Exceptions6
Enter your age:
18
Kia ora mate, have a nice day!:)
```

****

11.1. Vectors

Program	Output

//ADD() method
import java.util.*;
class vector1
{
public static void main(String[] arg)
{
Vector v = new Vector();
v.add(1);
v.add(2);
v.add("Java");
v.add("is easy");
v.add(3);
System.out.println("Vector is " + v);
}
}
	C:\Javapractice>javac vector1.java
	Note: vector1.java uses unchecked or unsafe operations.
	Note: Recompile with -Xlint:unchecked for details.
	
	C:\Javapractice>java vector1
	Vector is [1, 2, Java, is easy, 3]

//addAll()
import java.util.*;
class vector2
{
public static void main(String[] arg)
{
ArrayList arr = new ArrayList();
arr.add(3);
arr.add("Java");
arr.add("is easy");
arr.add(4);
Vector v = new Vector();
v.addAll(arr);
System.out.println("vector v:" + v);
}
}
	C:\Javapractice>javac vector2.java
	Note: vector2.java uses unchecked or unsafe operations.
	Note: Recompile with -Xlint:unchecked for details.
	
	C:\Javapractice>java vector2
	vector v:[3, Java, is easy, 4]

//CLEAR()
import java.util.*;
class vector3
{
public static void main(String[] arg)
{
Vector v = new Vector();
v.add(0, 1);
v.add(1, 2);
v.add(2, "geeks");
v.add(3, "forGeeks");
v.add(4, 3);
System.out.println("Vector is: " + v);
v.clear();
System.out.println("after clearing: " + v);
}
}
	C:\Javapractice>javac vector3.java
	Note: vector3.java uses unchecked or unsafe operations.
	Note: Recompile with -Xlint:unchecked for details.
	
	C:\Javapractice>java vector3
	Vector is: [1, 2, geeks, forGeeks, 3]
	after clearing: []

//CLONE()

```
import java.util.*;
class vector4
{
   public static void main(String[] arg)
   {
      Vector v = new Vector();
      Vector v_clone = new Vector();
      v.add(0, 1);
      v.add(1, 2);
      v.add(2, "Java");
      v.add(3, "forExperts");
      v.add(4, 3);
      v_clone = (Vector)v.clone();
      System.out.println("Clone of v: " + v_clone);
   }

}
```

```
C:\Javapractice>javac vector4.java
Note: vector4.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector4
Clone of v: [1, 2, Java, forExperts, 3]
```

// CONTAINS()

```
import java.util.*;
class vector5
 {
   public static void main(String[] arg)
   {
      Vector v = new Vector();
      v.add(1);
      v.add(2);
      v.add("Java");
      v.add("for Experts");
      v.add(3);
      //checking
      if (v.contains("forExperts"))
         System.out.println("forExperts exists");
   }
}
```

```
C:\Javapractice>javac vector5.java
Note: vector5.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector5
```

//ensureCapacity() method

```
import java.util.*;
class vector6
{
   public static void main(String[] arg)
   {
      Vector v = new Vector();
      v.ensureCapacity(22);
      System.out.println("Minimum capacity: " + v.capacity());
   }
}
```

```
C:\Javapractice>javac vector6.java

C:\Javapractice>java vector6
Minimum capacity: 22
```

//indexOf() method

```
import java.util.*;
class vector8
{
   public static void main(String[] arg)
   {
      Vector v = new Vector();
      v.add(1);
      v.add(2);
      v.add("Java");
      v.add("for beginers");
```

```
C:\Javapractice>javac vector8.java
Note: vector8.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector8
index of Geeks is: -1
```

```
      v.add(4);
      System.out.println("index of Geeks is: " +
v.indexOf("java"));
   }
}
```

//isEmpty() method

```
import java.util.*;
class vector9
{
   public static void main(String[] arg)
   {
      Vector v = new Vector();
      v.add(1);
      v.add(2);
      v.add("Java");
      v.add("for beginers");
      v.add(4);
      v.clear();
      if (v.isEmpty())
         System.out.println("Vector is clear");
   }
}
```

```
C:\Javapractice>javac vector9.java
Note: vector9.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector9
Vector is clear
```

//lastIndexof()

```
import java.util.*;
class vector10
{
   public static void main(String[] arg)
   {
      Vector v = new Vector();
      v.add(1);
      v.add(2);
      v.add("Java");
      v.add("for beginers");
      v.add(4);
      System.out.println("last occurance of 2 is: " +
v.lastIndexOf(2));
   }
}
```

```
C:\Javapractice>javac vector10.java
Note: vector10.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector10
last occurance of 2 is: 1
```

//REMOVE

```
import java.util.*;
class vector11
 {
   public static void main(String[] arg)
   {
      Vector v = new Vector();
      v.add(1);
      v.add(2);
      v.add("Java");
      v.add("for beginers");
      v.add(4);
      v.remove(1);
      System.out.println("after removal: " + v);
   }
}
```

```
C:\Javapractice>javac vector11.java
Note: vector11.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector11
after removal: [1, Java, for beginers, 4]
```

//equals() method

```
import  java.util.*;
class vector12
 {
   public static void main(String[] arg)
   {
      Vector v = new Vector();

      v.add(1);
      v.add(2);
      v.add("Java");
      v.add("forBegineers");
      v.add(4);
      Vector v_2nd = new Vector();
      v_2nd.add(1);
      v_2nd.add(2);
      v_2nd.add("Java");
      v_2nd.add("forBegineers");
      v_2nd.add(4);
      if (v.equals(v_2nd))
         System.out.println("both vectors are equal");
   }
}
```

```
C:\Javapractice>javac vector12.java
Note: vector12.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector12
both vectors are equal
```

//removeElement()

```
import java.util.*;
class vector13
{
   public static void main(String[] arg)
   {
      Vector vec = new Vector(7);
      vec.add(1);
      vec.add(2);
      vec.add(3);
      vec.add(4);
      vec.add(5);
      vec.add(6);
      vec.add(7);
      vec.removeElement(5);
      System.out.println("Vector after removal: " + vec);
   }
}
```

```
C:\Javapractice>javac vector13.java
Note: vector13.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector13
Vector after removal: [1, 2, 3, 4, 6, 7]
```

//capacity() method

```
import java.util.*;
class vector14
{
   public static void main(String[] arg)
   {
      Vector vec = new Vector(7);
      vec.add(1);
      vec.add(2);
      vec.add(3);
      vec.add(4);
      vec.add(5);
      vec.add(6);
      vec.add(7);
      System.out.println("Capacity of vector: " +
vec.capacity());
```

```
C:\Javapractice>javac vector14.java
Note: vector14.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Javapractice>java vector14
Capacity of vector: 7
```

} }	

11.2. Strings

Program	Output

//String to int using Integer.parseInt(String)
class string1
{
  public static void main(String args[])
  {
  String str="123";
  int inum = 100;
  int inum2 = Integer.parseInt(str);
  int sum = inum+inum2;
  System.out.println("Result is: "+sum);
  }
}
	C:\Javapractice>javac string1.java
	
	C:\Javapractice>java string1
	Result is: 223

//string functions
class string2
{
  public static void main(String[] args)
  {
  String targetString = "Java is easy to learn";
  String s1= "JAVA";
  String s2= "Java";
  String s3 = " Hello Java ";
  System.out.println("Char at index 2(third position): " + targetString.charAt(2));
  System.out.println("After Concat: "+ targetString.concat(" and Enjoy"));
  System.out.println("Checking equals ignoring case: " +s2.equalsIgnoreCase(s1));
  System.out.println("Checking equals with case: " +s2.equals(s1));
  System.out.println("Checking Length: "+ targetString.length());
  System.out.println("Replace function: "+ targetString.replace("fun", "easy"));
  System.out.println("SubString of targetString: "+ targetString.substring(8));
  System.out.println("SubString of targetString: "+ targetString.substring(8, 12));
  System.out.println("Converting to lower case: "+ targetString.toLowerCase());
  System.out.println("Converting to upper case: "+ targetString.toUpperCase());
  System.out.println("Triming string: " + s3.trim());
  System.out.println("searching s1 in targetString: " + targetString.contains(s1));
  System.out.println("searching s2 in targetString: " + targetString.contains(s2));
  char [] charArray = s2.toCharArray();
  System.out.println("Size of char array: " + charArray.length);
  System.out.println("Printing last element of array: " + charArray[3]);
  }
	C:\Javapractice>java string2
	Char at index 2(third position): v
	After Concat: Java is easy to learn and Enjoy
	Checking equals ignoring case: true
	Checking equals with case: false
	Checking Length: 21
	Replace function: Java is easy to learn
	SubString of targetString: easy to learn
	SubString of targetString: easy
	Converting to lower case: java is easy to learn
	Converting to upper case: JAVA IS EASY TO LEARN
	Triming string: Hello Java
	searching s1 in targetString: false
	searching s2 in targetString: true
	Size of char array: 4
	Printing last element of array: a

```
}
```

```
//palindrome string
class string3
{
  public static void main(String args[])
  {
    String palindrome = "Hey yeh";
    int len = palindrome.length();
    System.out.println( "String Length is : " + len );
  }
}
```

```
C:\Javapractice>javac string3.java

C:\Javapractice>java string3
String Length is : 7
```

```
//character position
class string4
{
 public static void main(String[] args)
 {
            String str_Sample = "RockStar";
            System.out.println("Character at position 5: " +
str_Sample.charAt(5));
            System.out.println("Index of character 'S': " +
str_Sample.indexOf('S'));
 }
}
```

```
C:\Javapractice>java string4
Character at position 5: t
Index of character 'S': 4
```

```
//Lowercase and Uppercase
class string5
{
 public static void main(String[] args)
 {
          String str_Sample = "Chandrayaan";
          System.out.println("Convert to LowerCase: " +
str_Sample.toLowerCase());
   System.out.println("Convert to UpperCase: " +
str_Sample.toUpperCase());
 }
}
```

```
C:\Javapractice>java string5
Convert to LowerCase: chandrayaan
Convert to UpperCase: CHANDRAYAAN
```

```
//string handling
class string6
{
 public static void main(String args[])
 {
  int n;
  String s = "Java programming", t = "", u = "";
  System.out.println(s);
  n = s.length();
  System.out.println("Number of characters = " + n);
  t = s.replace("Java", "C++");
  System.out.println(s);
  System.out.println(t);
  u = s.concat(" is fun");
  System.out.println(s);
  System.out.println(u);
 }
}
```

```
C:\Javapractice>javac string6.java

C:\Javapractice>java string6
Java programming
Number of characters = 16
Java programming
C++ programming
Java programming
Java programming is fun
```

```
//string concat
class string7
{
	public static void main(String arg[])
	{
		String s1="Aarna";
		String s2="Bafna";
		System.out.println("Combined String:
"+s1.concat(s2));
	}
}
```

```
C:\Javapractice>javac string7.java

C:\Javapractice>java string7
Combined String: AarnaBafna
```

```
//string compare
class string8
{
public static void main(String arg[])
{
	String s1="Aarna";
	String  s2="Bafna";
	String s3="Aarna";
	System.out.println("Compare String:
"+s1.equals(s2));
	System.out.println("Compare String:
"+s1.equals(s3));
}
}
```

```
C:\Javapractice>javac string8.java

C:\Javapractice>java string8
Compare String: false
Compare String: true
```

```
//string compare
class string9
{
	public static void main(String arg[])
	{
		String s1="Aarna";
		String s2="AARNA";
		String s3="Bafna";
		System.out.println("Compare String:
"+s1.equalsIgnoreCase(s2));
		System.out.println("Compare String:
"+s1.equalsIgnoreCase(s3));
	}
}
```

```
C:\Javapractice>javac string9.java

C:\Javapractice>java string9
Compare String: true
Compare String: false
```

```
class string10
{
	public static void main(String arg[])
	{
		String s1="Aarna";
		String s2="Bafna";
		int i;
		i=s1.compareTo(s2);
		if(i==0)
		{
			System.out.println("Strings are
same");
		}
		else
		{
			System.out.println("Strings are not
same");
		}
```

```
C:\Javapractice>javac string10.java

C:\Javapractice>java string10
Strings are not same
```

} }	

****

12. Packages

Program	Output

package letmecalculate;
public class Calculator
{
public int add(int a,int b)
{
return(a+b);
}
public int sub(int a, int b)
{
return(a-b);
}
public int mul(int a,int b)
{
return(a*b);
}
public int div(int a, int b)
{
return(a/b);
}
public static void main(String args[])
{
}
}
import letmecalculate.*;
public class Demo
{
public static void main(String args[])
{
Calculator obj = new Calculator();
System.out.println(obj.add(100,200));
System.out.println(obj.mul(100,200));
System.out.println(obj.sub(100,50));
System.out.println(obj.div(200,2));
}
}
	300
	20000
	50
	100

****

13. Multithreading

Program	Output

import java.util.*;
class bonjour implements Runnable
{
public void run()
{
for(int i =0; i<5;i++)
{
//Thread t1 = currentThread();
// System.out.println(t1.getName());
System.out.println("bonjour");
//if (i==2)
//stop();
try {Thread.sleep(500);}
catch(Exception e){};
}
}
}
class namaskaram implements Runnable
{
public void run()
{
for(int i =0; i<5;i++)
{
//Thread t2 = currentThread();
//System.out.println(t2.getName());
System.out.println("namaskaram");
try {Thread.sleep(500);}
catch(Exception e){};
}
}
}
class Multithreading28
{
public static void main(String args[])
{
namaskaram obj1= new namaskaram();
bonjour obj2= new bonjour();
Thread t1=new Thread(obj1);
Thread t2=new Thread(obj2);
t1.start();
t2.start();
System.out.println(t1.isAlive());
System.out.println(t2.isAlive());
try {t1.join();}
	true
	true
	bonjour
	namaskaram
	namaskaram
	bonjour
	namaskaram
	bonjour
	namaskaram
	bonjour
	namaskaram
	bonjour
	true
	True

catch(Exception e){};
//try {t1.join();}
//catch(Exception e){};
System.out.println(t1.isAlive());
System.out.println(t2.isAlive());
}
}	

****

14. I/O Streams

Program	Output

import java.io.*;
class MyInput2
{
public static void main(String[] args)
throws IOException
{
char c;
InputStreamReader isr = new
InputStreamReader(System.in);
BufferedReader br = new
BufferedReader(isr);
//BufferedReader br = new
BufferedReader(new
InputStreamReader(System.in));
System.out.println("Enter the character");
c = (char) br.read();
System.out.println(c);
}
}
	Enter the character
	a
	a

****

15.Student form using JavaFX

Program	Output

import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.geometry.Pos;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.PasswordField;
import javafx.scene.control.TextField;
import javafx.scene.layout.VBox;
import javafx.stage.Stage;
public class FirstjavaFX extends Application{
 	
@Override
public void start(Stage primaryStage) throws Exception {
 	
VBox vb = new VBox();
vb.setSpacing(10);
vb.setAlignment(Pos.CENTER);
 	
Label l1 = new Label("Your Username");
TextField tx1= new TextField();
tx1.setMaxWidth(160);
 	
Label l2 = new Label("Your Password");
PasswordField tx2 = new PasswordField();
tx2.setMaxWidth(160);
 	
Button button = new Button("LOGIN");
TextField tx3= new TextField();
tx3.setMaxWidth(160);
 	
vb.getChildren().addAll(l1,tx1,l2,tx2,button,tx3);
button.setOnAction(new EventHandler<ActionEvent>() {
@Override
public void handle(ActionEvent arg0) {
String userName = tx1.getText();
	

```
 String password = tx2.getText();
 if (userName.equals("TSEC") &&
password.equals("bandra")) {

 tx3.setText(" Login successful");

 } else {
 tx3.setText(" invalid user");

 }
 }
 });
 Scene scene=new Scene(vb,600,400);
 primaryStage.setTitle("First JavaFX
Application");
 primaryStage.setScene(scene);
 primaryStage.show();
 }
 public static void main (String[] args)
 {
 launch(args);
 }

}
```

****