1)Class :

A) Write a program To display name and age using class .

```
import java.util.*;
class Student
{
String name;
int age;
public void getInfo()
{
System.out.println(" Name is :="+name);
System.out.println("age is :="+age);
}
public static void main(String args[])
{
Student s=new Student();
s.name="avinash";
s.age=21;
s.getInfo();

}
}
```

2)Abstract Class

a) Write a program by Using abstract class To display Employee Name, Salary , age using printInfo method.

```
abstract class Base {
    abstract void printInfo();
}

class Derived extends Base {
    void printInfo()
    {
        String name="avinash";
        float salary = 333.3f;
        int age=21;

        System.out.println(name);
        System.out.println(salary);
        System.out.println(age);
    }
}

class abstraction {
    public static void main(String args[])
    {
        Base b = new Derived();
        b.printInfo();
    }
}
```

b) Write a program To display method print the addition and substraction by using abstraction.

```
abstract class Math {
    abstract void display();

}

class add extends Math {
    public void display() {
        int a=3,b=3;
        System.out.println(a+b);
    }
}

class sub extends Math {

    public void display() {
        int e=3,f=3;
        System.out.println(e-f);
    }

}

class abstraction {
    public static void main(String args[]) {
        add a = new add();
        sub b=new sub();
        a.display();
        b.display();
    }
}
```

3)Interface

A)

Develop a program To display student data by using interface

1) Name
2) Roll no

```
interface Student
{
    public void data();

}
class avi implements Student
{
    public void data ()
    {
        String name="avinash";
        int rollno=68;
        System.out.println(name);
        System.out.println(rollno);
    }
}
public class inter_face
{
    public static void main (String args
    []) {
        avi h= new avi();
        h.data();
    }
}
```

b) Develop a Simple program of interface mentioned below points;

1) Interface name animal
2) Method name walk
3) Implements horse class
4) Display the message is – walks on 4 legs

```
interface animal
{
    public void walk();

}
class horse implements animal
{
    public void walk ()
    {
       System.out.println("walk on 4 legs");
    }
}
public class inter_face
{
    public static void main (String args [])
    {
      horse h = new horse();
      h.walk();
    }
}
```

5) INHERITANCE

A) Single inheritance

Write a program using single inheritance which Is base class is shape and derived class is circle and calculate the area of circle

```
class shape
{
    public void area()
    {
        System.out.println("Display area");
    }

}
class cirle extends shape
{
    public void area (float r)
    {
        System.out.println(3.14*r*r);
    }

}
public class singleinheritance
{
    public static void main (String args [])
    {
        cirle t = new cirle();
        t.area(5);
    }
}
```

c) Multiple inheritance

Write a program using multiple inheritance base class is shape and circle and triangle are derived class and calculate the area of triangle and area of circle.

```
class shape
{
    public void area()
    {
        System.out.println("Display area");
    }

}
class circle extends shape
{
    public void area (float r)
    {

        System.out.println(3.14*r*r);
    }

}
class trianle extends shape
{
    public void area (int l, int h)
    {
        System.out.println(0.5*l*h);
    }
}
public class singleinheritance
{
    public static void main (String args [])
    {
        circle c = new circle();
        c.area(5);
        trianle t=new trianle();
        t.area(4,4);
    }
}
```

6) Overloading

Develop a program which show one method (as you want) is overloaded (method overloading)

```
class Student {
    String name;
    int age;

    public void displayInfo(String name) {
        System.out.println(name);
    }

    public void displayInfo(int age) {
        System.out.println(age);
    }

    public void displayInfo(String name, int age) {
        System.out.println(name);
        System.out.println(age);
    }

    public static void main(String args[]) {
        Student avi = new Student();
        avi.displayInfo("avinash");
        avi.displayInfo(22);
    }
}
```

b) develop a program which implements concepts of overriding(any example)

```
class Animal {
    public void makeSound() {
        System.out.println("Grr...");
    }
}
class Cat extends Animal {
    public void makeSound() {
        System.out.println("Meow");
    }
}
```

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

Exception Handling

A) Write a program to show the arithmetic exception using throws.

```
public class exception {
    static void checkAge(int age) throws ArithmeticException {
      if (age < 18) {
        throw new ArithmeticException("Access denied - You must be at least 18 years old.");
      } else {
        System.out.println("Access granted - You are old enough!");
      }
   }

   public static void main(String[] args) {
     checkAge(15);
   }
  }
```

B)

Write a program to check the arithmetic exception by using try and catch

```
import java.io.*;

class exception {
    public static void main(String[] args) {
        int a = 5;
        int b = 0;
        try {
            System.out.println(a / b);
        } catch (ArithmeticException e) {
            e.printStackTrace();
        }
    }

}
```

7) File System

Develop a program and write it to files by using Formatter class

```
import java.util.*;
public class file {
        public static void main(String args[]) {
            try {
                Formatter f = new Formatter("a.txt");
                f.format("avinash");
                f.format("rathod");
                f.close();

            } catch (Exception e) {
                System.out.println("Error");
            }
        }
    }
```

a) read

develop a program in which read the file by using scanner class.

```
import java.util.Scanner;
import java.io.File;

public class file {
  public static void main(String args[]) throws Exception {

    try {
      File x = new File("a.txt");
      Scanner sc = new Scanner(x);
      while (sc.hasNext()) {
        System.out.println(sc.next());
      }
      sc.close();
    } catch (Exception e) {
        System.out.println("Error");
    }
  }
}
```

8) Swing

a)

Develop a program to select multiple languages known to user.

1)Marathi

2)Hindi

3)English

```

import java.awt.*;

class Lan {
Lan() {
Frame f = new Frame();

Label l1 = new Label("Select known Languages");
l1.setBounds(100, 50, 120, 80);
f.add(l1);

Checkbox c2 = new Checkbox("Hindi");
c2.setBounds(100, 150, 50, 50);
f.add(c2);
Checkbox c3 = new Checkbox("English");
c3.setBounds(100, 200, 80, 50);
f.add(c3);
Checkbox c4 = new Checkbox("marathi");
c4.setBounds(100, 250, 80, 50);
f.add(c4);

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

public static void main(String ar[]) {
new Lan();
}
}
```

b) **Write a program to create three buttons with caption ok , reset, and cancel;**

```
import java.awt.*;

class But {
    But() {
        Frame f = new Frame();
        Button b1 = new Button("Ok");
        b1.setBounds(100, 50, 50, 50);
        f.add(b1);
        Button b2 = new Button("Reset");
        b2.setBounds(100, 101, 50, 50);
        f.add(b2);
        Button b3 = new Button("Cancel");
        b3.setBounds(100, 150, 80, 50);
        f.add(b3);
        f.setSize(500, 500);
        f.setLayout(null);
        f.setVisible(true);
    }

    public static void main(String a[]) {
        new But();
    }
}
```

c) Develop a program using label (swing) to display message “Welcome to java”;

```
import java.awt.*;

class L {
        L() {
                Frame f = new Frame();
                Label l1 = new Label("Welcome to Java");
                l1.setBounds(100, 50, 120, 80);
                f.add(l1);
                f.setSize(500, 500);
                f.setLayout(null);
                f.setVisible(true);
        }

        public static void main(String a[]) {
                new L();
        }
}
```

d)write a program to count the number of clicks performed by the user in a frame window

(using Swing in java)

```
package com.javaguides.javaswing.login;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class CounterTest extends JFrame implements ActionListener
    {  private int count = 0;
    JLabel lblData;
    CounterTest  ()
    {setLayout(new FlowLayout());
        lblData = new JLabel("button clicked a times");
        JButton btn = new JButton("Click me");
        btn.addActionListener((ActionListener) this);
        add(lblData);
        add(btn);

    }
    public void actionPerformed(ActionEvent e) {
        count++;
        lblData.setText(" Button clicked " + count + "times");

    }

    public static void main(String args[]) {

        CounterTest ex = new CounterTest();
        ex.setVisible(true);
    }
}
```

E)

Design an applet/application to create form using Text Field, Text Area, Button and Label.

Label 1 name : Enter your name

Lable 2 name : Adress

Button : submit

```
package com.javaguides.javaswing.login;

import java.awt.*;

public class BasicAWT
{
 public static void main(String args[])
 {
  Frame f = new Frame();
  f.setSize(400,400);
  f.setVisible(true);
  f.setLayout(new FlowLayout() );

  Label l1 = new Label();
  l1.setText("Enter Your Name ");

  TextField tf = new TextField(" enter name");

  Label l2 = new Label("Address");
  TextArea ta = new TextArea("",3,40);

  Button b = new Button("Submit");

  f.add(l1);
  f.add(tf);
  f.add(l2);
  f.add(ta);
  f.add(b);
 }
}
```

9) Multithreading

1) Java program to illustrate and defining thread by Extending thread class.

```
class Test extends Thread
{
    public void run()
    {
        System.out.println("Run method executed by child
    Thread"); }
    public static void main(String[] args)
    {
        Test t = new Test();
        t.start();
        System.out.println("Main method executed by main
    thread"); }
}
```

2) Java program to illustrate and defining implements extending implements class.

```
public class avi implements Runnable
{
       public static void main(String[] args) {
        Thread guruThread1 = new Thread("avi");
        Thread guruThread2 = new Thread("avi");
        guruThread1.start();
        guruThread2.start();
        System.out.println("Thread names are following:");
        System.out.println(guruThread1.getName());
        System.out.println(guruThread2.getName());
    }
    @Override
    public void run() {
    }
}
```

10) Java collections (4)

A) List

Write a program by using arraylist and Add the number 5,2,2,1,3 and remove 2 number index (1) and arranging the ascending order

```
import java.util.ArrayList;
import java.util.Collections;

public class MyClass {
    public static void main(String[ ] args) {
        ArrayList<Integer> number = new ArrayList<Integer>();
        number.add(5);
        number.add(2);
        number.add(2);
        number.add(1);
        number.add(3);
        System.out.println(number);
        number.remove(2);
        Collections.sort(number);
        System.out.println(number);
    }
}
```

C) SET

Write a program using hashset to add the a,b,c alphabet and remove the c alphabet and displayed the set and size of this set.

```
import java.util.HashSet;

public class MyClass {
    public static void main(String[ ] args) {
        HashSet<String> set = new HashSet<String>();
        set.add("A");
        set.add("B");
        set.add("C");
        System.out.println(set);
        set.remove("A");
        System.out.println(set);

        System.out.println(set.size());

    }
}
```

D)

QUEUE

Write a program using queue data structure , add the three aplhabates a,b,c and remove the b .

```
import java.util.*;

public class MyClass {

    public static void main(String args[])
    {
        Queue<String> pq = new PriorityQueue<>();

        pq.add("a");
        pq.add("b");
        pq.add("c");
        System.out.println(pq);
        pq.remove("b");
        System.out.println(pq);

    }
}
```

MAP

A) HASHMAP

Write a program using hashmap and store the student data , key is roll no 1,2,3, etc and value is name of student a,b,c and access the name of roll -no = 2 .

```
import java.util.HashMap;

public class MyClass {
    public static void main(String[ ] args) {
        HashMap<Integer, String> student = new HashMap<Integer,String
        >(); student.put(1, "a");
        student.put(2, "b");
        student.put(3, "c");
        System.out.println(student.get(2));
    }
}
```