

MARYGIRI COLLEGE OF ARTS AND SCIENCE
KOOTHATTUKULAM
(Affiliated to Mahatma Gandhi University)



DEPARTMAENT OF COMPUTER SCIENCE
Record of Practical Work in
WEB PROGRAMMING USING PHP

Name :

Register Number :

Course :

Semester :

Subject :

Year :

Professor In-charge

Head of the Department

Submitted for the practical examination held on

Internal Examiner

External Examiner

I. Basic Programs

1.Swap two numbers.

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Swap

{

int x,y,temp;

public void read()

{

Scanner ob1=new Scanner(System.in);

System.out.println("Enter the value for x and y:");

x=ob1.nextInt();

y=ob1.nextInt();

System.out.println("Before swapping value of x="+x);

System.out.println("Before swapping value of y="+y);

System.out.println(" ");

}

public void find()

{

temp=x;

x=y;

y=temp;

System.out.println("After swapping value of x="+x);

System.out.println("After swapping value of y="+y);
```

```
}
```

```
}
```

```
public class Sample
```

```
{
```

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

```
{
```

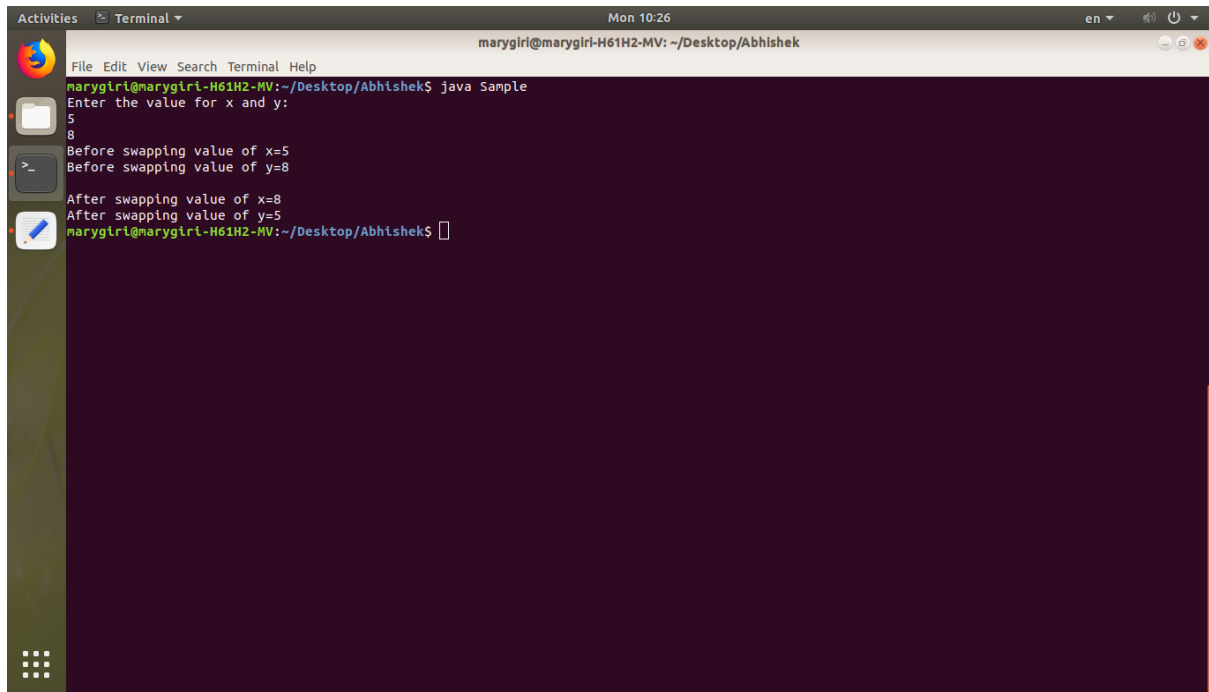
```
Swap ob2=new Swap();
```

```
ob2.read();
```

```
ob2.find();
```

```
}
```

OUTPUT



A screenshot of a Linux terminal window. The window title is "Terminal" and the current directory is "~/Desktop/Abhishek". The user has executed a Java program named "Sample". The program prompts for two values, x and y. The user enters 5 for x and 8 for y. The program then displays the values before and after swapping. The output shows that x is now 8 and y is now 5.

```
marygirl@marygirl-H61H2-MV: ~/Desktop/Abhishek
File Edit View Search Terminal Help
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$ java Sample
Enter the value for x and y:
5
8
Before swapping value of x=5
Before swapping value of y=8

After swapping value of x=8
After swapping value of y=5
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$
```

2. Find max and min number

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Maxmin

{

    int x,y;

    public void read()

    {

        Scanner ob1=new Scanner(System.in);

        System.out.println("Enter the values:");

        x=ob1.nextInt();

        y=ob1.nextInt();

    }

    public void find()

    {

        if(x>y)

        {

            System.out.println("Maximum="+x);

            System.out.println("Minimum="+y);

        }

        else

        {

            System.out.println("Maximum="+y);
```

```
System.out.println("Minimum="+x);
```

```
}
```

```
}
```

```
}
```

```
public class Sample1
```

```
{
```

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

```
{
```

```
Maxmin ob2=new Maxmin();
```

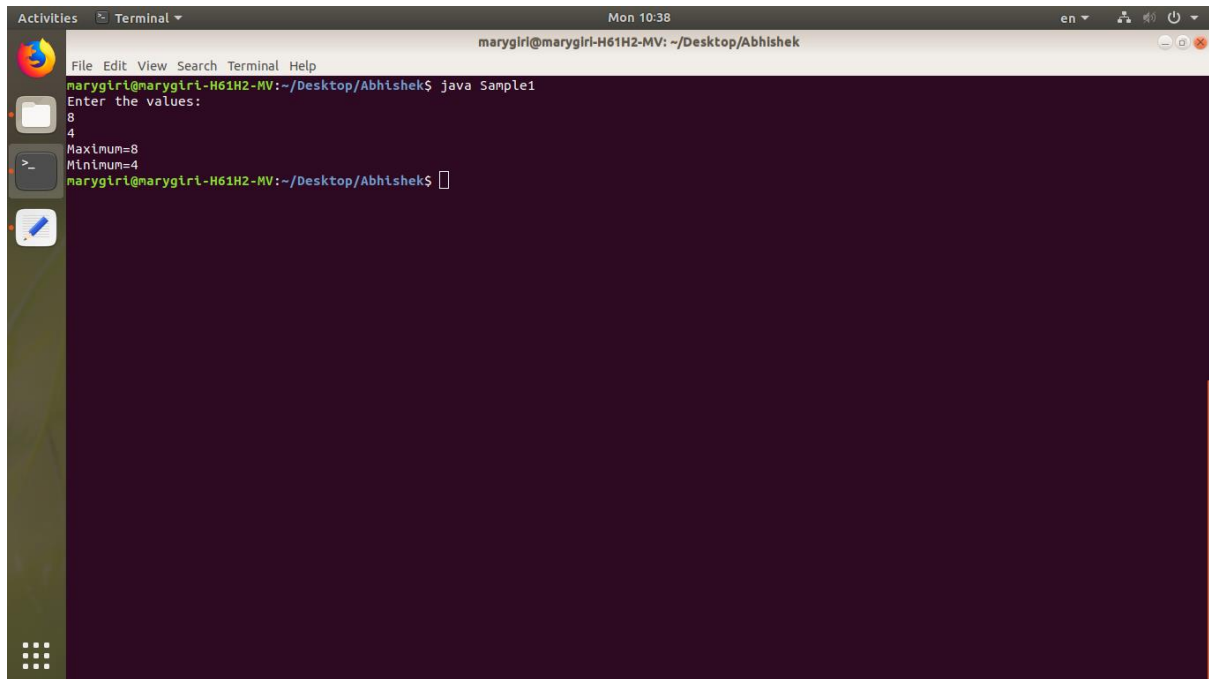
```
ob2.read();
```

```
ob2.find();
```

```
}
```

```
}
```

OUTPUT



The screenshot shows a terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Mon 10:38, en, and system icons). The terminal content is as follows:

```
marygirl@marygirl-H61H2-MV: ~/Desktop/Abhishek
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$ java Sample1
Enter the values:
8
4
Maximum=8
Minimum=4
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$
```


3. Find the factorial

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Factorial

{

int n,i,fact=1;

public void read()

{

Scanner ob1=new Scanner(System.in);

System.out.println("Enter the number:");

n=ob1.nextInt();

}

public void find()

{

for(i=1;i<=n;i++)

{

fact=fact*i;

}

System.out.println("Factorial="+fact);

}

}

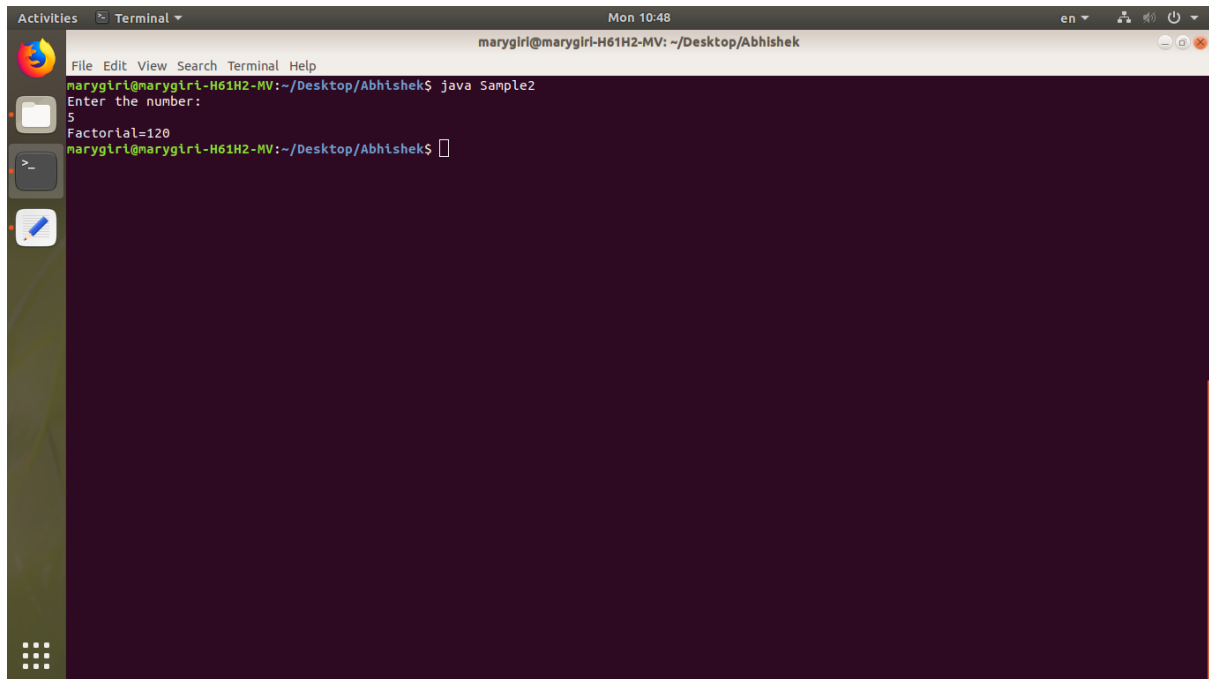
public class Sample2

{

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

```
{  
Factorial ob2=new Factorial();  
ob2.read();  
ob2.find();  
}  
}
```

OUTPUT



The image shows a terminal window titled "Terminal" with a menu bar containing "File", "Edit", "View", "Search", "Terminal", and "Help". The window title bar also displays "Mon 10:48" and "en". The terminal content shows the following sequence of commands and output:

```
marygirl@marygirl-H61H2-MV: ~/Desktop/Abhishek  
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$ java Sample2  
Enter the number:  
5  
Factorial=120  
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$
```

The terminal window has a dark purple background and a light gray border. The left sidebar shows icons for the Dash, Home, and Files applications. The bottom of the window shows a grid of application icons.

4. Check Armstrong number

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Armstrong

{

    int n,num,r,sum=0;

    public void read()

    {

        Scanner ob1=new Scanner(System.in);

        System.out.println("Enter the number:");

        n=ob1.nextInt();

        num=n;

    }

    public void find()

    {

        while(num>0)

        {

            r=num%10;

            sum=sum+(r*r*r);

            num=num/10;

        }

        if(n==sum)

        {
```

```
System.out.println(n+" "+"is an amstrong number");

}

else

{

System.out.println(n+" "+"is not an amstrong number");

}

}

}

public class Sample3

{

public static void main(String args[])

{

Amstrong ob2=new Amstrong();

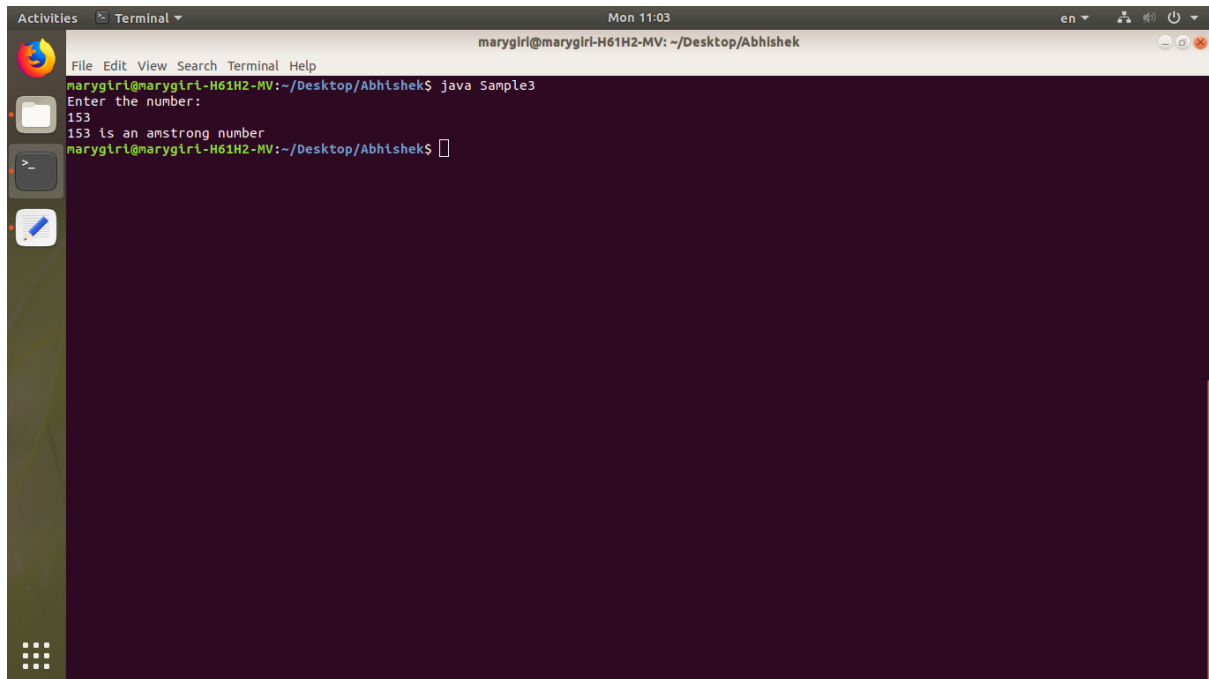
ob2.read();

ob2.find();

}

}
```

OUTPUT



The screenshot shows a Linux terminal window titled "Terminal" with a menu bar containing "File", "Edit", "View", "Search", "Terminal", and "Help". The window's title bar also displays "Mon 11:03", "en", and system icons. The terminal content shows the user "marygirl" at host "marygirl-H61H2-MV" in the directory "~/Desktop/Abhishek". The user runs the command "java Sample3". The program prompts "Enter the number:" and the user enters "153". The program then outputs "153 is an amstrong number". The prompt "marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek\$" is visible at the end of the line.

```
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$ java Sample3
Enter the number:
153
153 is an amstrong number
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$
```

5. Reverse the number

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Reverse

{

int n,r=0,rem;

public void read()

{

Scanner ob1=new Scanner(System.in);

System.out.println("Enter the number:");

n=ob1.nextInt();

}

public void find()

{

while(n>0)

{

rem=n%10;

r=r*10+rem;

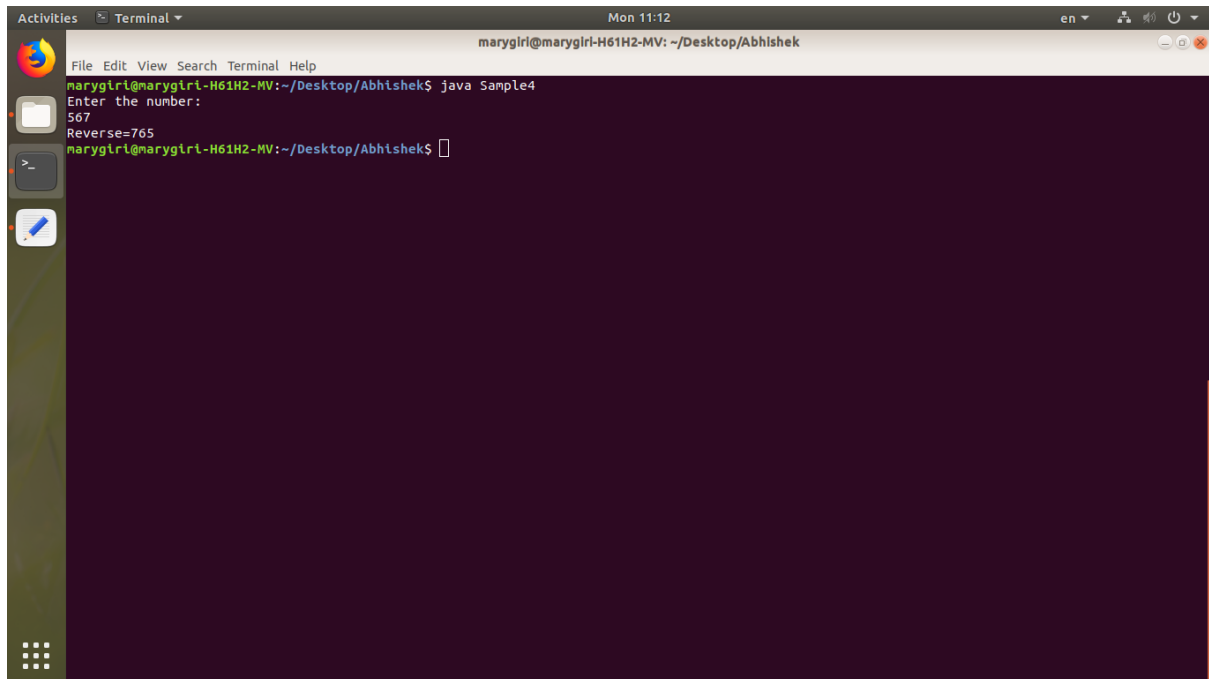
n=n/10;

}

System.out.println("Reverse="+r);
```

```
}  
  
}  
  
public class Sample4  
  
{  
  
    public static void main(String args[])  
  
    {  
  
        Reverse ob2=new Reverse();  
  
        ob2.read();  
  
        ob2.find();  
  
    }  
  
}
```


OUTPUT



The screenshot shows a terminal window titled "Terminal" with a menu bar containing "File", "Edit", "View", "Search", "Terminal", and "Help". The window title bar also displays "Mon 11:12" and "en". The terminal content shows the user "marygiri" at host "marygiri-H61H2-MV" in the directory "~/Desktop/Abhishek". The user has executed the command "java Sample4". The program prompts "Enter the number:" and the user has entered "567". The program then outputs "Reverse=765". The terminal prompt is now "marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek\$".

```
marygiri@marygiri-H61H2-MV: ~/Desktop/Abhishek
File Edit View Search Terminal Help
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$ java Sample4
Enter the number:
567
Reverse=765
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$
```

6. Write a Java program to accept a number from user and count zero, odd and even digits of the entered number.

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Count

{

    int n,zcount,ocount,ecount,r;

    public void read()

    {

        Scanner ob1=new Scanner(System.in);

        System.out.println("Enter the number:");

        n=ob1.nextInt();

    }

    public void find()

    {

        while(n>0)

        {

            r=n%10;

            if(r==0)

            {

                zcount++;

            }

            else if(r%2==0 && r!=0)
```

```
{  
    ecount++;  
}  
else  
{  
    ocount++;  
}  
n=n/10;  
}  
}  
  
public void display()  
{  
    System.out.println(" ");  
    System.out.println("No.of Zero's="+zcount);  
    System.out.println("No.of Even's="+ecount);  
    System.out.println("No.of Odd's="+ocount);  
}  
}  
  
public class Sample8  
{  
    public static void main(String args[])  
    {  
        Count ob1=new Count();
```

```
ob1.read();
```

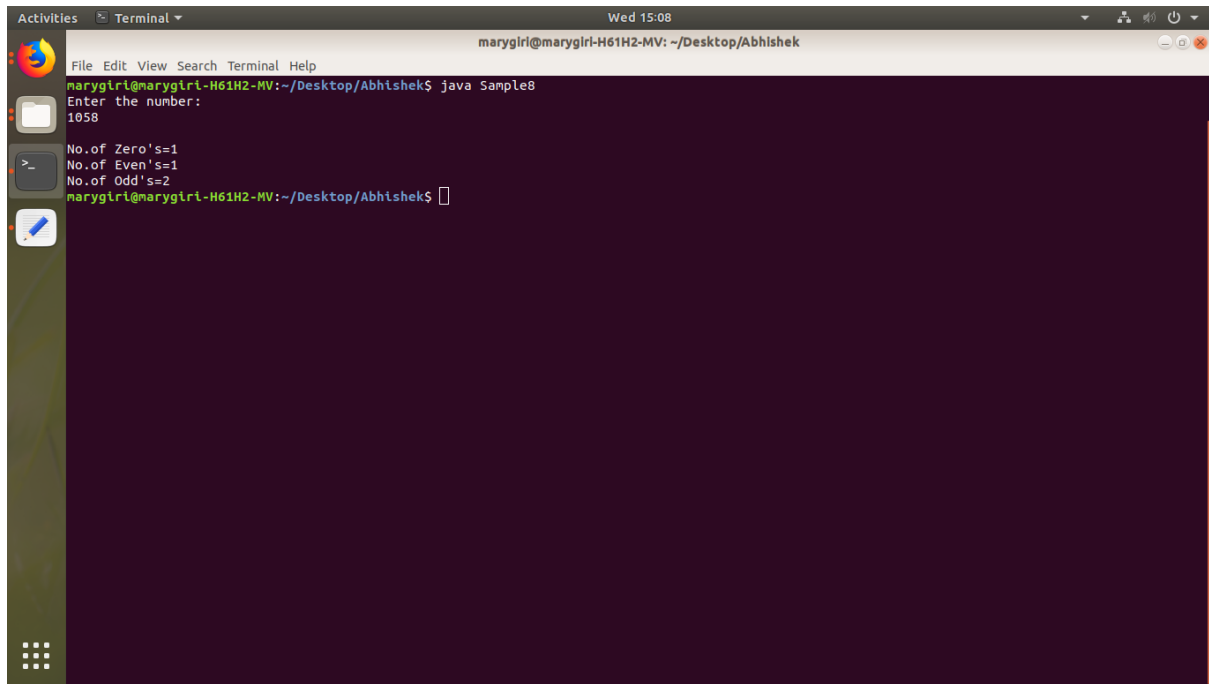
```
ob1.find();
```

```
ob1.display();
```

```
}
```

```
}
```

OUTPUT



The screenshot shows a terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Wed 15:08). The terminal content is as follows:

```
marygiri@marygiri-H61H2-MV: ~/Desktop/Abhishek
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$ java Sample8
Enter the number:
1058
No.of Zero's=1
No.of Even's=1
No.of Odd's=2
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$
```

The terminal output displays the results of the Java program for the input number 1058: 1 zero, 1 even digit, and 2 odd digits.

II. Method Overloading

1.Program to find area of square and rectangle.

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Area

{

int l,b,area,s;

public void ar(int x,int y)

{

l=x;

b=y;

area=l*b;

System.out.println("Area of Rectangle="+area);

}

public void ar(int x)

{

s=x;

area=s*s;

System.out.println("Area of Square="+area);

}

}

public class Sample6

{
```

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

```
{
```

```
    Area ob2=new Area();
```

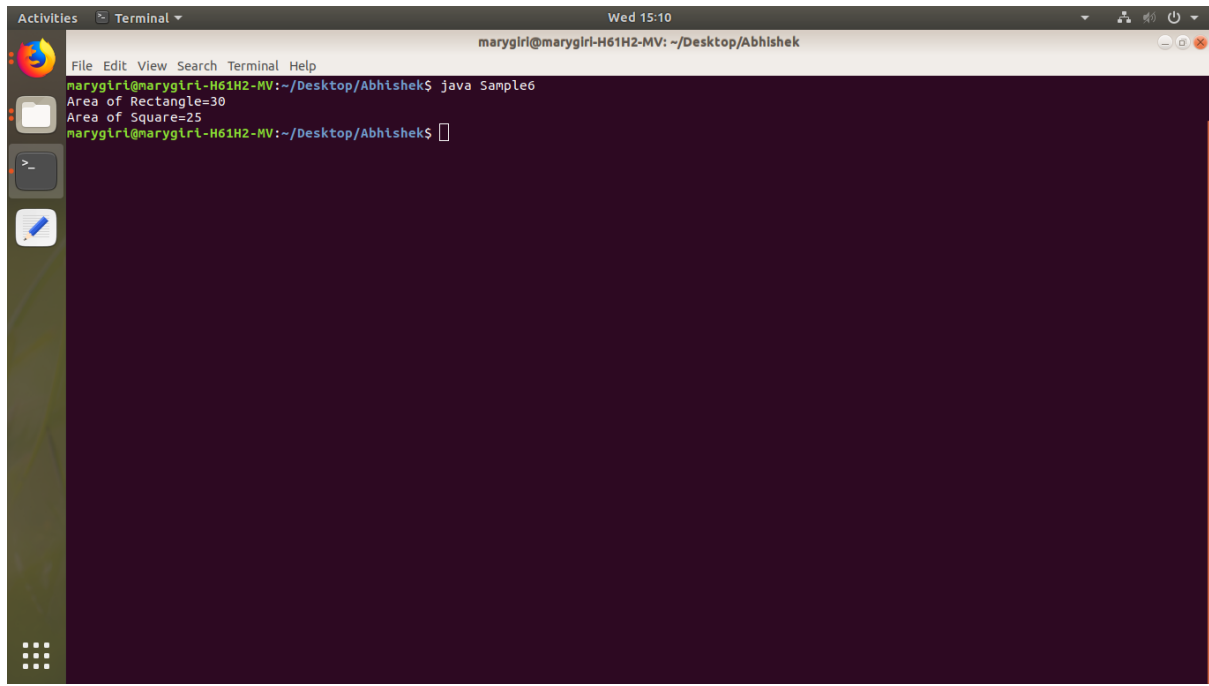
```
    ob2.ar(5,6);
```

```
    ob2.ar(5);
```

```
}
```

```
}
```


OUTPUT



The image shows a screenshot of a Linux terminal window. The window title is "Terminal" and the current directory is "~/Desktop/Abhishek". The terminal displays the output of a Java program named "Sample6". The output consists of three lines: "Area of Rectangle=30", "Area of Square=25", and a prompt character. The terminal has a dark purple background and a light gray border. The window is part of a desktop environment with a sidebar on the left containing icons for Activities, Home, and Applications. The top of the window shows the system clock as "Wed 15:10" and various system icons.

```
marygiri@marygiri-H61H2-MV: ~/Desktop/Abhishek
File Edit View Search Terminal Help
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$ java Sample6
Area of Rectangle=30
Area of Square=25
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$
```

2. Program to find volume of cube and rectangular box.

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Volume

{

int l,b,h,v,a;

public void vol(int x,int y,int z)

{

l=x;

b=y;

h=z;

v=l*b*h;

System.out.println("Volume of Rectangle="+v);

}

public void vol(int x)

{

a=x;

v=a*a*a;

System.out.println("Volume of Cube="+v);

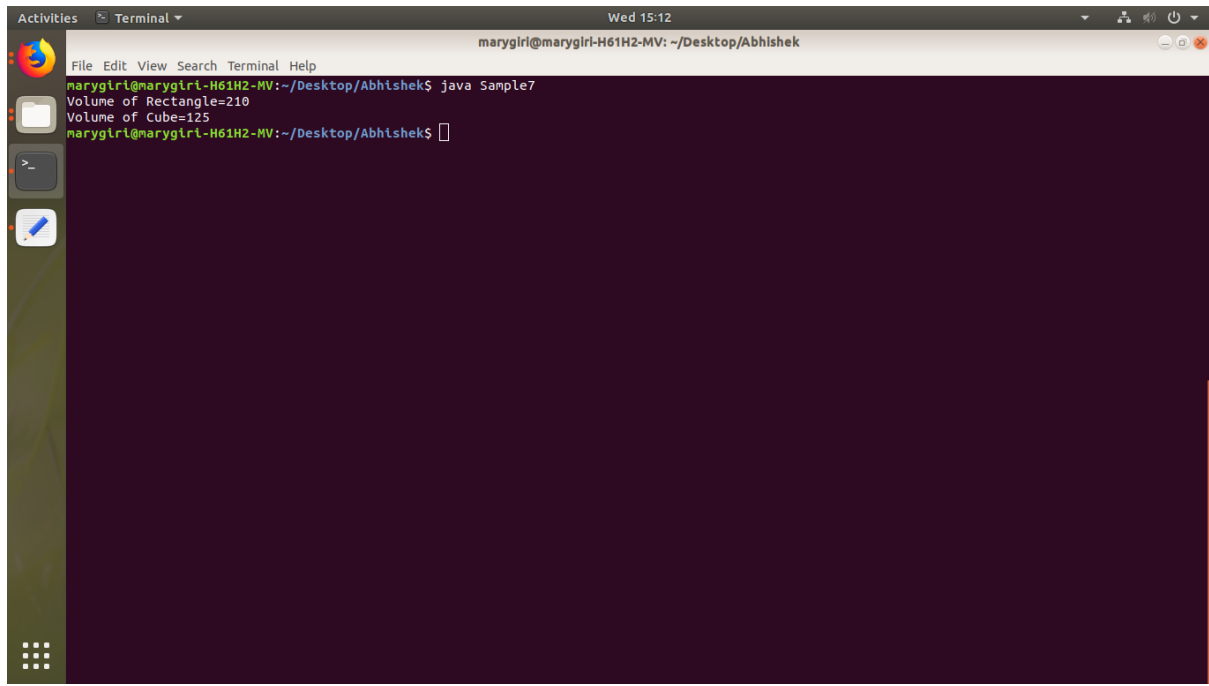
}

}

public class Sample7
```

```
{  
  
public static void main(String args[])  
  
{  
  
Volume ob2=new Volume();  
  
ob2.vol(5,6,7);  
  
ob2.vol(5);  
  
}  
  
}
```

OUTPUT



The screenshot shows a terminal window titled "Terminal" with a menu bar containing "File", "Edit", "View", "Search", "Terminal", and "Help". The window's title bar also displays "Wed 15:12" and system icons. The terminal content shows the user "marygiri" at host "marygiri-H61H2-MV" in the directory "~/Desktop/Abhishek". The command "java Sample7" has been executed, resulting in the output "Volume of Rectangle=210" and "Volume of Cube=125". The prompt "marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek\$" is visible at the bottom of the terminal.

```
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$ java Sample7
Volume of Rectangle=210
Volume of Cube=125
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$
```

III. Constructor Overloading

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Student

{

int age,rollno;

String name;

Student(String x,int y)

{

name=x;

rollno=y;

}

Student(String x,int y,int a)

{

name=x;

rollno=y;

age=a;

}

public void display()

{

System.out.println("Name Of student:"+name);

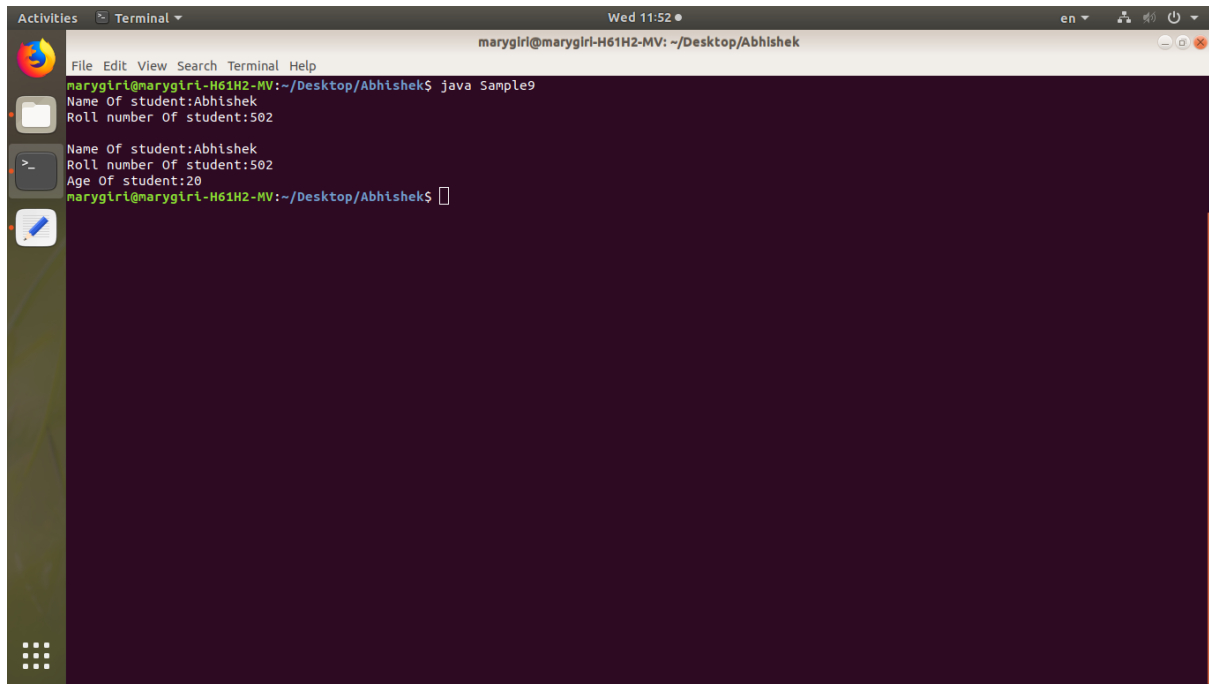
System.out.println("Roll number Of student:"+rollno);

}

public void display1()
```

```
{  
  
System.out.println(" ");  
  
System.out.println("Name Of student:"+name);  
  
System.out.println("Roll number Of student:"+rollno);  
  
System.out.println("Age Of student:"+age);  
  
}  
  
}  
  
public class Sample9  
  
{  
  
public static void main(String args[])  
  
{  
  
Student ob1=new Student("Abhishek",502);  
  
Student ob2=new Student("Abhishek",502,20);  
  
ob1.display();  
  
ob2.display1();  
  
}  
  
}
```

OUTPUT

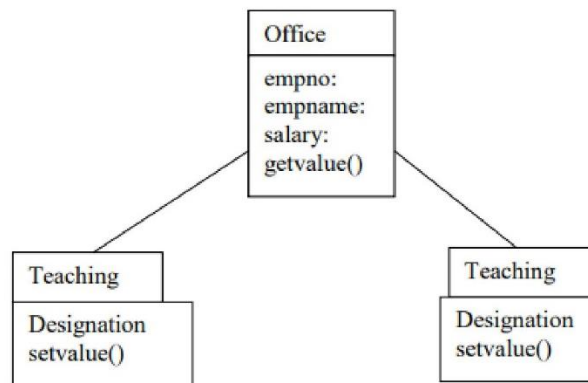


The screenshot shows a terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Wed 11:52, en, and system icons). The terminal content is as follows:

```
marygiri@marygiri-H61H2-MV: ~/Desktop/Abhishek
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$ java Sample9
Name Of student:Abhishek
Roll number Of student:502
Name Of student:Abhishek
Roll number Of student:502
Age Of student:20
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$
```


IV. Inheritance

1. Write a java program that implements educational hierarchy using inheritance.



SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

class Employee

{

    int empno;

    String name;

    float salary;

    public void read()

    {

        Scanner ob1=new Scanner(System.in);

        System.out.print("Enter name of employee:");

        name=ob1.nextLine();

        System.out.print("Enter employee id:");

        empno=ob1.nextInt();
```

```
System.out.print("Enter salary of employee:");

salary=ob1.nextFloat();

}

public void display()

{

System.out.println("Employee id:"+empno);

System.out.println("Name of employee:"+name);

System.out.println("Salary of employee:"+salary);

}

}

class Teaching1 extends Employee

{

String des;

public void setValue()

{

Scanner ob4=new Scanner(System.in);

System.out.print("Enter designation of employee:");

des=ob4.nextLine();

}

public void display1()

{

System.out.println("Designation of employee:"+des);

}
```

```
}

class Teaching2 extends Employee

{

String des1;

public void setValue1()

{

Scanner ob5=new Scanner(System.in);

System.out.print("Enter designation of employee:");

des1=ob5.nextLine();

}

public void display2()

{

System.out.println("Designation of employee:"+des1);

}

}

public class Sample10

{

public static void main(String args[])

{

Teaching1 ob2=new Teaching1();

Teaching2 ob3=new Teaching2();

ob2.read();

ob2.setValue();

ob2.display();
```

```
ob2.display1();
```

```
ob3.read();
```

```
ob3.setValue1();
```

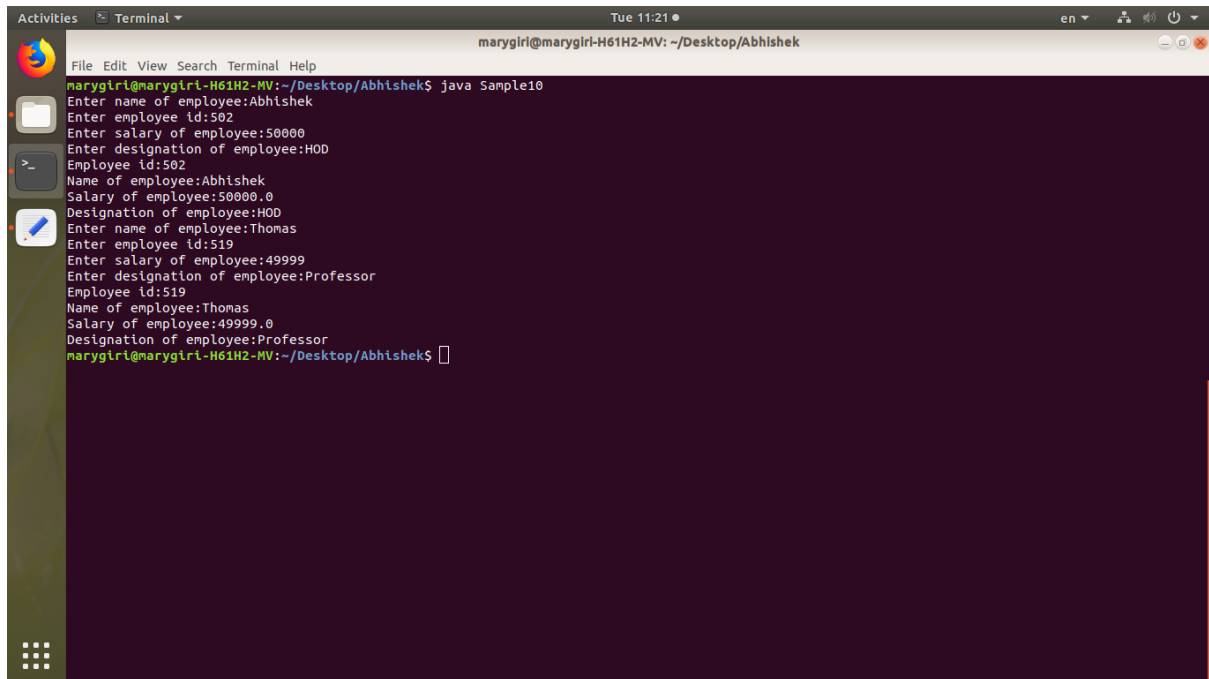
```
ob3.display();
```

```
ob3.display2();
```

```
}
```

```
}
```

OUTPUT



```
marygirl@marygirl-H61H2-MV: ~/Desktop/Abhishek
File Edit View Search Terminal Help
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$ java Sample10
Enter name of employee:Abhishek
Enter employee id:502
Enter salary of employee:50000
Enter designation of employee:HOD
Employee id:502
Name of employee:Abhishek
Salary of employee:50000.0
Designation of employee:HOD
Enter name of employee:Thomas
Enter employee id:519
Enter salary of employee:49999
Enter designation of employee:Professor
Employee id:519
Name of employee:Thomas
Salary of employee:49999.0
Designation of employee:Professor
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$
```

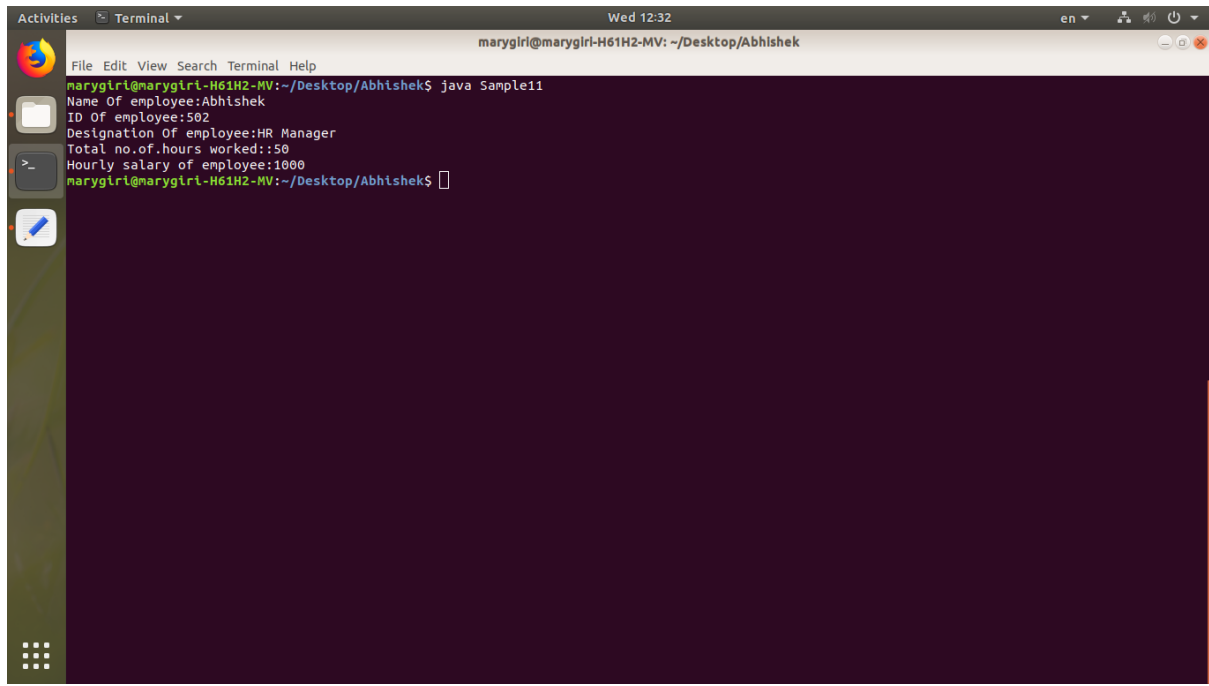
2. Define a class Employee with empno, name, designation. Define a class partEmp which is the subclass Of Employee has its data member as noofhrsworked and hourlySal, Write a complete program to implement it(use consturctors to initialize the values.)

SOURCE CODE

```
import java.lang.*;
import java.util.Scanner;
class Employee
{
    int empno;
    String name,des;
    Employee(String b,String c,int a)
    {
        name=b;
        des=c;
        empno=a;
    }
    public void display()
    {
        System.out.println("Name Of employee:"+name);
        System.out.println("ID Of employee:"+empno);
        System.out.println("Designation Of employee:"+des);
    }
}
class partEmp extends Employee
{
    int hrswork;
    int hsal;
    partEmp(int x,int y,String b,String c,int a)
    {
        super(b,c,a);
        hrswork=x;
        hsal=y;
    }
}
```

```
public void show()
{
    System.out.println("Total no.of.hours worked::"+hrswork);
    System.out.println("Hourly salary of employee:"+hsal);
}
}
public class Sample11
{
    public static void main(String args[])
    {
        partEmp ob1=new partEmp(50,1000,"Abhishek","HR Manager",502);
        ob1.display();
        ob1.show();
    }
}
```


OUTPUT



The screenshot shows a terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Wed 12:32, en, and system icons). The terminal content is as follows:

```
marygiri@marygiri-H61H2-MV: ~/Desktop/Abhishek
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$ java Sample11
Name Of employee:Abhishek
ID Of employee:502
Designation Of employee:HR Manager
Total no.of.hours worked::50
Hourly salary of employee:1000
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$
```

3. Program to show Dynamic method dispatch

SOURCE CODE

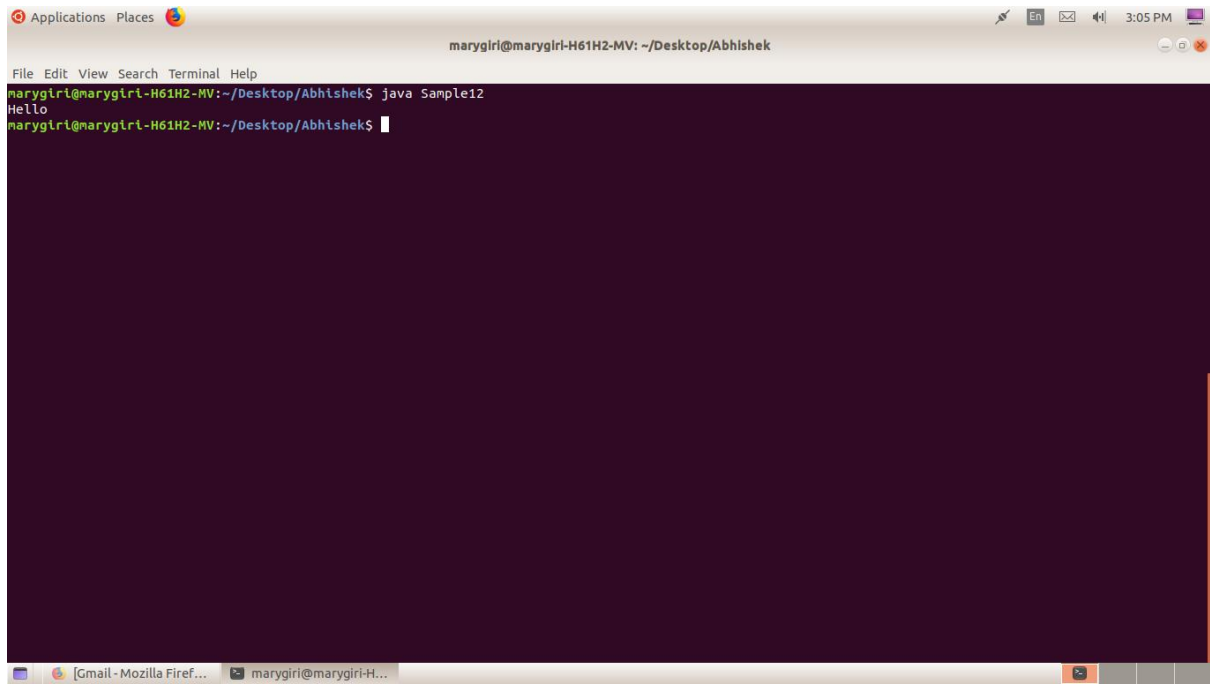
```
import java.lang.*;

class A
{
    public void display()
    {
        System.out.println("Hai");
    }
}

class B extends A
{
    public void display()
    {
        System.out.println("Hello");
    }
}

public class Sample12
{
    public static void main(String args[])
    {
        A ob1=new B();
        ob1.display();
    }
}
```

OUTPUT



The image shows a screenshot of a Linux terminal window. The window's title bar reads "marygiri@marygiri-H61H2-MV: ~/Desktop/Abhishek". The terminal has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The command prompt shows the user "marygiri" at host "marygiri-H61H2-MV" in the directory "~/Desktop/Abhishek". The user has entered the command "java Sample12". The output of the program is "Hello". The terminal window is overlaid on a desktop environment with a taskbar at the bottom showing icons for "Gmail - Mozilla Firef..." and "marygiri@marygiri-H...". The system clock in the top right corner indicates "3:05 PM".

```
marygiri@marygiri-H61H2-MV: ~/Desktop/Abhishek
File Edit View Search Terminal Help
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$ java Sample12
Hello
marygiri@marygiri-H61H2-MV:~/Desktop/Abhishek$
```

4. Create an abstract class called Figure which contains three data members (length, breadth and height). Include an abstract method to find the area. Figure class also contains concrete methods to read the data members and to display them. Derive two classes Rectangle and Triangle from Figure and override area () to find the area of a rectangle and triangle.

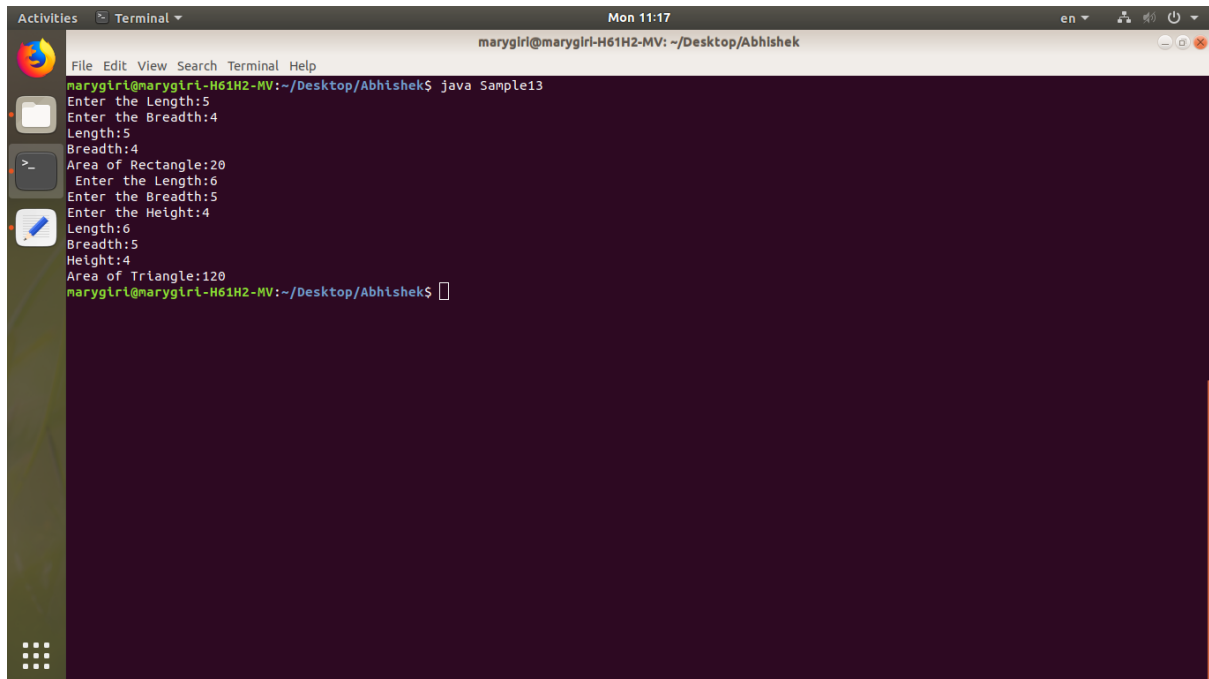
SOURCE CODE

```
import java.lang.*;
import java.util.Scanner;
abstract class Figure
{
    int l,b,h,ar;
    abstract public void area();
    public void read()
    {
        Scanner ob1=new Scanner(System.in);
        System.out.print("Enter the Length:");
        l=ob1.nextInt();
        System.out.print("Enter the Breadth:");
        b=ob1.nextInt();
    }
    public void read1()
    {
        Scanner ob4=new Scanner(System.in);
        System.out.print(" ");
        System.out.print("Enter the Length:");
        l=ob4.nextInt();
        System.out.print("Enter the Breadth:");
        b=ob4.nextInt();
        System.out.print("Enter the Height:");
        h=ob4.nextInt();
    }
    public void display()
    {
```

```
System.out.println("Length:"+l);
System.out.println("Breadth:"+b);
}
public void display1()
{
System.out.println("Length:"+l);
System.out.println("Breadth:"+b);
System.out.println("Height:"+h);
}
}
class Rectangle extends Figure
{
public void area()
{
ar=l*b;
System.out.println("Area of Rectangle:"+ar);
}
}
class Triangle extends Figure
{
public void area()
{
ar=l*b*h;
System.out.println("Area of Triangle:"+ar);
}
}
public class Sample13
{
public static void main(String args[])
{
Rectangle ob2=new Rectangle();
Triangle ob3=new Triangle();
ob2.read();
ob2.display();
```

```
ob2.area();  
ob3.read1();  
ob3.display1();  
ob3.area();  
}  
}
```

OUTPUT



A terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Mon 11:17, en, and system icons). The window shows the execution of a Java program named "Sample13". The user enters the command "java Sample13". The program prompts for "Enter the Length:5", "Enter the Breadth:4", and "Length:5", then outputs "Breadth:4" and "Area of Rectangle:20". It then prompts for "Enter the Length:6", "Enter the Breadth:5", and "Enter the Height:4", then outputs "Length:6", "Breadth:5", and "Height:4". Finally, it outputs "Area of Triangle:120". The prompt "marygirl@marygirl-H61H2-MV: ~/Desktop/Abhishek" is visible at the bottom.

```
marygirl@marygirl-H61H2-MV: ~/Desktop/Abhishek$ java Sample13
Enter the Length:5
Enter the Breadth:4
Length:5
Breadth:4
Area of Rectangle:20
Enter the Length:6
Enter the Breadth:5
Enter the Height:4
Length:6
Breadth:5
Height:4
Area of Triangle:120
marygirl@marygirl-H61H2-MV: ~/Desktop/Abhishek$
```

V Interface, Arrays

1. Create an interface Department containing attributes deptName and deptHead.it as an abstract method showData() for printing the attribute. Create a class Hostel containing hostelname, hostellocation and noofrooms and also have methods readData() and printData() for reading and printing the details. Then write another class named Student extending the Hostel class and implementing the Department interface. This class contains which contains the attributes studname, regno, electivesub and avgmark and use readData() and showData() for reading and printing the details.

SOURCE CODE

```
import java.lang.*;

import java.util.Scanner;

interface Department

{

String deptName="CS";

String deptHead="ABC";

abstract void showData();

}

class Hostel

{

String hostelname,hostellocation;

int noofrooms;

public void readData()

{

Scanner ob1=new Scanner(System.in);
```

```
System.out.print("Enter the hostel name:");

hostelname=ob1.nextLine();

System.out.print("Enter the hostel location:");

hostellocation=ob1.nextLine();

System.out.print("Enter the no of rooms:");

noofrooms=ob1.nextInt();

}

public void printData()

{

System.out.println("Hostel name:"+hostelname);

System.out.println("Hostel location:"+hostellocation);

System.out.println("Number of rooms:"+noofrooms);

}

}

class Student extends Hostel implements Department

{

public void show()

{

System.out.println("Department Name:"+deptName);

System.out.println("Department Head:"+deptHead);

}

String studname,electivesub;

int regno,avgmark;

public void readData1()
```

```
{  
  
Scanner ob2=new Scanner(System.in);  
  
System.out.print("Enter the student name:");  
  
studname=ob2.nextLine();  
  
System.out.print("Enter the elective subject:");  
  
electivesub=ob2.nextLine();  
  
System.out.print("Enter the register number:");  
  
regno=ob2.nextInt();  
  
System.out.print("Enter the average mark:");  
  
avgmark=ob2.nextInt();  
  
}  
  
public void showData()  
  
{  
  
System.out.println("Student name:"+studname);  
  
System.out.println("Student register number:"+regno);  
  
System.out.println("Elective subject:"+electivesub);  
  
System.out.println("Average mark:"+avgmark);  
  
}  
  
}  
  
public class Sample15  
  
{  
  
public static void main(String args[])  
  
{  
  
Student ob3=new Student();
```

```
ob3.readData();
```

```
ob3.printData();
```

```
ob3.show();
```

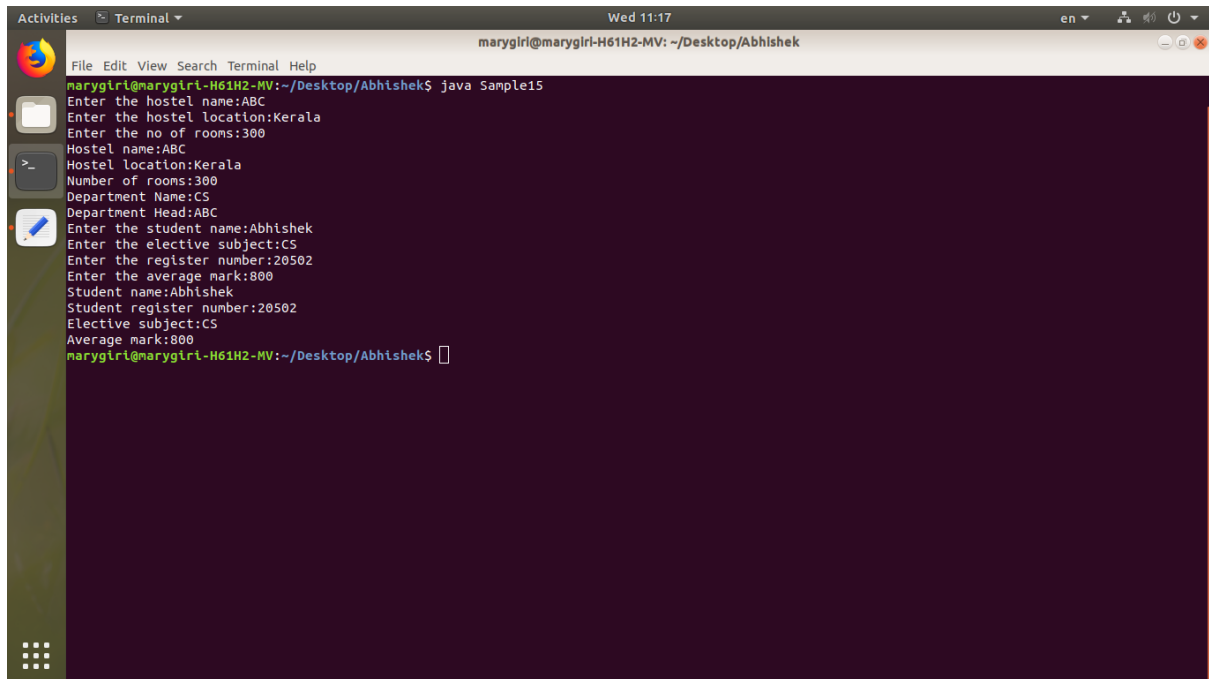
```
ob3.readData1();
```

```
ob3.showData();
```

```
}
```

```
}
```

OUTPUT



```
marygirl@marygirl-H61H2-MV: ~/Desktop/Abhishek
File Edit View Search Terminal Help
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$ java Sample15
Enter the hostel name:ABC
Enter the hostel location:Kerala
Enter the no of rooms:300
Hostel name:ABC
Hostel location:Kerala
Number of rooms:300
Department Name:CS
Department Head:ABC
Enter the student name:Abhishek
Enter the elective subject:CS
Enter the register number:20502
Enter the average mark:800
Student name:Abhishek
Student register number:20502
Elective subject:CS
Average mark:800
marygirl@marygirl-H61H2-MV:~/Desktop/Abhishek$
```