

Defaulter Practical: Interface

Name:- Vedika Kawankar

Class:- SY-03

Batch:- J batch

PRN :- 20101103198

Subject:- OOPS

Defaulter practical:- Interface

10.2 Code with Error

```
interface FamousLine
{
    void Showline();
}
class Nove11 implements FamousLine
{
    public void ShowLine()
    {
        System.out.println("To be, or not to be");
    }
}
class Nove12 implements FamousLine
{
    public void AuthorName()
    {
        System.out.println("Shakespeare");
    }
}
public class UseInterface
{
    public static void main(String args[])
    {
        Nove11 hamlet = new Nove11();
        Nove12 juliet = new Nove12();
        hamlet.ShowLine();
        juliet.AuthorName();
    }
}
```

Output:-

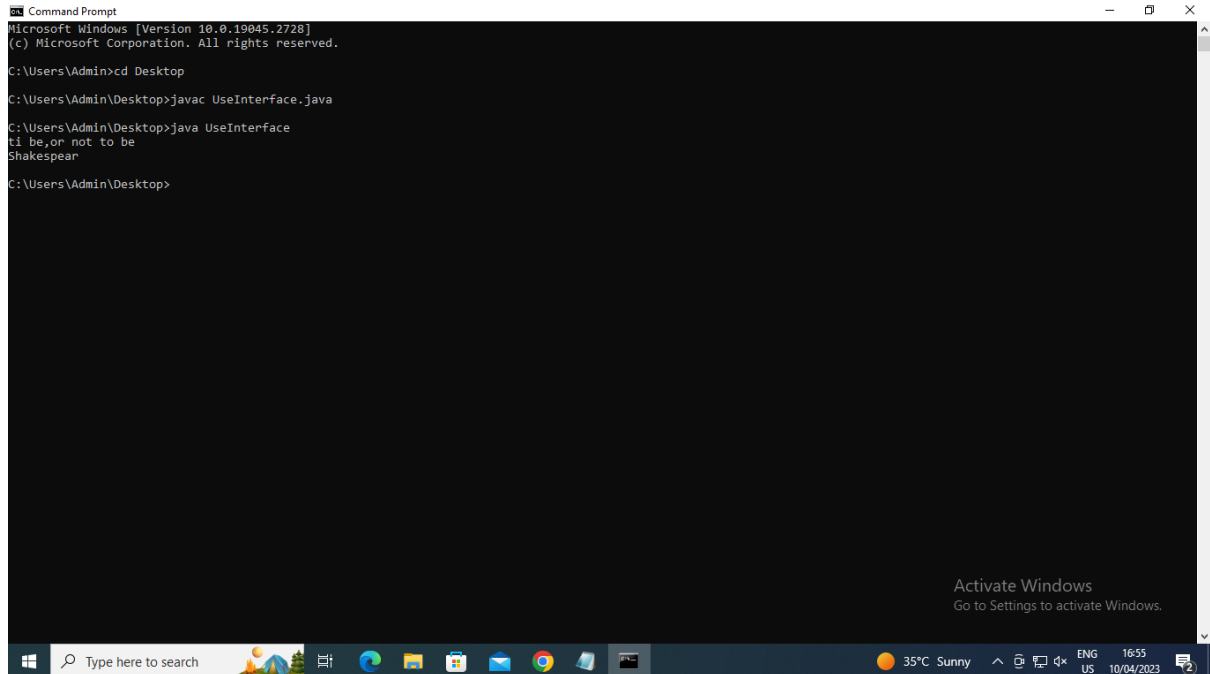
```
Command Prompt
Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd Desktop
C:\Users\Admin\Desktop>javac UseInterface.java
UseInterface.java:6: error: Novell1 is not abstract and does not override abstract method Showline() in Famousline
class Novell1 implements Famousline
^
UseInterface.java:13: error: Novel2 is not abstract and does not override abstract method Showline() in Famousline
class Novel2 implements Famousline
^
2 errors
C:\Users\Admin\Desktop>
```

10.2 Code Without Error

```
interface Famousline
{
    void showline();
}
class Novell1 implements Famousline
{
    public void showline()
    {
        System.out.println("ti be,or not to be");
    }
}
class Novel2 implements Famousline
{
    public void showline()
    {
        System.out.println("use of interface in novel2");
    }
    public void authorname()
    {
        System.out.println("Shakespear");
    }
}
public class UseInterface
{
    public static void main(String args[])
    {
        Novell1 hamlet=new Novell1();
        Novel2 juliet=new Novel2();
        hamlet.showline();
        juliet.authorname();
    }
}
```

Output:-



```
Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd Desktop
C:\Users\Admin\Desktop>javac UseInterface.java
C:\Users\Admin\Desktop>java UseInterface
ti be,or not to be
Shakespear
C:\Users\Admin\Desktop>
```

10.3 Code with Error

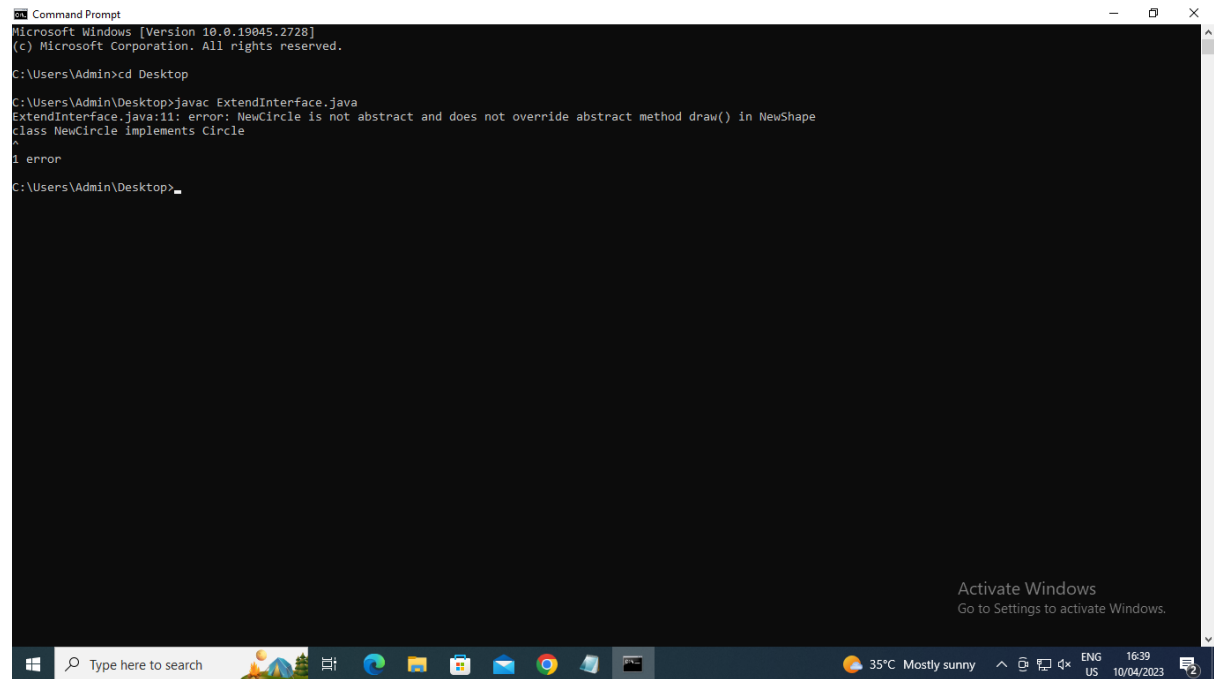
```
interface NewShape
{
    void draw();
}
interface Circle extends NewShape
{
    void getRadius ();
    int radius=10;
}
class NewCircle implements Circle
{
    public void getRadius()
    {
        System.out.println(radius);
    }
}
class ExtendInterface extends NewCircle
{
    public static void main(String[] args)
```

```

{
    Circle nc = new NewCircle();
    nc.getRadius();
}
}

```

Output:-



```

Command Prompt
Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd Desktop

C:\Users\Admin\Desktop>javac ExtendInterface.java
ExtendInterface.java:11: error: NewCircle is not abstract and does not override abstract method draw() in NewShape
class NewCircle implements Circle
^
1 error

C:\Users\Admin\Desktop>

```

Activate Windows
Go to Settings to activate Windows.

10.3 Code without Error

```

interface newshape
{
    void draw();
}
interface circle extends newshape
{
    void getradius();
    int radius=10;
}
class newcircle implements circle
{
    public void draw()
    {
        System.out.println("new method");
    }
    public void getradius()

```

```

    {
        System.out.println(radius);
    }
}
class ExtendInterface extends newcircle
{
    public static void main(String[] args)
    {
        circle nc=new newcircle();
        nc.getradius();
    }
}

```

Output:-

```

Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd Desktop
C:\Users\Admin\Desktop>javac ExtendInterface.java
C:\Users\Admin\Desktop>java ExtendInterface
10
C:\Users\Admin\Desktop>

```

10.5 Code with Error

```

interface NewShape
{
    void draw();
    int radius = 10;
}
class NewCircle implements NewShape
{
    public void draw()

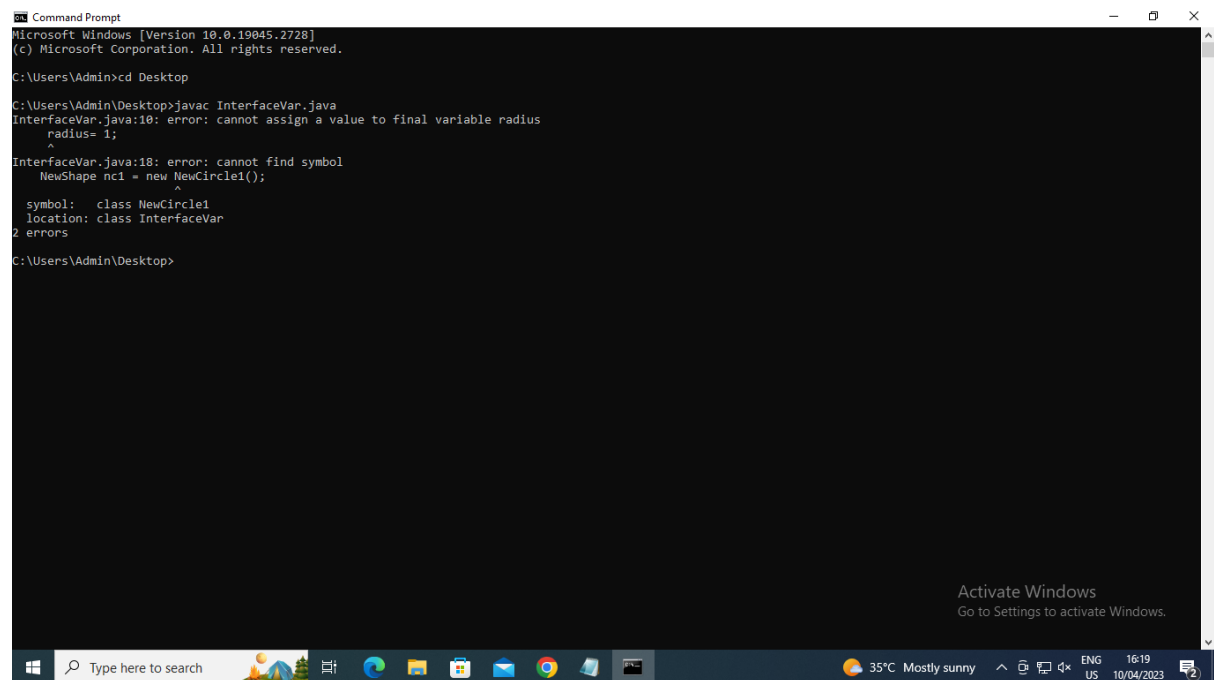
```

```

    {
        radius= 1;
        System.out.println ("Radius is: "+radius);
    }
}
class InterfaceVar
{
    public static void main (String[] args)
    {
        NewShape nc1 = new NewCircle1();
        nc1.draw();
    }
}

```

Output:-



```

Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd Desktop

C:\Users\Admin\Desktop>javac InterfaceVar.java
InterfaceVar.java:10: error: cannot assign a value to final variable radius
    radius= 1;
    ^
InterfaceVar.java:18: error: cannot find symbol
    NewShape nc1 = new NewCircle1();
                      ^
  symbol:   class NewCircle1
  location: class InterfaceVar
2 errors

C:\Users\Admin\Desktop>

```

Activate Windows
Go to Settings to activate Windows.

10.5 Code without Error

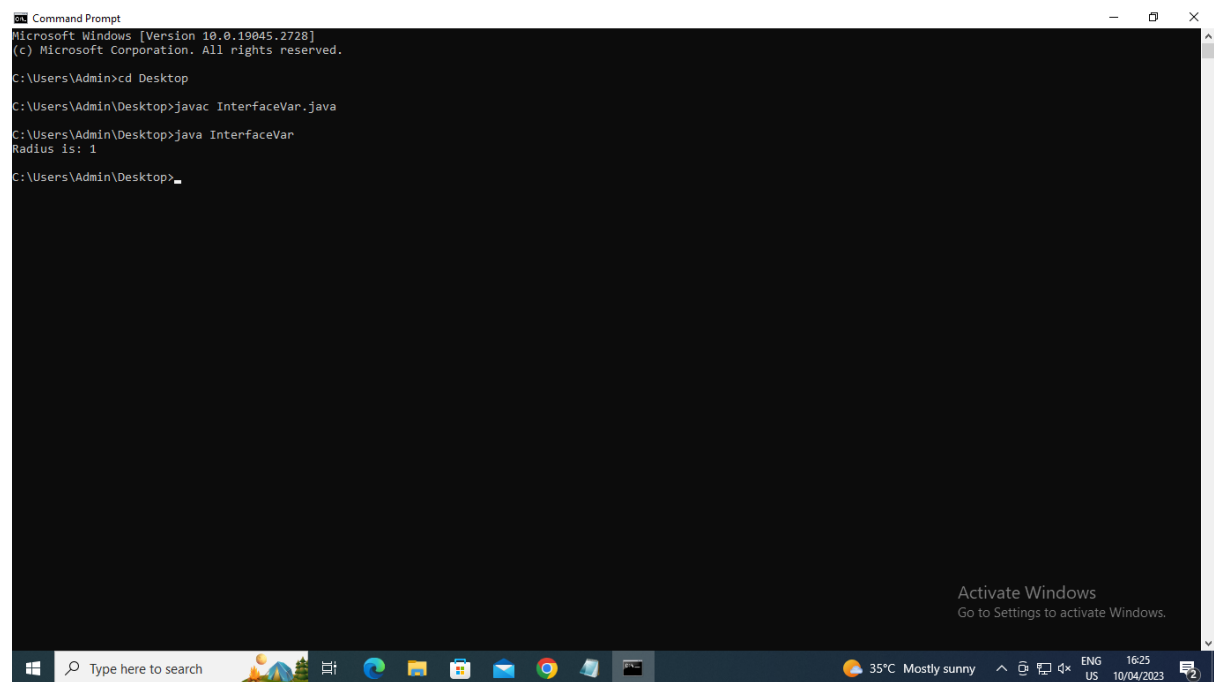
```

interface NewShape
{
    void draw();
}
class NewCircle implements NewShape
{
    public void draw()

```

```
{
    int radius= 1;
    System.out.println ("Radius is: "+radius);
}
}
class InterfaceVar
{
    public static void main (String[] args)
    {
        NewShape nc1 = new NewCircle();
        nc1.draw();
    }
}
```

Output:-



```
Command Prompt
Microsoft Windows [Version 10.0.19045.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd Desktop
C:\Users\Admin\Desktop>javac InterfaceVar.java
C:\Users\Admin\Desktop>java InterfaceVar
Radius is: 1
C:\Users\Admin\Desktop>
```

Activate Windows
Go to Settings to activate Windows.

Type here to search

35°C Mostly sunny

ENG US 16:25 10/04/2023

Interface Code:-

```
class student
{
    int rollNumber;
    void getNumber(int n)
    {
        rollNumber=n;
    }
    void putNumber()
    {
        System.out.println("Roll No :"+rollNumber);
    }
}
class test extends student
{
    double part1,part2;
    void getMarks (double m1,int m2)
    {
        part1=m1;
        part2=m2;
    }
    void putMarks()
    {
        System.out.println("Marks obtained");
        System.out.println("part 1=" +part1);
        System.out.println("part 2=" +part2);
    }
}
interface sports
{
    int sportwt=6;
    void putwt();
}
class results extends test implements sports
{
    double total;
    public void putwt()
    {
        System.out.println("Sports wt="+sportwt);
    }
    void display()
    {
        total=part1+part2+sportwt;
        putNumber();
        putMarks();
        putwt();
        System.out.println("Total score="+total);
    }
}
```

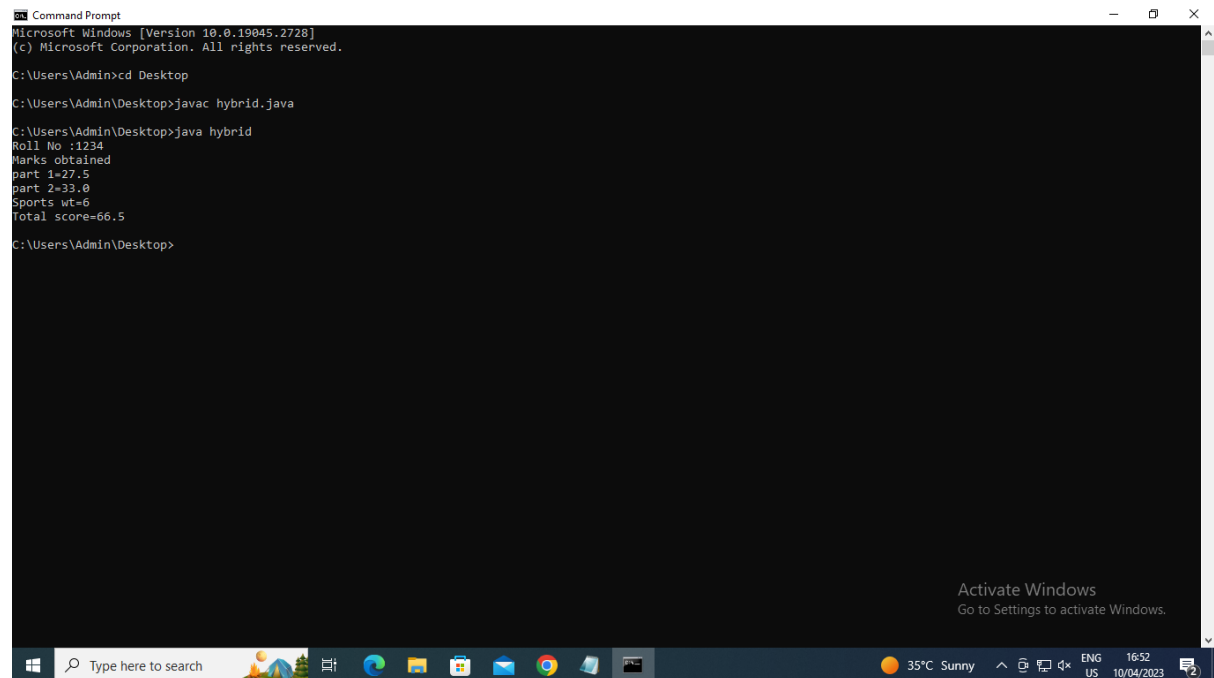


```

    }
}
class hybrid
{
    public static void main(String args[])
    {
        results student1=new results();
        student1.getNumber(1234);
        student1.getMarks(27.5,33);
        student1.display();
    }
}

```

Output:-



```

C:\Users\Admin>cd Desktop
C:\Users\Admin\Desktop>javac hybrid.java
C:\Users\Admin\Desktop>java hybrid
Roll No :1234
Marks obtained
part 1=27.5
part 2=33.0
Sports wt=6
Total score=66.5
C:\Users\Admin\Desktop>

```

Daigram Code:-

```

interface emp_info
{
    void salary();
    int basic_salary=50000;
}
class employee implements emp_info
{

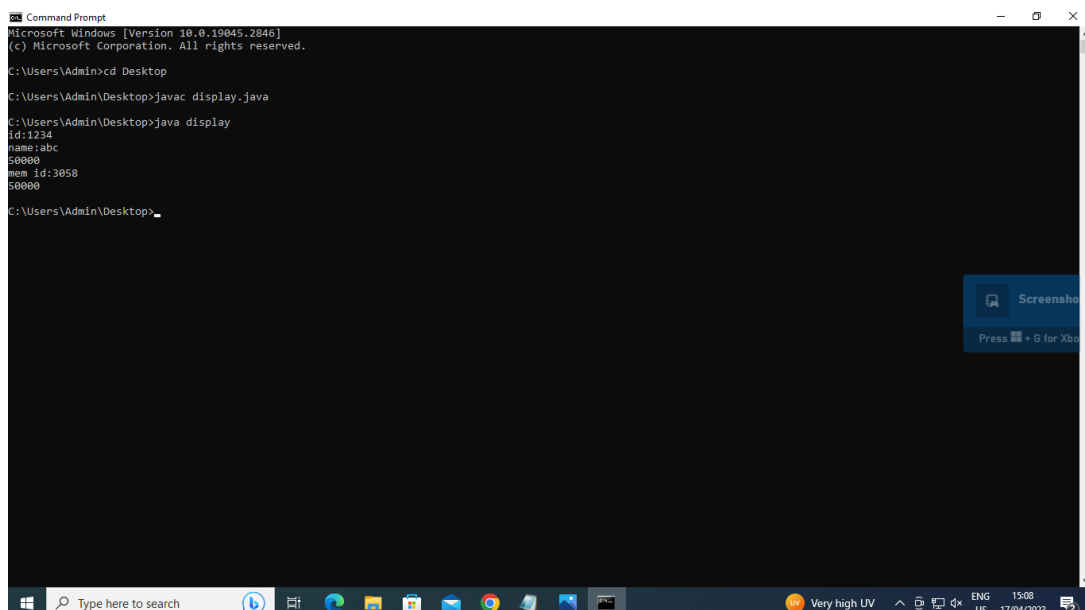
```

```

    public void salary()
    {
        Integer emp_id=1234;
        String emp_name=("abc");
        System.out.println("id:"+emp_id);
        System.out.println("name:"+emp_name);
        System.out.println(basic_salary);
    }
}
class emp_union implements emp_info
{
    public void salary()
    {
        int member_id=3058;
        System.out.println("mem id:"+member_id);
        System.out.println(basic_salary);
    }
}
class display
{
    public static void main(String[] args)
    {
        emp_info a=new employee();
        a.salary();
        emp_info b=new emp_union();
        b.salary();
    }
}

```

Output:-



```

Microsoft Windows [Version 10.0.19045.2846]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd Desktop
C:\Users\Admin\Desktop>javac display.java
C:\Users\Admin\Desktop>java display
id:1234
name:abc
50000
mem id:3058
50000
C:\Users\Admin\Desktop>

```