BCSE-506L (Performance Analysis of Programming Languages Lab)

EXPERIMENT NO. 6

AIM: WAP to implement the concept of Interface and Abstract class.

ALGORITHM: INTERFACE

STEP 1: Start the program.
STEP 2: Create the class Interface having all abstract methods.
STEP 2: Create another class Test that implements the interface and also define all abstract methods of interface class
STEP 3: In the main (), declare the objects of Test class implementing interface.
STEP 4: Access the abstract method of Interface class using object of Test class.
Step 5: Exit

INPUT :

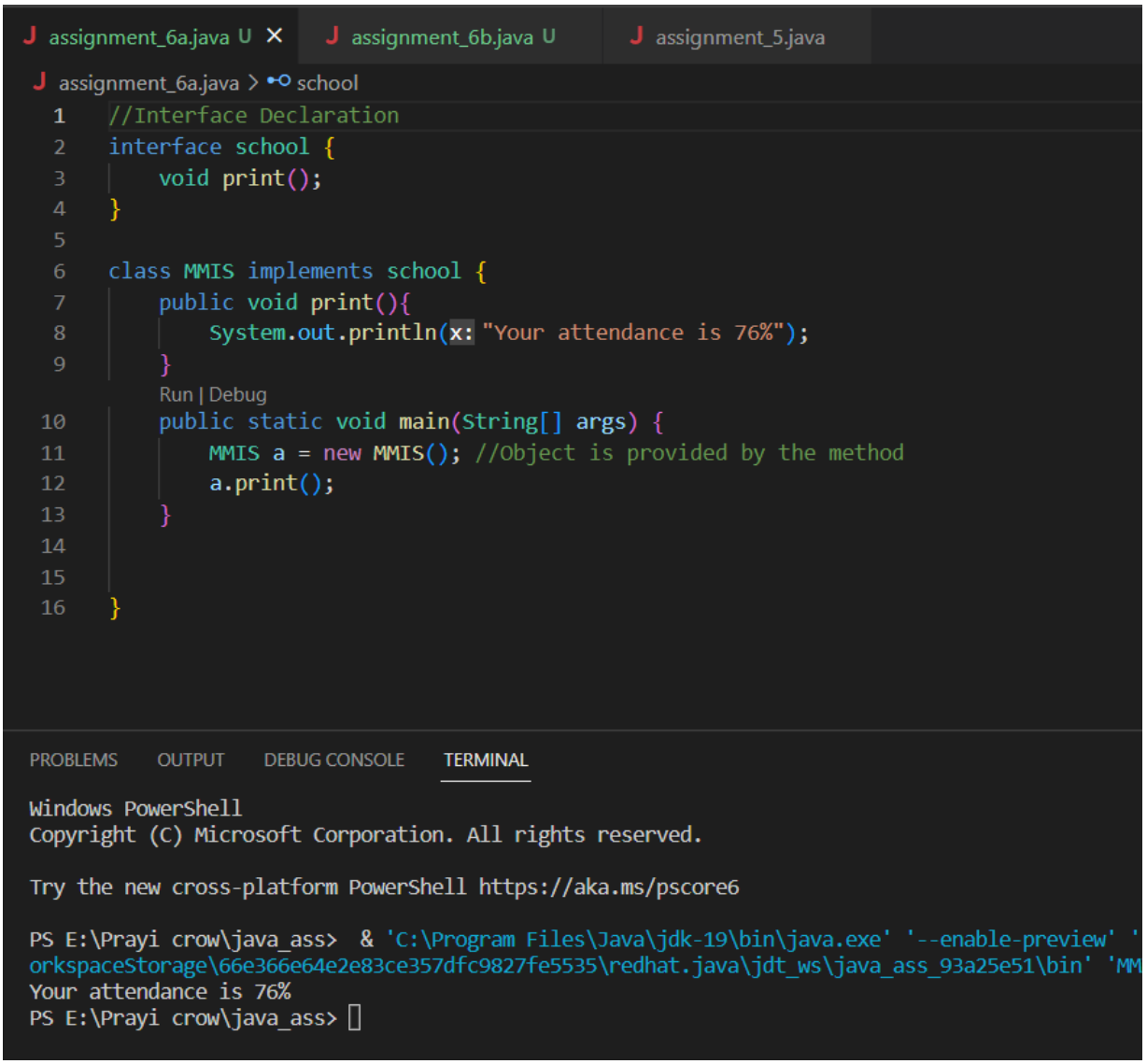
```
//Interface Declaration
interface school {
    void print();
}

class MMIS implements school {
    public void print(){
        System.out.println("Your attendance is 76%");
    }
    public static void main(String[] args) {
        MMIS a = new MMIS(); //Object is provided by the method
        a.print();
  }
}
```

Output :

```
PS E:\Prayi crow\java_ass>  & 'C:\Program
Files\Java\jdk-19\bin\java.exe' '--enable-preview'
'-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
'C:\Users\bajra\AppData\Roaming\Code\User\workspaceStorage\66e36
6e64e2e83ce357dfc9827fe5535\redhat.java\jdt_ws\java_ass_93a25e51
\bin' 'MMIS'
Your attendance is 76%.
```

IDE:


assignment_6a.java U
assignment_6b.java U
assignment_5.java
assignment_6a.java > school
1 //Interface Declaration
2 interface school {
3 void print();
4 }
5
6 class MMIS implements school {
7 public void print(){
8 System.out.println(x: "Your attendance is 76%");
9 }
Run | Debug
10 public static void main(String[] args) {
11 MMIS a = new MMIS(); //Object is provided by the method
12 a.print();
13 }
14
15
16 }
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS E:\Prayi crow\java_ass> & 'C:\Program Files\Java\jdk-19\bin\java.exe' '--enable-preview' '
orkspaceStorage\66e366e64e2e83ce357dfc9827fe5535\redhat.java\jdt_ws\java_ass_93a25e51\bin' 'MM
Your attendance is 76%
PS E:\Prayi crow\java_ass>

ALGORITHM: ABSTRACT CLASS

STEP 1: Start the program.
STEP 2: Create the abstract class Abstract having all abstract and non-abstract methods.
STEP 2: Create another class Test that inherit the abstract class and also define all abstract methods of Abstract class
STEP 3: In the main (), declare the objects of Test.
STEP 4: Access the abstract and non-abstract method of Interface class using object of Test class.
Step 5: Exit

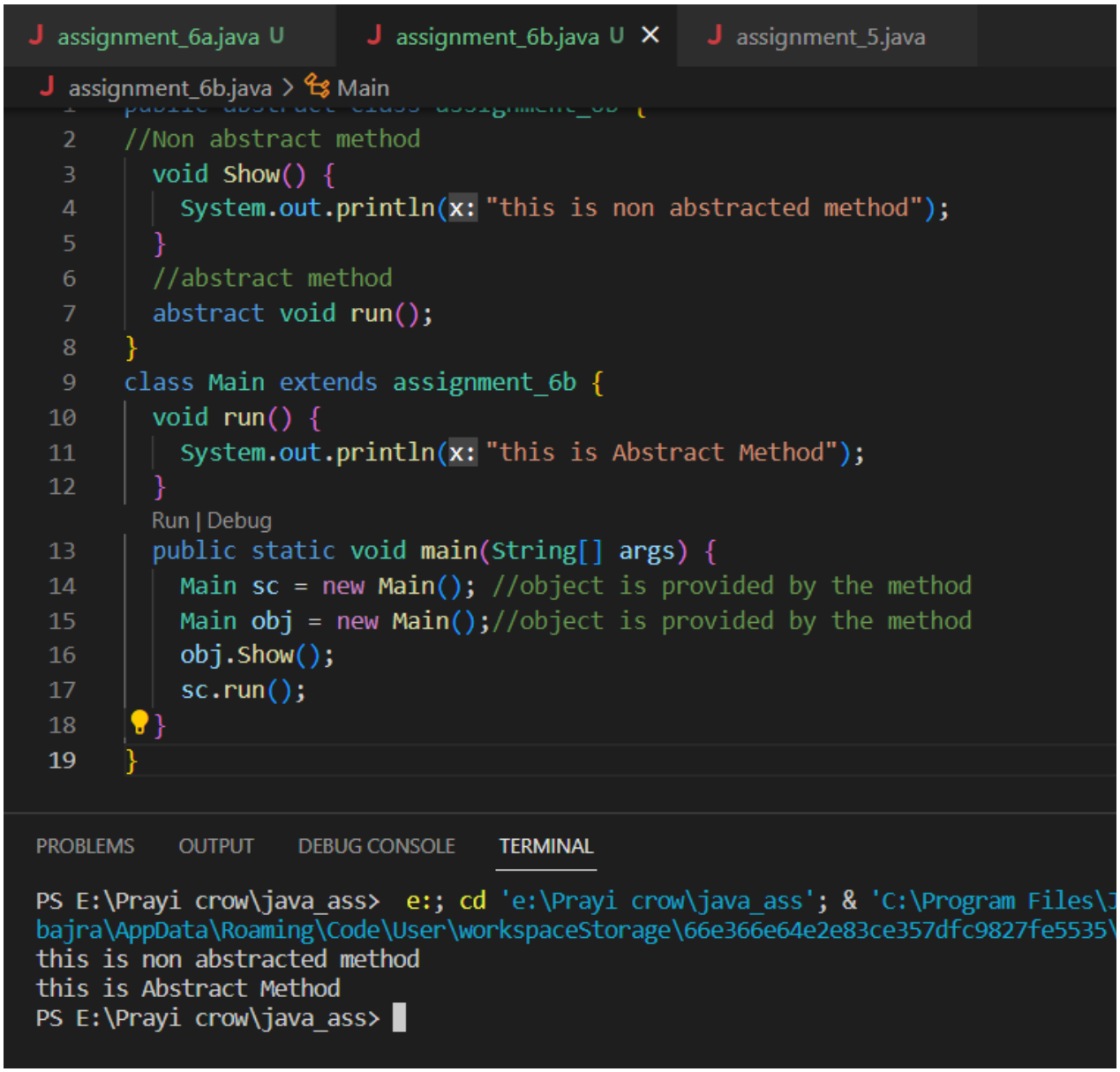
INPUT :

```
public abstract class assignment_6b {
//Non abstract method
  void Show() {
    System.out.println("this is non abstracted method");
  }
  //abstract method
  abstract void run();
}
class Main extends assignment_6b {
  void run() {
    System.out.println("this is Abstract Method");
  }
  public static void main(String[] args) {
    Main sc = new Main(); //object is provided by the method
    Main obj = new Main();//object is provided by the method
    obj.Show();
    sc.run();
  }
}
```

Output :

```
PS E:\Prayi crow\java_ass>  e:; cd 'e:\Prayi crow\java_ass'; &
'C:\Program Files\Java\jdk-19\bin\java.exe' '--enable-preview'
'-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
'C:\Users\bajra\AppData\Roaming\Code\User\workspaceStorage\66e36
6e64e2e83ce357dfc9827fe5535\redhat.java\jdt_ws\java_ass_93a25e51
\bin' 'Main'
this is non abstracted method
this is Abstract Method
```

IDE:



Bajrang
Roll No. 11212766
Section B3