Day1_Class,method,object

DAY1:

1.Java Introduction
2.Class,Method,Object
3.Same package and Different package
4.Encapsulation

QUESTIONS(Theory):

1.What is platform independent?
2.What is open source?
3.Difference between JDK,JRE,JVM?
4.why we go for java?
5.What is the latest version of JDK and which version you are using in your project?
6.What is the latest version of eclispe and which version you are using in your project?
7.Difference between c++ and java?
8.Features of java?
9.What type of tool you are using in your project to execute java?
10.Difference between class,method,object?
11.Where object stores?
12.How to access one class method in to another package in different package?
13.What is encapsulation?
14.Coding standard to create project,class,method,package and object ?
15.What gives Java it's "write once and run anywhere" nature?

QUESTIONS(Programs):

QUESTION 1:

Project :EmployeeDetails
Package :org.emp
Class :Employee
Methods :empId(),empName(),empDob(),empPhone(),empEmail(),empAddress()

Description:
Create an object for employee class and call above methods also follow the all coding standards.

QUESTION 2:

Project :GreensAddress
Package :org.add
Class :GreensTech
Methods :greensOmr(),greensAdayar(),greensTambaram(),greensVelacherry(),greensAnnaNagar()

Description:
Create an object for GreensTech class and call above methods also follow the all coding standards.

QUESTION 3:

Project :CompanyDetails
Package :org.company
Class :CompanyInfo
Methods :companyName(),companyId(),companyAddress()

Description:
Create an object for CompanyDetails class and call above methods also follow the all coding standards.

QUESTION 4:

Project :MyPhone
Package :org.phone
Class :PhoneInfo
Methods :phoneName(),phoneMieiNum(),Camera(),storage(),osName()

Description:
Create an object for PhoneInfo class and call above methods also follow the all coding standards.

QUESTION 5:

Project :LanguageDetails
Package :org.lang

Class :LanguageInfo
Methods :tamilLanguage(),englishLanguage(),hindiLanguage()

Class :StateDetails
Methods :southIndia(),northIndia()

Description:
Create an object for LanguageInfo and StateDetails inside the StateDetails class and call both classes methods also follow the all coding standards.

QUESTION 6:

Project :EmployeeInformation
Package :org.emp
Class :Employee
Methods :empName()

Package :org.company
Class :Company
Methods :companyName()

Package :org.client
Class :Client
Methods :clientName()

Package :org.project
Class :Project
Methods :projectName()

Description:
Create an object for all 4 classes inside the Employee class and call all classes methods also follow the all coding standards.

QUESTION 7:

Project :PhoneDetails
Package :org.phone
Class :ExternalStorage
Methods :size()

Class :InternalStorage
Methods :processorName(),ramSize()

Description:
Create an object for ExternalStorage and InternalStorage inside the InternalStorage class and call both classes methods also follow the all coding standards.

QUESTION 8:

Project :CollegeInformation
Package :org.college
Class :College
Methods :collegeName(),collegeCode(),collegeRank()

Class :Student
Methods :studentName(),studentDept(),studentId()

Class :Hostel
Methods :hostelName()

```
     Class     :Dept
     Methods   :deptName()

Description:
Create an object for all 4 classes  inside the College class and call all classes methods also
follow the all coding standards.


QUESTION 9:
------------
     Project   :VehicleInformation
     Package   :org.allvehicle
     Class     :Vehicle
     Methods   :VehicleNecessery()

     Package   :org.twowheeler
     Class     :TwoWheller
     Methods   :bike(),cycle()

     Package   :org.threewheeler
     Class     :ThreeWheeler
     Methods   :Auto()

     Package   :org.fourwheeler
     Class     :FourWheeler
     Methods   :car(),bus(),lorry()

Description:
Create an object for all 4 classes  inside the Vehicle class and call all classes methods also
follow the all coding standards.

QUESTION 10:
--------------
     Project   :TransportInformation
     Package   :org.transport
     Class     :Transport
     Methods   :TransportForm

     Package   :org.road
     Class     :Road
     Methods   :bike(),cycle(),bus(),car()

     Package   :org.air
     Class     :Air
     Methods   :aeroPlane(),heliCopter()

     Package   :org.water
     Class     :Water
     Methods   :boat(),ship()

Description:
Create an object for all 4 classes  inside the Transport class and call all classes methods also
follow the all coding standards.


QUESTION 11:
--------------
     Project   :NetworkInformation
     Package   :org.network
     Class     :Wifi
     Methods   :wifiName()

     Class     :MobileData
     Methods   :dataName()

     Class     :Lan
```

Methods :lanName()

Class :Wireless
Methods :modamName()

Description:
Create an object for all 4 classes inside the Wifi class and call all classes methods also follow the all coding standards.

Day2_Inheritance,Scanner,Datatype,access specifier
DAY2:

1.Inheritance
2.Access specifiers
3.Data types
4.Scanner class

QUESTIONS(Theory)

1.What is mean by inheritance?
2.Types of inheritance and explain all types?
3.What is mean by multiple inheritance,why java won't support multiple inheritance?
4.Difference between hybrid and hierachical inheritance?
5.What is the use of access specifier and types?
6.Difference between public and protected?
7.What is mean by Wrapper class?
8.What is default value of String?
9.What is difference between primitive and non primitive datatypes?
10.What is default package in java?
11.What is the super class of all java class?
12.What is use of scanner class?
13.What are the different methods available in Scanner class?
14.Scanner class is under which package?
15.Difference between next() and nextLine()?

QUESTIONS(Programs)

QUESTION 1:

Description: Using Scanner class get the below details
empId
empName
empEmail
empPhoneno
empSalary
empGender
empCity

QUESTION 2:

Description: Using Scanner class get the below details
studentId
studentName
Mark1
Mark2
Mark3
Mark4
Mark5
:Find the total and average of marks

QUESTION 3:
---
package name: org.all
Project name: LanguageDetails
Class name : Languageclass
Methods : alllanguage

package name: org.tamil
Project name: LanguageDetails
Class name : Tamil
Methods : tamillanguage

package name: org.english

```
     Project name: LanguageDetails
     Class name  : English
     Methods     : englishlanguage

     package name: org.telgu
     Project name: LanguageDetails
     Class name  : Telgu
     Methods     : telgulanguage
```

Description:
create above 4 packages and call all your class methods into the Languageclass using multilevel inheritance.

QUESTION 4:

```
     package name: org.india
     Project name: SouthIndia
     Class name  : India
     Methods     : india

     package name: org.tamilnadu
     Project name: SouthIndia
     Class name  : TamiladuN
     Methods     : tamillanguage

     package name: org.kerala
     Project name: SouthIndia
     Class name  : kerala
     Methods     : malayalam

     package name: org.andrapradesh
     Project name: SouthIndia
     Class name  : AndhraPradesh
     Methods     : telugu
```

Description:
create above 4 packages and call all your class methods into the India using multilevel inheritance.

QUESTION 5:

```
     Project   :CollegeInformation
     Package   :org.college
     Class     :College
     Methods   :collegeName(),collegeCode(),collegeRank()

     Class     :Student
     Methods   :studentName(),studentDept(),studentId()

     Class     :Hostel
     Methods   :HostelName()

     Class     :dept
     Methods   :deptName()
```

Description:
create above 4 class and call all your class methods into the Student using multilevel inheritance.

QUESTION 6:

```
     Project   :COmputer
     Class     :Computer
     Methods   :computerModel()

     Class     :Desktop
     Methods   :desktopSize()
```

Description:

create above 2 class and call all your class methods into the Desktop using single inheritance.

QUESTION 7:

Project :LanguageDetails
Package :org.lang
Class :LanguageInfo
Methods :tamilLanguage(),englishLanguage(),hindiLanguage()

Class :StateDetails
Methods :southIndia(),northIndia()

Description:
create above 2 class and call all your class methods into the LanguageInfo using single inheritance.

QUESTION 8:

Description: Using Scanner class get the below details
StudentId
StudentName
StudentEmail
StudentPhoneno
StudentDept
StudentGender
StudentCity

QUESTION 9:

Project :BankDetails
Package :org.bank
Class :BankInfo
Methods :saving(),fixed()

Class :AxisBank
Methods :deposit()

Description:
create above 2 class and call all your class methods into the BankInfo using single inheritance.

QUESTION 10:

Project :CompanyDetails
Package :org.company
Class :Company
Methods :companyName()

Package :org.client
Class :Client
Methods :clientName()

Description:
create above 2 packages and call all your class methods into the Comapany using single inheritance.

QUESTION 11:

Project :EducationInformation
Package :org.edu
Class :Education
Methods :ug(),pg()

Class :Arts
Methods :bsc(),bEd(),bA(),bBA()

Class :Engineering
Methods :bE(),bTech()

Class :Medicine
Methods :physiyo(),dental(),mbbs()

Description:
create above 4 class and call all your class methods into the Education using multilevel inheritance.

Day3_Polymorphism,Abstraction

DAY3:

1.Polymorphism
2.Abstraction

QUESTIONS(Theory)

1.What is mean by polymorphism?
2.Difference between method overloading and method overriding?
3.What is mean by Abstraction?
4.Difference between Abstract class and interface?
5.What is mean by abstract method?
6.Can we create object for abstract class?
7.In interface,can we make method as static?
8.In interface,can we make method as final?
9.How will achieve multiple inheritance in java,write a code for that?

QUESTIONS(Programs)

QUESTION 1:

Find the answer for below questions and tell whether it is possible or not?
I implements I
I implements C
I implements A
I extends I
I extends C
I extends A

C implements I
C implements C
C implements A
C extends I
C extends C
C extends A

A implements I
A implements C
A implements A
A extends I
A extends C
A extends A

A-abstract class
C-class
I- interface

QUESTION 2:

Project :EmployeeDetails
Package :org.emp
Class :Employee
Methods :empId()

Description
You have to overload the method empId() based on different datatype in arguments.

QUESTION 3:

Project :CompanyDetails
Package :org.company
Class :CompanyInfo
Methods :companyName()

Description
You have to overload the method companyName() based on different Number of arguments.

QUESTION 4:

Project :MyPhone
Package :org.phone
Class :Phone
Methods :phoneInfo()

Description
You have to overload the method phoneInfo() based on different datatype order in arguments.

QUESTION 5:

Project :GreensAddress
Package :org.add
Class :GreensTech
Methods :greensOmr()

Description
You have to overload the method greensOmr() based on order,type,number.

QUESTION 6:

Project :BankDetails
Package :org.bank
Class :BankInfo
Methods :saving(),fixed(),deposit()

Class :AxisBank
Methods :deposit()

Description:
You have to override the method deposit in AxisBank.

QUESTION 7:

Project :EducationInformation
Package :org.edu
Class :Education
Methods :ug(),pg()

Class :Arts
Methods :bSc(),bEd(),bA(),bBA(),ug(),pg()

Description:
You have to override the method ug(),pg() in Arts.

QUESTION 8:

Project :UniversityInformation
Package :org.univ
Class :University
Methods :ug(),pg()

Class :College
Methods :ug(),pg()

Description:
ug(),pg() is just a templete in University class and You have to override the method ug(),pg() in College class.

QUESTION 9:

Project :BikeInformation
Package :org.bike

```
     Interface :Bike
     Methods   :cost(),speed()

     Class     :Ktm
     Methods   :cost(),speed()

Description:
cost(),speed() is just a templete in Bike Interface and You have to override the method
cost(),speed() in Ktm class.

QUESTION 10:
-------------
     Project   :Computer
     Interface :HardWare
     Methods   :hardwareResources()

     Interface :Software
     Methods   :softwareResources()

     Class     :Desktop
     Methods   :desktopModel()

Description:
create 2 Interface and archieve multiple inheritance.
```

Day4_ControlStatements

DAY4:

1.if/else if
2.Loopings(for,while,do-while)
3.Switch case
4.break/continue

QUESTIONS(Theory)

1.What is difference between break and continue?
2.Whether we can use continue statement in switch?
3.What is mean by control statments and types?
4.What is mean by for loop?
5.Can you explain about for loop execution process?
6.What is difference between while and do-while?
7.What is the use of default keyword in switch?
8.Difference between for and while loop?

QUESTIONS(Find the output)

QUESTION 1:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 100; i++) {
      if (i == 5) {

      }
      System.out.println(i);

    }

  }

}
```

QUESTION 2:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 10; i++) {
      if (i == 5) {
        break;
      }
      System.out.println(i);

    }

  }

}
```

QUESTION 3:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 10; i++) {
      if (i == 5) {
```

```
        continue;
      }
      System.out.println(i);

    }

  }

}
```

QUESTION 4:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = 1; j <= 3; j++) {
        System.out.println(j);
      }

    }

  }

}
```

QUESTION 5:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = 1; j <= 3; j++) {
        System.out.println(i);
      }

    }

  }

}
```

QUESTION 6:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = 1; j <= i; j++) {
        System.out.println(j);
      }

    }

  }

}
```

QUESTION 7:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
```

```
    for (int i = 1; i <= 3; i++) {
      for (int j = i + 1; j <= 3; j++) {
        System.out.println(j);
      }

    }

  }

}
```

QUESTION 8:

```
------------
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = i + 1; j <= i; j++) {
        System.out.println(j);
      }

    }

  }

}
```

QUESTION 9:

```
------------
package org.test;

public class Hello {
  public static void main(String[] args) {
    int i=5;
      if (i == 5) {
        break;
      }
      System.out.println(i);
  }

}
```

QUESTION 10:

```
------------
package org.test;

public class Hello {
  public static void main(String[] args) {
    int i=5;
      if (i == 5) {
        continue;
      }
      System.out.println(i);
  }

}
```

QUESTIONS(Programs)

```
-------------------
```

QUESTION 1:

```
-----------
```

Description: Write Java program to allow the user to input his/her age.
Then the program will show if the person is eligible to vote.
A person who is eligible to vote must be older than or equal 1 to 18 years old.

Example:

Input = 10
Output = print not eligible.

QUESTION 2:

Description: Write a program to find even or odd number

Example:

Input = 10
Output = Even

QUESTION 3:

Description: Write a program to print even number from 1 to 100

Example:

Output = 2,4,....100

QUESTION 4:

Description: Find the sum of odd number 1 to 100

Example:

Output = 2500

QUESTION 5:

Description: Count of even number 1 to 100

Example:

Output = 50

QUESTION 6:

Description: Write a program to find the factorial of a number.

Example:

Input = 5
Output = 120

QUESTION 7:

Description: Write a program to print the fibonacci series of a number 1 to 100.

Example:

Output = 0,1,1,2,3,5.....

QUESTION 8:

Description: Find prime number or not.

Example:

Input = 11
Output = prime number

QUESTION 9:

Description : Print the below patterns using for loop.

Output:

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7
----
          *
         * *
        * * *
      * * * *
    *   * * * *
----
*
* *
* * *
* * * *
* * * * *

QUESTION 10:

Description: Find Amstrong number or not

Example:

Input = 153
Output = Amstrong number

QUESTION 11:

Description: Reverse the number

Example:

Input = 123
Output = 321

QUESTION 12:

Description: Count of the number

Example:

Input = 123
Output = 3

QUESTION 13:

Description: Sum of the number

Example:

Input = 123
Output = 6

QUESTION 14:

Description: Verify the number is palindrome number not

Example:

```
Input  = 141
Output = Palindrome
```

DAY5:

1.Constructor and types
2.this and super
3.types of var
4.static/final

QUESTIONS(Theory)

1.What is mean by constructor and types?
2.Explain the types of variable
3.Do constructors have any return type?
4.Syntax for creating constructor?
5.What are the rules for defining a constructor?
7.Why a return type is not allowed for constructor?
8.Can we declare constructor as 'private'?
9.Why a compiler given constructor is called as default constructor?
10.What is constructor chaining and how can it be achieved in Java?
11.Can we use this() and super() in a method?
12.What are the common uses of "this" keyword in java?
13.Types of variable?
14.What is meant by local variable,instance variable,class/static variable?
15.What is mean by static keyword in java?
16.Can we override static method in java?
17.Can we overload static method in java?
18.What is mean by static variable?
19.What is mean by static method?
20.What is mean by final keyword and what's happend when we declare final as in class,method,variable?
21.What is difference between final and finalize keyword?
22.Where local,static and class variables stores in jvm?

QUESTIONS(Find the below Output)

QUESTION 1:

```
package org.test;

public class A {
  public A() {
    this("JAVA");
    System.out.println("Default const...");
  }

  public A(int id) {
    this(3456.5678f);
    System.out.println(id);
  }

  public A(String name) {
    this(12);
    System.out.println(name);
  }

  public A(float sal) {
    System.out.println(sal);
  }

  public static void main(String[] args) {
    A a = new A();
  }

}
```

QUESTION 2:

```
package org.test;

public class A extends B{
  public A() {

    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }

}

package org.test;

public class B {
  public B() {
    System.out.println("Super class");
  }

}
```

QUESTION 3:

```
------------
package org.test;

public class A extends B{
  public A() {

    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }

}

package org.test;

public class B {
  public B() {
    System.out.println("Super class");
  }

  public B(int id) {
    System.out.println(id);
  }

}
```

QUESTION 4:

```
------------
package org.test;

public class A extends B {
  public A() {
    super(12);
    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }
```

```
}

package org.test;

public class B {
  public B() {
    System.out.println("Super class");
  }

  public B(int id) {
    System.out.println(id);
  }

}
```

QUESTION 5:

```
-----------
package org.test;

public class B {

  public B(int id) {
    System.out.println(id);
  }

}

package org.test;

public class A extends B {
  public A() {
    super(12);
    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }

}
```

QUESTION 6:

```
------------
package org.test;

public class A extends B {
  public A() {

    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }

}

package org.test;

public class B {

  public B(int id) {
    System.out.println(id);
  }

}
```

DAY6:

1.String functions
2.Literal String
3.Non Literal String
4.Mutable String
5.ImMutable String

QUESTIONS(Theory)

1.What is mean by string?
2.How to find length of the string?
3.How to find partcular character in string?
4.How to split the string?
5.What is difference between literal String and non literal string?
6.What is mutable and immutable string?
7.Difference between stringbulider and stringbuffer?
8.Method name to identify memory location?
9.What are the string functions available in java?
10.What is difference between charAt() and contains() method?
11.What is the return type of compareTo()
12.Where the Literal String and non literal String stores?
13.What is mean by ASCII value?

QUESTIONS(Programs)

QUESTION 1:

Description: Find the length of the below string
   String 1: GreensTechnology
   String 2: SeleniumAutomationtool
   String 3: velmurugan
   String 4: j a v a p r o g r a m
   String 5: 9095484678

QUESTION 2:

Description: Find the particular character index in the given string
   String 1: GreensTechnology
             Find the last index of o

   String 2: SeleniumAutomationtool
             Find the index of o

   String 3: Velmurugan
             Find the index of n

   String 4: j a v a p r o g r a m
             Find the last index of (emptyspace)

   String 5: 9095484678
             Find the index of 8

Question 3:

Description: Find the particular character in the given string
   String 1: GreensTechnology
             print the character h

   String 2: SeleniumAutomationtool
             print the character o

   String 3: velmurugan
             print the character u

```
   String 4: j a v a p r o g r a m
             print the character p

   String 5: 9095484678
             print the character 7

QUESTION 4:
-----------
Description : Get two input from user and check the equality
            : print in the output whether it is Equal or not
Example:
--------
Input :
String 1 : Java
String 2 : Java

Example:
---------
Input :
String 1 : Java
String 2 : java

Example:
---------
Input :
String 1 : Green Technology
String 2 : GreenTechnology.

Example(use equalsIgnoreCase):
---------
Input :
String 1 : Java
String 2 : java

Example(use equalsIgnoreCase) :
---------
Input :
String 1 : Nisha
String 2 : nisha

QUESTION 5:
------------
QUESTION 5.1:
------------
Description: Get the email id from the user and verify '@' is present or not?

Example:
------------
Input  = velmurugank451@gmail.com
Output = valid email id

QUESTION 5.2:
------------
Description:Get the address from the user and verify "pincode" is present or not?

Example:
------------
Input  = 5-35-2a,venkatesh nivas,Aruppukottai
Output = invalid address

QUESTION 5.3:
------------
Description:Get the email from the user and verify '@' is present or not and return true or false?

Example:
------------
```

```
Input  = Nishakerala24@gmail.com
Output = True/False

QUESTION 5.4:
------------
Description:Get the phonenumber from the user and verify any character is present or not .
            If character is present return invalid number

Example:
------------
Input  = 90954a6o78
Output = False

QUESTION 6:
-----------
Description:Get the phonenumber from the user  .
            If phonenumber exceeds greater than 10 then return invalid number

Example:
--------
Input  = 89034256972365
output = invalid
Example 2:
---------
Input  = 9095484678
Output = valid

QUESTION 7:
------------
QUESTION 7.1:
-------------
Description:Given string as "Welcome to java class" and replace java into sql.

Example:
-----------
Input  = Welcome to class java
output = Welcome to class sql

QUESTION 7.2:
-------------
Description:Given string as "Greens Adayar"and replace Adayar into Omr.

Example:
-----------
Input  = Greens Adayar
Output = Greens Omr

QUESTION 7.3:
-------------
Description: Given String as "Welcome to java class" and Replace space into '#'

Example:
----------
input:Welcome to java class
output:Welcome#to#java#class


QUESTION 7.4:
------------
Description:Get the email from the user and verify "gmail" is present or not.
            If present replace that gmail into yahoo

Example:
------------
Input  = Nishakerala24@gmail.com
Output = Nishakerala24@yahoo.com
```

QUESTION 7.5:

Description:Get the address from the user and verify "pincode" is present or not.
If present replce the pincode with empty space

Example:

Input = 5-35-2a,venkatesh nivas,Aruppukottai,pincode-626101
Output = 5-35-2a,venkatesh nivas,Aruppukottai

QUESTION 8

QUESTION 8.1

Description:Get the input from the user and print that word in lowercase

Example:

Input = NISHANTHI
Output = nishanthi

QUESTION 8.2

Description:Get the input from the user and print that word in Uppercase

Example:

Input = nishanthi
Output = NISHANTHI

QUESTION 8.3

Description:Convert all small letter and into capital letter

Example:

Input = WelcomE
Output = wELCOMe

QUESTION 8.4

Description:Find the number of uppercase count and lowercase count in the given String

Example:

Input = WelComeToJava
Output:

UpperCase=4
LowerCase=9

QUESTION 9

QUESTION 9.1

Description: Given String as "Welcome to java class" and verify whether the given string startsWith welcome

Example:

Input = Welcome to class java
output = True

QUESTION 9.2

Description: Given String as "Hai i am nisha" and verify whether the given string startsWith welcome

Example:

Input = Hai i am nisha
output = False

QUESTION 9.3

Description: Given String as "Welcome to java class" and verify whether the given string endsWith class

Example:

Input = Welcome to java class
output = True

QUESTION 9.4

Description: Given String as "Welcome to java class" and verify whether the given string endsWith java

Example:

Input = Welcome to java class
output = False

QUESTION 9.5

Description: Given String as "Welcome to java class" and verify whether the string is empty or not

Example:

Input = Welcome to java class
output = False

QUESTION 9.6

Description: Given String as "" and verify whether the string is empty or not

Example:

Input = ""
Output = False

QUESTION 10

Description : Get two input from user and Compare

Example

String 1 : Nisha
String 2 : nisha

Example

String 1 : Nia
String 2 : nisha

QUESTION 11

QUESTION 11.1

Description : Generate the two literal string and find the identityHashCose()

Example

String 1 : Nisha
String 2 : Nisha

QUESTION 11.2

Description : Generate the two non literal string and find the identityHashCose()

Example

String 1 : Nisha
String 2 : Nisha

QUESTION 11.3

Description : Generate the three non literal string and find the identityHashCose()

Example

String 1 : Nisha
String 2 : Rengan
String 3 : NishaRengan

QUESTION 11.4

Description : Generate the three literal string and find the identityHashCose()

Example

String 1 : Nisha
String 2 : Rengan
String 3 : NishaRengan

QUESTION 12

QUESTION 12.1

Description: Given String as "Welcome to java class" and split it by space.

Example:

Input :Welcome to java class
Output:

Welcome
to
java
class

QUESTION 12.2

Description: Given String as "Welcome to java class" and split it by l

Example:

Input :Welcome to java class
Output:

We
come to java c
ass

QUESTION 13

QUESTION 13.1

Description: Given String as "Welcome to java class" and generate a substring.

Example:

Input :Welcome to java class
Output:

Welcome

QUESTION 14

Example:

Description: Given String as "Welcome" and the number of consonant count and vowels count

Example:

Input = Welcome
output:
----
vowels   = 3
consonant = 4

QUESTION 15:

Description: Find the count of caps,small,number and special character in given string

Example:

Input : Welcome To Java class @123
Output
------
caps count :3
small count :15
number count:3
Special char:5

QUESTION 16

Description: Replace all vowels char into '@'

Example:

Input = Welcome
Output = W@lc@m@

DAY7:

1.Arrays
2.Collections Introduction
3.List(ArrayList only)

QUESTIONS(Theory)

1.What do you mean by an array and How to create an Array?
2.Can you change size of array once created?
3.Is it legal to initialize an array int i[] = {1, 2, 3, 4, 5}
4.Advantages and disadvantages of Array?
5.Can we change the size of an array at run time?
6.Can you declare an array without assigning the size of an array?
7.What is the default value of Array?
8.How to print element of Array?
9.How to compare Two Arrays?
10.How to sort an Array?
11.Can we declare array size as a negative number?
12.Can we add or delete an element after assigning an array?
13.Can we use Generics with the array?
14.What is collection and explain about types?
15.What is the difference between ArrayList and Vector?
16.What is the difference between ArrayList and LinkedList?
17.How to convert Array to List and List to Array

QUESTIONS(Find the below Output)

QUESTION 1:

Description : Write a Java program to sum values of an array
        Input a[]= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
        Output   = 55

QUESTION 2:

Description : Write a Java program to calculate the average value of array elements.
        Input a[]= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
        Output   = average = 7.0

QUESTION 3:

Description : Write a Java program to remove duplicates from array
        Input a[]= {10,10,20,50,60,80,60,50}
        Output a[]= {10,20,50,60,80}

QUESTION 4:

QUESTION 4.1:

Description : Create a new ArrayListlist with values and find the length of it
              Input : List = 10,20,30,90,10,10,40,50

QUESTION 4.2:

Description : Create a new LinkedListlist with values and find the length of it
              Input : List = 100,200,300,400,500,600,700

QUESTION 4.3:

Description : Create a new vector with values and find the length of it
              Input : List = 105,205,305,405,505,605,705,805

QUESTION 4.4:

Description : Create a new LinkedListlist with values and find the size() of it.
Input : List = 100,200,300,400,500,600,700

QUESTION 5:

QUESTION 5.1:

Description : Get the first index value of 10
Input: List = 10,20,30,90

QUESTION 5.2:

Description : Get the last index value of 10
Input: List = 10,20,30,90,10,10,40,50

QUESTION 5.3:

Description : Get the index value of 50
Input: List = 10,20,30,90,10,10,40,50

QUESTION 5.4:

Description : Get the index value of 90
Input: List = 10,20,30,90,10,10,40,50

QUESTION 5.5:

Description : Get the each index value of 10 present in below list
Input: List = 10,20,30,90,10,10,40,50,10

QUESTION 5.6:

Description : Get the each index value of 70 present in below list
Input: List = 10,20,30,90,10,10,40,50,10

QUESTION 6:

QUESTION 6.1:

Description : Get the value present at 2nd index
Input: List = 10,20,30,40,50,60

QUESTION 6.2:

Description : Get the value present at 4th index
Input: List = 100,200,300,400,500,600,700

QUESTION 6.3:

Description : Get the value present at 8th index
Input: List = 105,205,305,405,505,605,705,805

QUESTION 6.4:

Description : Get the each value of list by using normal for loop
Input: List = 105,205,305,405,505,605,705,805

QUESTION 6.5:

Description : Get the each value of list by using enhanced for loop
Input: List = 105,205,305,405,505,605,705,805

QUESTION 7:

QUESTION 7.1:

Description : Remove the value present at 2nd index

```
              Input:   List = 10,20,30,40,50,60

QUESTION 7.2:
--------------
Description : Remove the value present at 10th index
              Input:   List = 10,20,30,90,10,10,40

QUESTION 7.3:
--------------
Description : Remove the last value of 10 present in the list
              Input:   List = 10,20,30,90,10,10,40

QUESTION 8:
-------------
QUESTION 8.1:
-------------
Description : Add a value 50 in the 2nd index and display the list after adding.
              Input : List = 10,20,30,90,10,10,40,50

QUESTION 8.2:
-------------
Description : Add a value 70 at the end of the list
              Input : List = 10,20,30,90,10,10,40,50

QUESTION 8.3:
-------------
Description : Add a value 80 at the 8th index of list
              Input : List = 10,20,30,90,10,10,40,50

QUESTION 8.4:
-------------
Description : Add a value 100 at the last index of 10 in the list
              Input : List = 10,20,30,90,10,10,40,50

QUESTION 9:
-----------
QUESTION 9.1:
-------------
Description : Replace the value 300 into 350 in the list
              Input : List = 100,200,300,400,500,600,700

QUESTION 9.2:
--------------
Description : Replace the value present in 7th index as 90
              Input:   List = 10,20,30,90,10,10,40,50,10

QUESTION 9.3:
--------------
Description : Replace the 10 into 100 in List
        Input:   List = 10,20,30,90,10,10,40,50,30
        Output:  List = 100,20,30,90,100,100,40,50,30


QUESTION 10:
-------------
QUESTION 10.1:
--------------
Description : Create a new ArrayListlists with values and return the common values
              Input : List = 10,20,30,90,10,10,40,50
              Input : List = 30,40,50,60,80

QUESTION 10.2:
--------------
Description : Create a new ArrayListlists with values and return the common values
              Input : List = 10,20,30,90,10,10,40,50
              Input : List = 10,20,60,50,40,70,80,90
```

QUESTION 10.3:

Description : Create a new ArrayListlists with values and return the common values
Input : List = 10,20,30,40,50,60,70,80
Input : List = 100,200,300,400,500,600,700,8000

QUESTION 11:

QUESTION 11.1:

Description : Create a new ArrayListlists with values and perform removeAll() function
Input : List = 10,20,30,90,10,10,40,50
Input : List = 30,40,50,60,80

QUESTION 11.2:

Description : Create a new ArrayListlists with values and perform removeAll() function
Input : List = 10,20,30,90,10,10,40,50
Input : List = 10,20,60,50,40,70,80,90

QUESTION 11.3:

Description : Create a new ArrayListlists with values and perform removeAll() function
Input : List = 10,20,30,40,50,60,70,80
Input : List = 100,200,300,400,500,600,700,8000

DAY8:

1.Userdefine List
2.LinkedList vs Vector
3.Set and types

Day8:

1.Userdefine List
2.LinkedList vs Vector
3.Set and types

QUESTIONS(Theory)

1.Describe the Collections type hierarchy. What are the main interfaces, and what are the differences between them?
2.Explain about user defined list?
3.Mention what is Linked List?
4.what is difference between set and List?
5.What is the difference between HashSet and TreeSet ?
6.Difference between Enumurator,Iterator and List Iterator
7.How to convert List into Set

QUESTIONS:

QUESTION 1.1:

Description : Create a HashSet for the below values
              Input : List = 10,20,30,40,50,60,70,80,90,10,20

QUESTION 1.2:

Description : Create a LinkedHashSet for the below values
              Input : List = 10,20,30,40,50,60,70,80,90,10,20

QUESTION 1.3:

Description : Create a TreeSet for the below values
              Input : List = 10,20,30,40,50,60,70,80,90,10,20

QUESTION 2:

QUESTION 2.1:

Description : Convert the below list in to set(use addAll()) method
              Input : List = 10,20,30,90,10,10,40,50

QUESTION 2.2:

Description : Convert the below list in to set(use addAll()) method
              Input : List = 105,205,305,405,505,605,705,805,505,605

QUESTION 2.3:

Description : Convert the below list in to set(use addAll()) method
              Input : List = 100,200,300,400,500,600,700,100,300,500

QUESTION 3:

QUESTION 3.1:

Description : Get the each value of set by using enhanced for loop
              Input: List = 105,205,305,405,505,605,705,805

QUESTION 3.2:

```
Description : Create a TreeSet and iterate each value in the set by using enhanced for loop
              Input : List = 10,20,30,40,50,60,70,80,90,10,20


QUESTION 3.3:
-------------
Description : Create a HashSet and iterate each value in the set by using enhanced for loop
              Input : List = 10,20,30,40,50,60,70,80,90,10,20

QUESTION 4:
-------------
QUESTION 4.1:
--------------
Description : Create a new HashSet with values and return the common values
              Input : List = 10,20,30,90,10,10,40,50
              Input : List = 30,40,50,60,80

QUESTION 4.2:
--------------
Description : Create a new LinkedHashSet with values and return the common values
              Input : List = 10,20,30,90,10,10,40,50
              Input : List = 10,20,60,50,40,70,80,90

QUESTION 4.3:
--------------
Description : Create a new TreeSet with values and return the common values
              Input : List = 10,20,30,40,50,60,70,80
              Input : List = 100,200,300,400,500,600,700,8000

QUESTION 5:
-------------
QUESTION 5.1:
--------------
Description : Create a new HashSet with values and perform removeAll() function
              Input : List = 10,20,30,90,10,10,40,50
              Input : List = 30,40,50,60,80

QUESTION 5.2:
--------------
Description : Create a new LinkedHashSet with values and perform removeAll() function
              Input : List = 10,20,30,90,10,10,40,50
              Input : List = 10,20,60,50,40,70,80,90

QUESTION 5.3:
--------------
Description : Create a new TreeSet with values and perform removeAll() function
              Input : List = 10,20,30,40,50,60,70,80
              Input : List = 100,200,300,400,500,600,700,8000

QUESTION 6:
------------
Description: Create a userdefine Set and insert the 10 employee details
       Input : empId,name,phone,address,dob,doj,eMail,gender,Sal
       Output: Using scanner class insert 10 employee details

QUESTION 7:
------------
Description: Create a userdefine Set and insert the 10 Student details
             Key    :  stdId,stdName,stdPhone,address,dOB,eMail,gender
             Give the related values for key for each Student
```

DAY9:

1.Map and types
2.User define Map

QUESTIONS(Theory)

1.What is map?
2.What is difference between Hash Map and Hash Table?
3.What is difference between set and Map?
4.What are the classes implementing List interface?
5.Which all classes implement Set interface ?
6.Explain about user defined Map?
7.How null allows in below maps:
  HashMap   :k?,v?
  LinkedHashMap:k?,v?
  TreeMap   :k?,v?
  HashTable   :k?,v?
8.How to Iterate Map ?
9.Return type entrySet?

QUESTIONS(practical)

QUESTION 1.1:

Description : Create a HashMap with the below key and values
              key   : 10,20,30,40,50,60,10,50,40
              values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

QUESTION 1.2:

Description : Create a LinkedHashMap with the below key and values
              key   : 10,20,30,40,50,60,10,50,40
              Values : 10,20,30,40,50,60,10,50,40

QUESTION 1.3:

Description : Create a TreeHashMap with the below key and values
              key   : !,@,#,$,%,^,&,*,(,
              Values : 10,20,30,40,50,60,10,50,40

QUESTION 1.4:

Description : Create a HashTable with the below key and values
              Key   : vel,Ganesh,Dinesh,Vengat,subash
              Values : Selenium,framework,oracle,corejava,jira

QUESTION 2:

QUESTION 2.1:

Description : Create a HashMap with the below key and values and get(print) the key only in the map.
              key   : 10,20,30,40,50,60,10,50,40
              values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

QUESTION 2.2:

Description : Create a LinkedHashMap with the below key and values and get(print) the key only in
the map.
              key   : 10,20,30,40,50,60,10,50,40
              Values : 10,20,30,40,50,60,10,50,40

QUESTION 2.3:

Description : Create a TreeMap with the below key and values and get(print) the key only in the map.
key : !,@,#,$,%,^,&,*,(,
Values : 10,20,30,40,50,60,10,50,409

QUESTION 2.4:

Description : Create a HashTable with the below key and values and get(print) the key only in the map.
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira

QUESTION 3:

QUESTION 3.1:

Description : Create a HashMap with the below key and values and get(print) the values only in the map.
key : 10,20,30,40,50,60,10,50,40
values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

QUESTION 3.2:

Description : Create a LinkedHashMap with the below key and values and get(print) the values only in the map.
key : 10,20,30,40,50,60,10,50,40
Values : 10,20,30,40,50,60,10,50,40

QUESTION 3.3:

Description : Create a TreeHashMap with the below key and values and get(print) the values only in the map.
key : !,@,#,$,%,^,&,*,(,
Values : 10,20,30,40,50,60,10,50,409

QUESTION 3.4:

Description : Create a HashTable with the below key and values and get(print) the key only in the map.
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira

QUESTION 4:

QUESTION 4.1:

Description : Create a HashMap with the below key and values and iterate it using enhanced for loop.
key : 10,20,30,40,50,60,10,50,40
values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

QUESTION 4.2:

Description : Create a LinkedHashMap with the below key and values and iterate it using enhanced for loop.
key : 10,20,30,40,50,60,10,50,40
Values : 10,20,30,40,50,60,10,50,40

QUESTION 4.3:

Description : Create a TreeHashMap with the below key and values and iterate it using enhanced for loop.
key : !,@,#,$,%,^,&,*,(,
Values : 10,20,30,40,50,60,10,50,409

QUESTION 4.4:

Description : Create a HashTable with the below key and values and iterate it using enhanced for loop.
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira

QUESTION 5:

QUESTION 5.1:

Description : Create a HashMap with the below key and values and iterate it using enhanced for loop and get the key and values combination.
key : 10,20,30,40,50,60,10,50,40
values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

Sample Output:

10
java
20
sql
30
oops
40
sql
Description : like this you have to itearte the map

QUESTION 5.2:

Description : Create a LinkedHashMap with the below key and values and iterate it using enhanced for loop and get the key and values combination..
key : 10,20,30,40,50,60,10,50,40
Values : 10,20,30,40,50,60,10,50,40

QUESTION 5.3:

Description : Create a TreeHashMap with the below key and values and iterate it using enhanced for loop and get the key and values combination.
key : !,@,#,$,%,^,&,*,(,
Values : 10,20,30,40,50,60,10,50,409

QUESTION 5.4:

Description : Create a HashTable with the below key and values and iterate it using enhanced for loop and get the key and values combination.
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira
QUESTION 6:

Description: Create a userdefine Map and insert the 10 employee details
Key : empId,name,phone,address,dob,doj,eMail,gender,Sal
Give the related values for key for each employee
QUESTION 7:

Description: Create a userdefine Map and insert the 10 Student details
Key : stdId,stdName,stdPhone,address,dOB,eMail,gender
Give the related values for key for each Student

DAY10:

1.Exception
2.throw and throws
3.User define exception

QUESTIONS(Theory)

1.what is Exception?
2.Explain about types of Expection?
3.Difference between checked expection and unchecked expection?
4.What are the differences between exception and error?
5.What is the super class for Exception and Error?
6.Exceptions are defined in which java package
7.What is throw keyword in java?
8.Can we have try block without catch block?
9.Can we write multiple catch blocks under single try block?
10.How to write user defined exception or custom exception in java?
11.What are the different ways to print exception message on console?
12.What are the differences between final finally and finalize in java?
13.Can we write return statement in try and catch blocks?
14.Can we write return statement in finally block?
15.What are the differences between throw and throws?
16.What are the Exception Handling Keywords in Java?
17.Explain Java Exception Hierarchy?
18.How to create custom Exception?

QUESTIONS(Programs)

QUESTION 1:

Description : Find the output for the program:

```
        public class Test
        {
          public static void main(String[] args)
            {
          try
          {
                System.out.printf("1");
              int sum = 9 / 0;
                System.out.printf("2");
          }
          catch(ArithmeticException e)
          {
                System.out.printf("3");
          }
           catch(Exception e)
          {
                System.out.printf("4");
          }
          finally
          {
             System.out.printf("5");
          }
             }
         }
```

QUESTION 2:

Description : Find the output for the program:

```
          public class Test
            {
           private void m1()
```

```
            {
         m2();
          System.out.printf("1");
               }
           private void m2()
              {
          m3();
          System.out.printf("2");
               }
         private void m3()
             {
         System.out.printf("3");
        try
        {
              int sum = 4/0;
              System.out.printf("4");
                 }
        catch(ArithmeticException e)
        {
             System.out.printf("5");
        }

             System.out.printf("7");
              }
          public static void main(String[] args)
           {
              Test obj = new Test();
             obj.m1();
             }
              }
```

QUESTION 3:

Description : Find the output for the program:

```
           public class Test
            {
           public static void main(String[] args)
            {
              try
            {
             System.out.printf("1");
              int data = 5 / 0;
            }
            catch(ArithmeticException e)
            {
            System.out.printf("2");
            System.exit(0);
              }
            finally
               {
             System.out.printf("3");
              }
            System.out.printf("4");
              }
               }
```

QUESTION 4:

Description : Find the output for the program:

```
            public class Test
              {
            public static void main(String[] args)
              {
              try
              {
```

```
                System.out.printf("1");
                  int data = 5 / 0;
                  }
                catch(ArithmeticException e)
                  {
                Throwable obj = new Throwable("Sample");
                 try
                  {
                 throw obj;
                   }
                 catch (Throwable e1)
                    {
                 System.out.printf("8");
                    }
                    }
               finally
                {
               System.out.printf("3");
                }
            System.out.printf("4");
               }
                }
```

QUESTION 5:

Description : Find the output for the program:

```
               import java.io.EOFException;
               import java.io.IOException;

                public class Test
                {
               public static void main(String[] args)
               {
               try
               {
               System.out.printf("1");
               int value = 10 / 0;
                throw new IOException();
                }
               catch(EOFException e)
                  {
             System.out.printf("2");
               }
            catch(ArithmeticException e)
              {
             System.out.printf("3");
                }
            catch(NullPointerException e)
             {
             System.out.printf("4");
             }
            catch(IOException e)
              {
             System.out.printf("5");
             }
                catch(Exception e)
            {
             System.out.printf("6");
               }
               }
               }
```

DAY11:

1.JDBC Connection
2.Mysql Introduction

QUESTIONS(Theory)

1.What is JDBC?
2.What is JDBC Driver?
3.What are the steps to connect to the database in java?
4.What are the JDBC statements?
5.What is the difference between Statement and PreparedStatement interface?
6.what is schema?
7.what is primary key?
8.what is constraints?

QUESTIONS(Program)

Task 1:
Download Sql work bench and do all setup like creating table,assign primary key and retrieve all the values from Demoqa register table(manually).
Input :FirstName,LastName,MaritalStatus(Single,Married,Divorced),
Hobby(Dance,Reading,Cricket),Country,DateOfBirth,PhoneNumber,
Username,E-mail,AboutYourself,Password,ConfirmPassword
Output:
Table Created.
Inserted value in table
Deleted value in table
Modified value in table

Task 2:
Download Sql work bench and do all setup like creating table,assign primary key and retrieve all the values from employee table.
Input :EmpId(PrimaryKey),EmpName,EmpSalary,EmpAddress
Output:Print all the values from the table

Sample Output:

EmpId	EmpName	EmpSalary	EmpAddress
101	Arun	2000	Chennai
102	Siva	3000	Madurai
103	Varun	2000	Selam
104	Tharun	1000	Erode

File Operation:

QUESTIONS(Theory)

1.What is mean by File? In which package it is available? 2.What are the methods available in File ?
3.While creating a file if you not mention the format then under which format it will save the file?
4.What are the difference between append and updating the file?
5.How to compare paths of two files?
6.How to create a new file?
7.How to get last modification date of a file?
8.How to create a file in a specified directory?
9.How to check a file exist or not?
10.How to make a file read-only?
11.How to renaming a file ?
12.How to get a file's size in bytes?
13.How to change the last modification time of a file ?
14.How to create a temporary file?
15.How to append a string in an existing file?
16.How to copy one file into another file?
17.How to delete a file?
18.How to read a file?
19.How to write to a file?
20.How to create directories recursively?
21.How to delete a directory?
22.How to get the fact that a directory is empty or not?
23.How to get a directory is hidden or not?
24.How to print the directory hierarchy?
25.How to print the last modification time of a directory?
26.How to get the parent directory of a file?
27.How to search all files inside a directory?
28.How to get the size of a directory?
29.How to traverse a directory?
30.How to find current working directory?
31.How to display root directories in the system?
32.How to search for a file in a directory?
33.How to display all the files in a directory?
34.How to display all the directories in a directory?

QUESTIONS(Practical)

QUESTION 1

NOTE: Create a new file
write the file with some 10 lines of text.

QUESTION 2

NOTE: Retrieve the text from the file
Check "java" word contains or not?

QUESTION 3

NOTE: Find the row count from the file.

QUESTION 4

NOTE: Print the last 5 lines from the file.

QUESTION 5

NOTE: Print the odd lines from the file.

QUESTION 6

NOTE: Count the number of dublicate words available in the file.

QUESTION 7

NOTE: Check directory D:\Java is available or not.

QUESTION 8

NOTE: Check directory D:\Java is available or not.
If not create new directory.

QUESTION 9

NOTE: Create sub directory D:\Java\Selenium\Material.

QUESTION 10

NOTE: Print all the available files from an existing folder.

CORE JAVA TASK 2022

CLASS ,METHOD,OBJECT		INHERITANCE,SCANNER,DATA TYPES,ACCES SPECIFIER		POLYMORPHISM, ABSTRACTION		CONTROL STATEMENT		CONTROL STSTEMENT		CONSTRUCTOR TYPES		STRING TYPES		ARRAY,COLLECTION,UST		LIST AND SET		MAP		EXCEPTION		JDBC/Mysql/file Operation		File Operation	
DAY-01	STATUS	DAY-02	STATUS	DAY-03	STATUS	DAY-04	STATUS	DAY-04	STATUS	DAY-05	STATUS	DAY-06	STATUS	DAY-07	STATUS	DAY-08	STATUS	DAY-09	STATUS	DAY-10	STATUS	DAY-11	STATUS	DAY-12	STATUS
1	ok	1	OK	1		1	OK	1	OK	1	OK	1	OK	1	OK	1	OK	1	OK	1	OK	1		1	ok
2	ok	2	OK	2	OK	2	OK	2	OK	2	OK	2	OK	2	OK	2	OK	2	OK	2	OK	2		2	
3	ok	3	ok	3	OK	3	OK	3	OK	3	OK	3	OK	3	OK	3	OK	3	OK	3	OK			3	OK
4	ok	4	ok	4	OK	4	OK	4	OK	4	OK	4	OK	4	OK	4	OK	4	OK	4	OK			4	OK
5	ok	5	ok	5	OK	5	OK	5	OK	5	OK	5	OK	5	5.5/5.6	5	OK	5	OK	5	OK			5	OK
6	ok	6	ok	6	OK	6	OK	6	OK	6	OK	6	OK	6	OK	6		6						6	
7	ok	7	ok	7	OK	7	OK	7	OK			7	OK	7	7.2/7.3 WRONG	7		7						7	OK
8	ok	8	DUPLICATE	8	OK	8	OK	8	OK			8	OK	8	8.4 WRONG									8	OK
9	ok	9	ok	9	OK	9	OK	9	OK			9	OK	9	OK									9	OK
10	ok	10	ok	10	OK	10	OK	10	OK			10	OK	10	OK									10	OK
11	ok	11	ok				OK	11	OK			11	OK	11	OK										
								12	OK			12	OK												
								13	OK			13	OK												
								14	OK			14	OK												
												15	OK												
												16	OK												
	WORD		WORD		WORD						WORD														
TOTAL	11		11		10		10		14		6		16		14		7		7		5		2		10	
	123																								

TOTAL	123
	128
	59
	310

CORE JAVA
SELENIUM
FRAME WORK

PDF CONVERSION
DAY-1
DAY-2
DAY-3
DAY-5
DAY-6
DAY-4

QUESTION 1:

Project :EmployeeDetails
Package :org.emp
Class :Employee
Methods
:empId(),empName(),empDob(),empPhone(),empEmail(),empAddress()

Description:
Create an object for employee class and call above methods also follow the all coding standards.

QUESTION 2:

Project :GreensAddress
Package :org.add
Class :GreensTech
Methods
:greensOmr(),greensAdayar(),greensTambaram(),greensVelacherry(),greensAnnaNagar()

Description:
Create an object for GreensTech class and call above methods also follow the all coding standards.

QUESTION 3:

Project :CompanyDetails
Package :org.company
Class :CompanyInfo
Methods :companyName(),companyId(),companyAddress()

Description:
Create an object for CompanyDetails class and call above methods also follow the all coding standards.

QUESTION 4:

Project :MyPhone
Package :org.phone
Class :PhoneInfo
Methods :phoneName(),phoneMieiNum(),Camera(),storage(),osName()

Description:
Create an object for PhoneInfo class and call above methods also follow the all coding standards.

QUESTION 5:

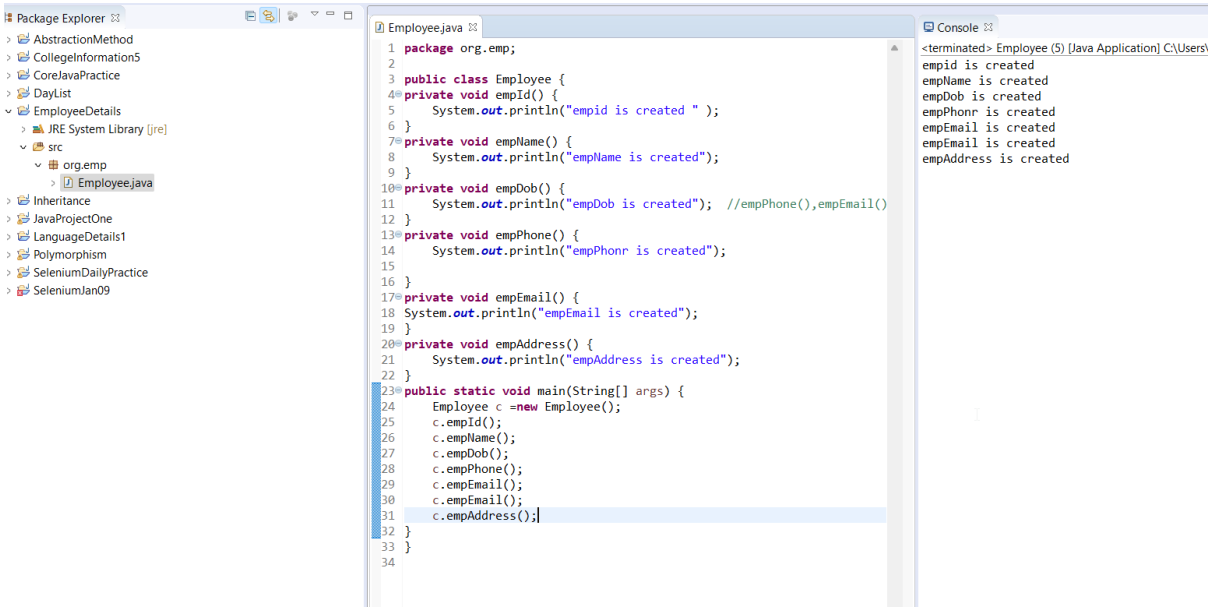
Project :LanguageDetails
Package :org.lang

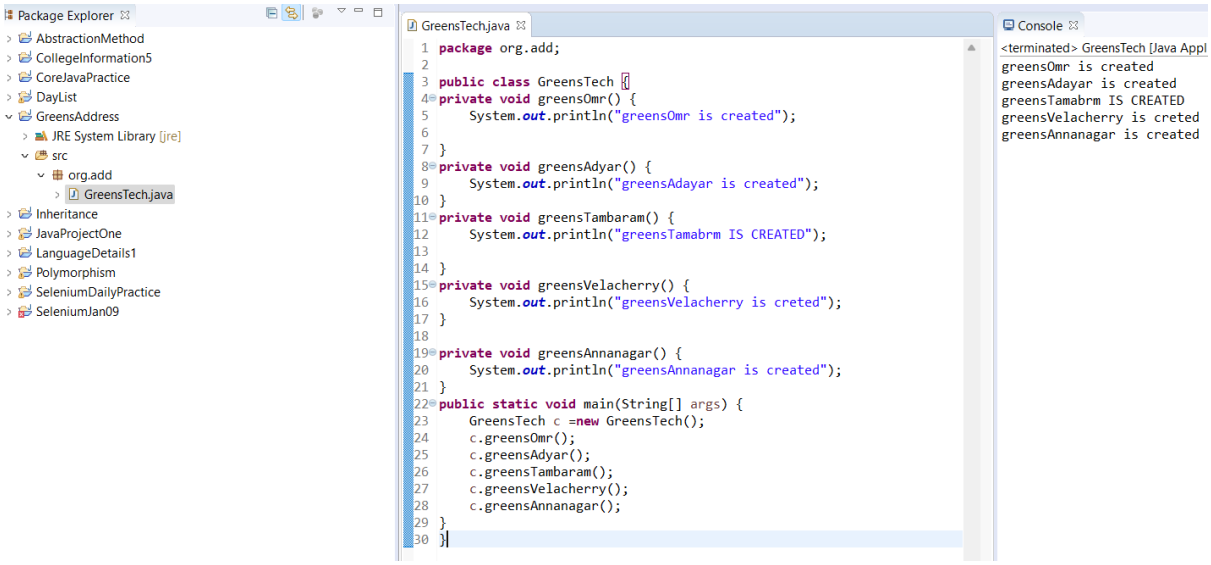
Class :LanguageInfo
Methods :tamilLanguage(),englishLanguage(),hindiLanguage()

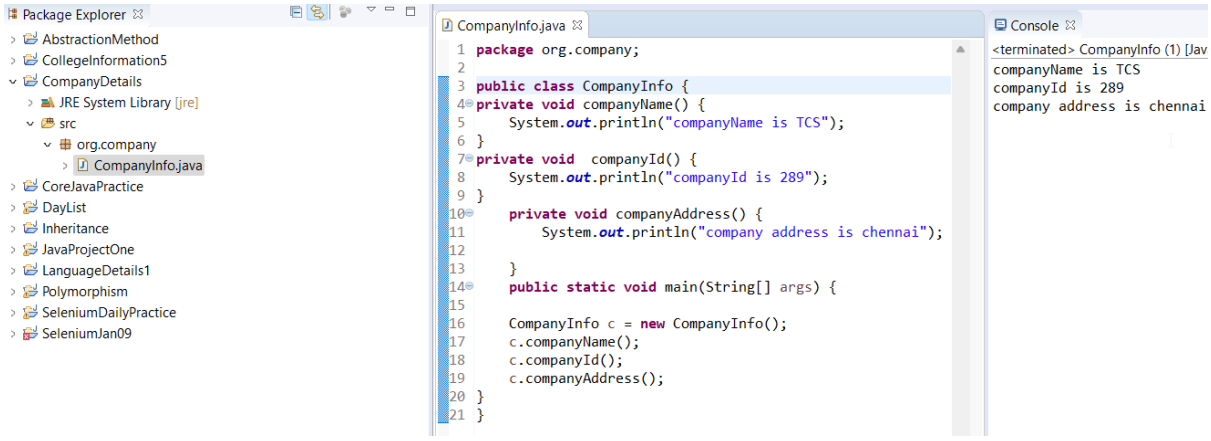
Class :StateDetails
Methods :southIndia(),northIndia()

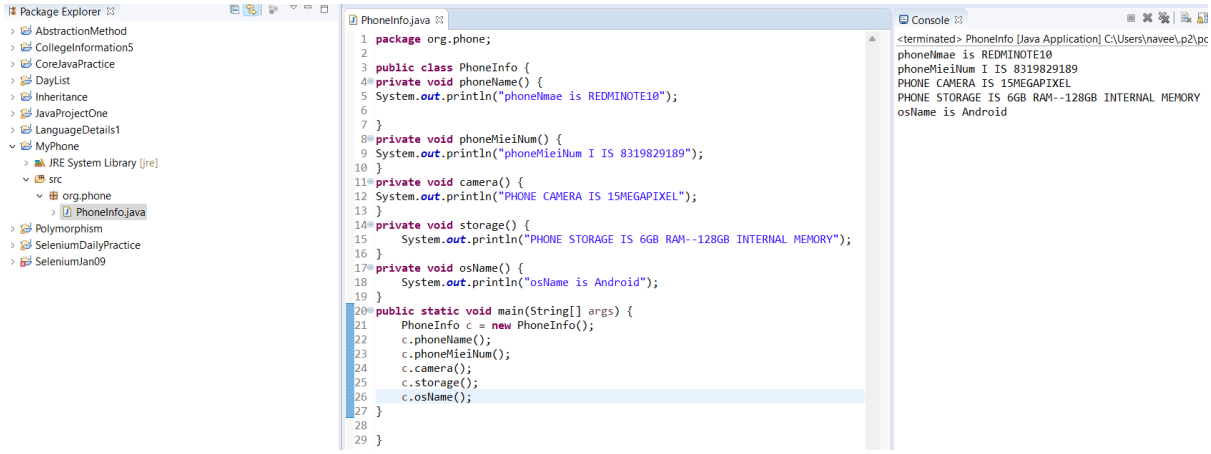
Description:

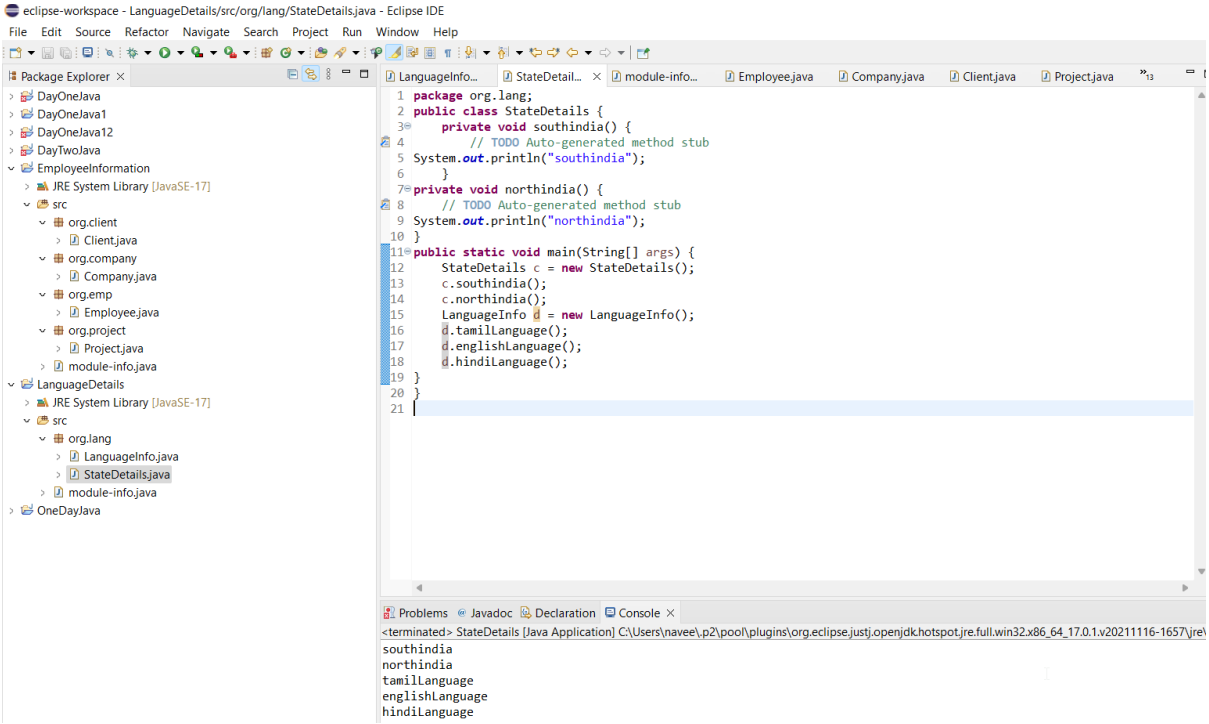
Create an object for LanguageInfo and StateDetails inside the StateDetails class and call both classes methods also follow the all coding standards

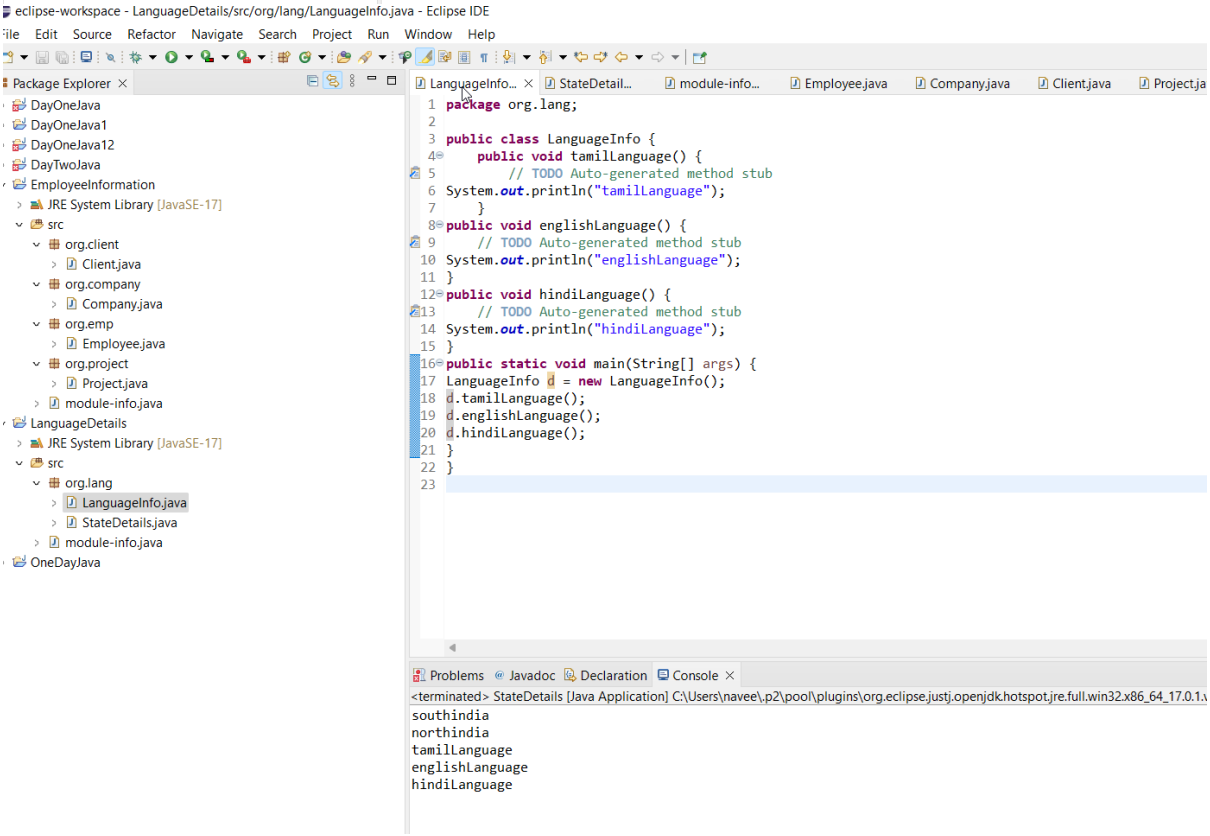





Package Explorer
AbstractionMethod
CollegeInformation5
CompanyDetails
JRE System Library [jre]
src
org.company
CompanyInfo.java
CoreJavaPractice
DayList
Inheritance
JavaProjectOne
LanguageDetails1
Polymorphism
SeleniumDailyPractice
SeleniumJan09
CompanyInfo.java
1 package org.company;
2
3 public class CompanyInfo {
4 private void companyName() {
5 System.out.println("companyName is TCS");
6 }
7 private void companyId() {
8 System.out.println("companyId is 289");
9 }
10 private void companyAddress() {
11 System.out.println("company address is chennai");
12
13 }
14 public static void main(String[] args) {
15
16 CompanyInfo c = new CompanyInfo();
17 c.companyName();
18 c.companyId();
19 c.companyAddress();
20 }
21 }
Console
<terminated> CompanyInfo (1) [Jav
companyName is TCS
companyId is 289
company address is chennai


Package Explorer
AbstractionMethod
CollegeInformation5
CoreJavaPractice
DayList
Inheritance
JavaProjectOne
LanguageDetails1
MyPhone
JRE System Library [jre]
src
org.phone
PhoneInfo.java
Polymorphism
SeleniumDailyPractice
SeleniumJan09
PhoneInfo.java
1 package org.phone;
2
3 public class PhoneInfo {
4 private void phoneName() {
5 System.out.println("phoneNmae is REDMINOTE10");
6
7 }
8 private void phoneMieiNum() {
9 System.out.println("phoneMieiNum I IS 8319829189");
10 }
11 private void camera() {
12 System.out.println("PHONE CAMERA IS 15MEGAPIXEL");
13 }
14 private void storage() {
15 System.out.println("PHONE STORAGE IS 6GB RAM--128GB INTERNAL MEMORY");
16 }
17 private void osName() {
18 System.out.println("osName is Android");
19 }
20 public static void main(String[] args) {
21 PhoneInfo c = new PhoneInfo();
22 c.phoneName();
23 c.phoneMieiNum();
24 c.camera();
25 c.storage();
26 c.osName();
27 }
28
29 }
Console
<terminated> PhoneInfo [Java Application] C:\Users\naveel\.p2\pc
phoneNmae is REDMINOTE10
phoneMieiNum I IS 8319829189
PHONE CAMERA IS 15MEGAPIXEL
PHONE STORAGE IS 6GB RAM--128GB INTERNAL MEMORY
osName is Android





UESTION 6:

```
Project   :EmployeeInformation
Package   :org.emp
Class     :Employee
Methods   :empName()
```

```
     Package   :org.company
     Class     :Company
     Methods   :companyName()

     Package   :org.client
     Class     :Client
     Methods   :clientName()

     Package   :org.project
     Class     :Project
     Methods   :projectName()
```

Description:
Create an object for all 4 classes inside the Employee class and call all classes methods also follow the all coding standards.

QUESTION 7:

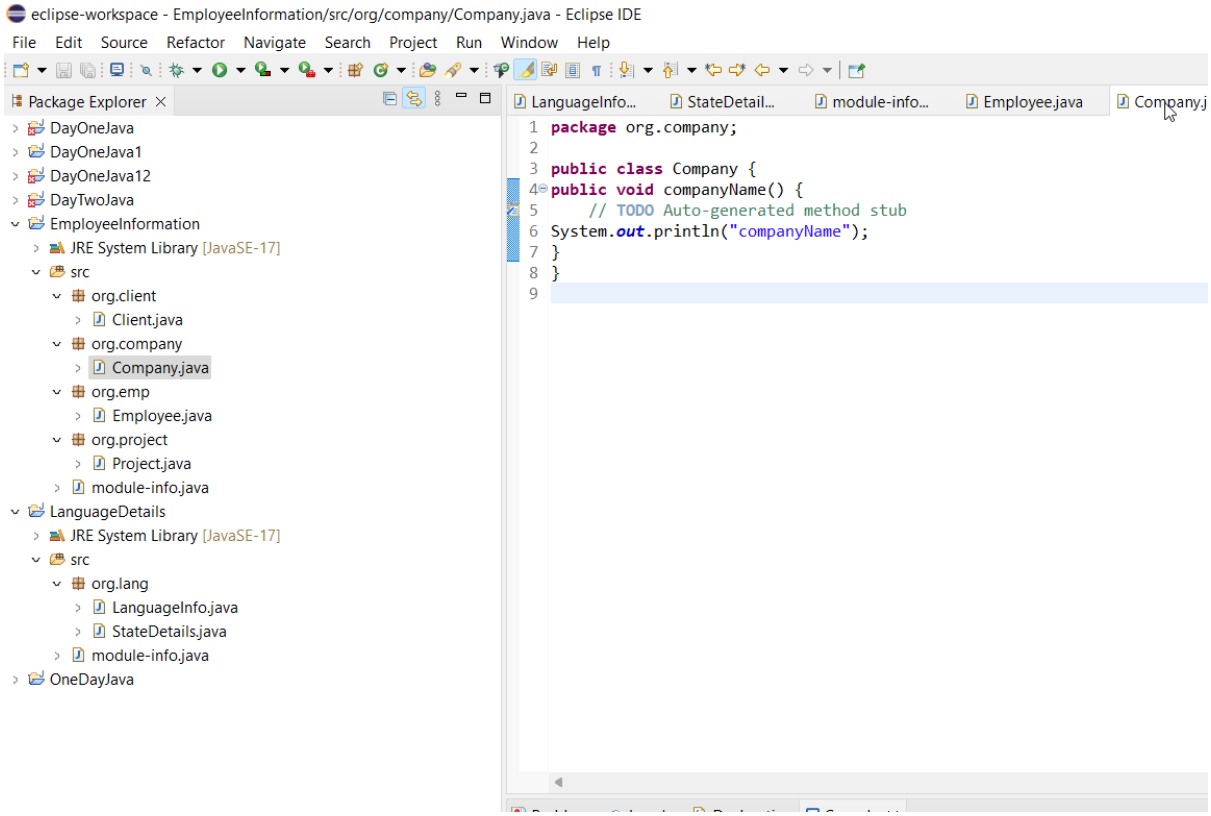
```
     Project   :PhoneDetails
     Package   :org.phone
     Class     :ExternalStorage
     Methods   :size()

     Class     :InternalStorage
     Methods   :processorName(),ramSize()
```

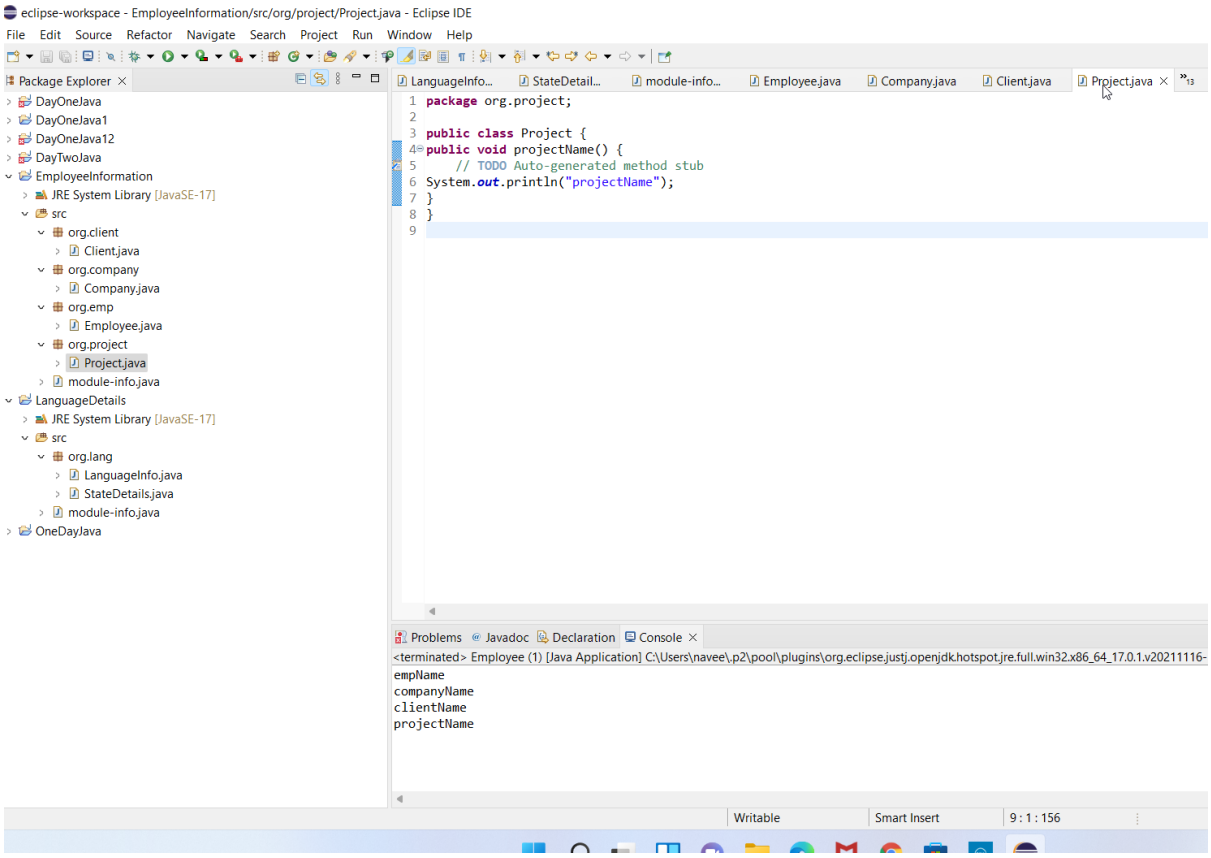
Description:
Create an object for ExternalStorage and InternalStorage inside the InternalStorage class and call both classes methods also follow the all coding standards.

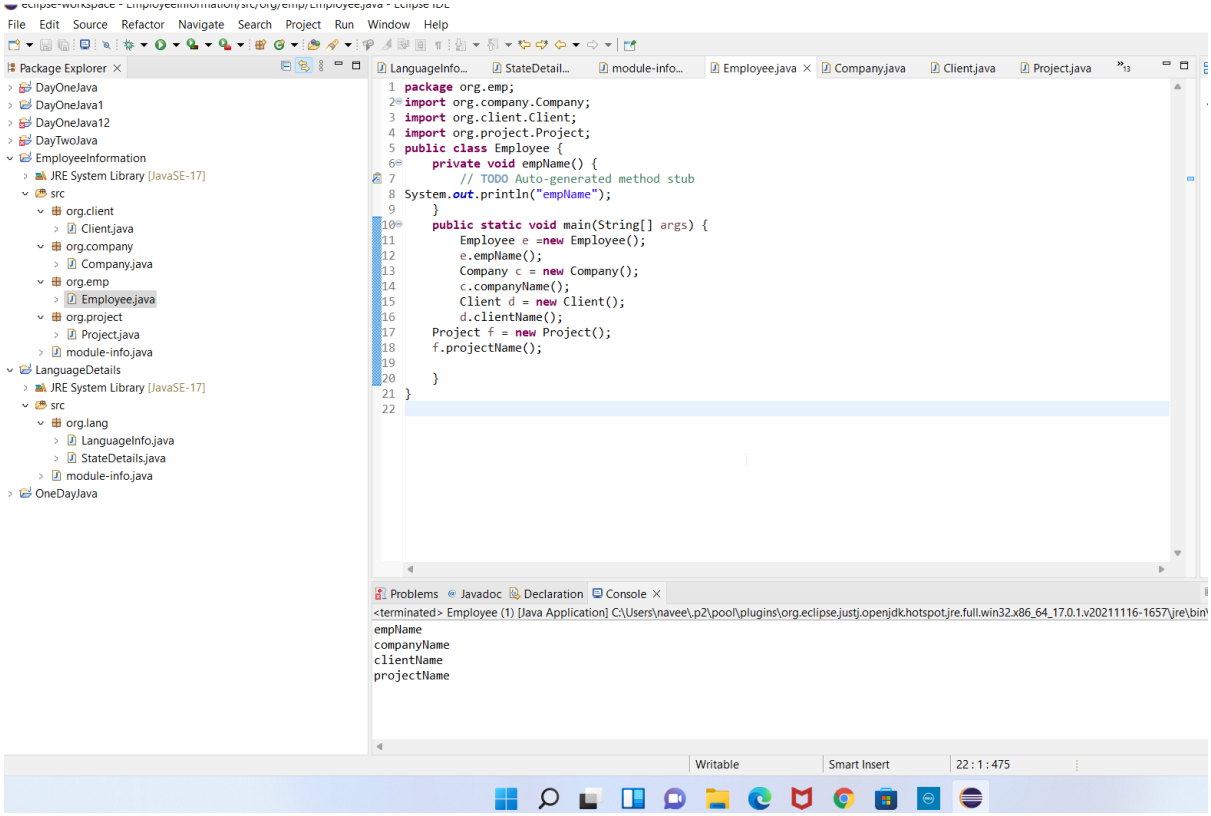
QUESTION 8:

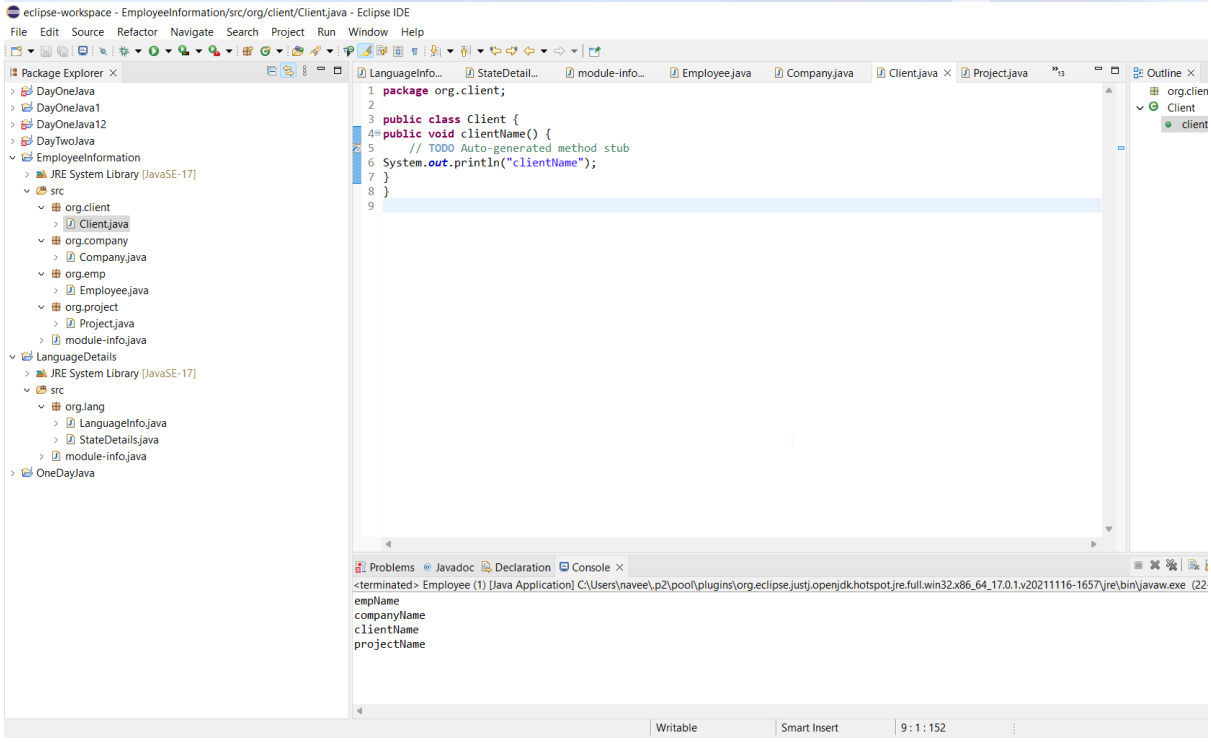
```
     Project   :CollegeInformation
     Package   :org.college
     Class     :College
     Methods   :collegeName(),collegeCode(),collegeRank()

     Class     :Student
     Methods   :studentName(),studentDept(),studentId()

     Class     :Hostel
     Methods   :hostelName()

     Class     :Dept
     Methods   :deptName()
```

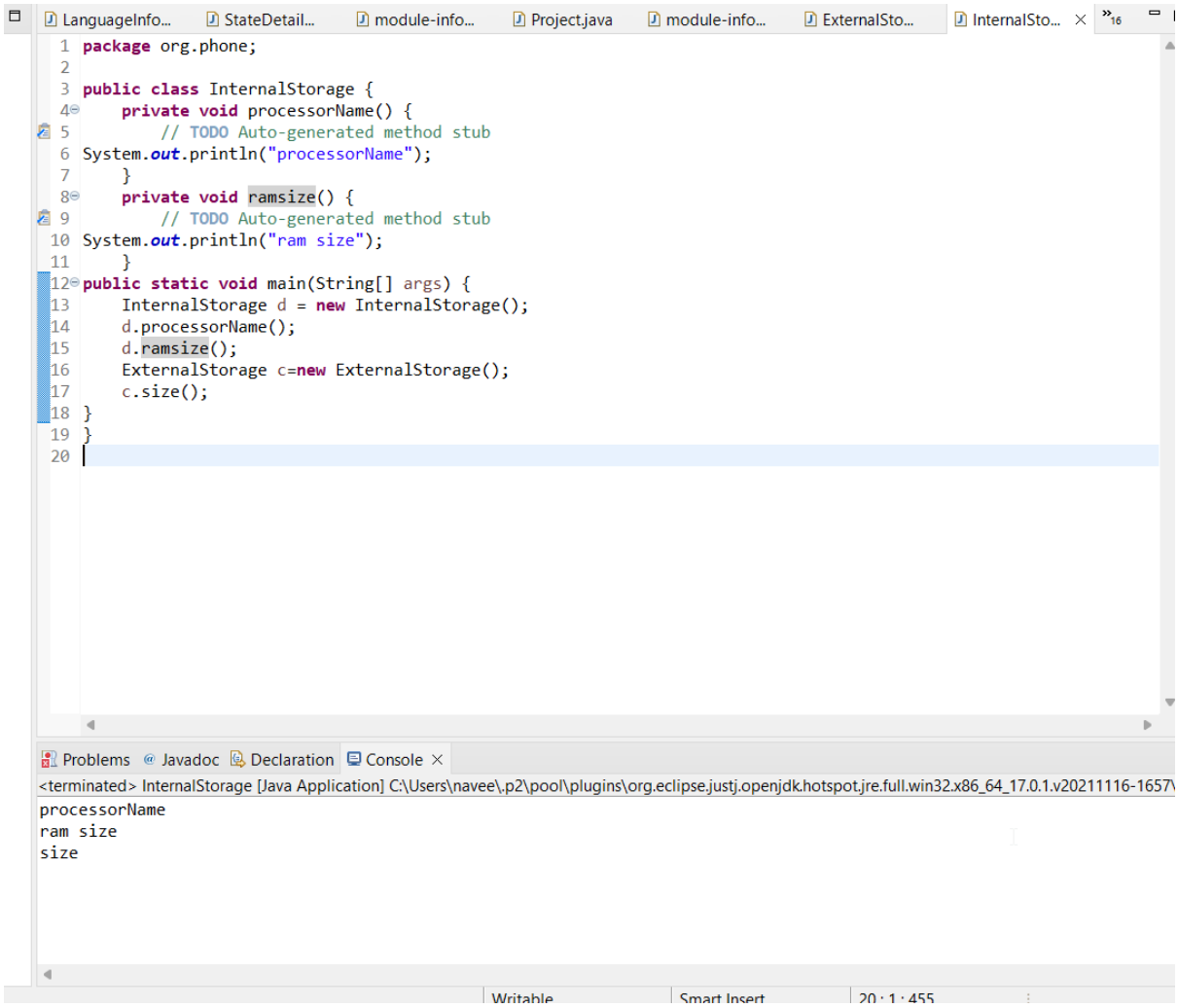
Description:
Create an object for all 4 classes inside the College class and call all classes methods also follow the all coding standards.

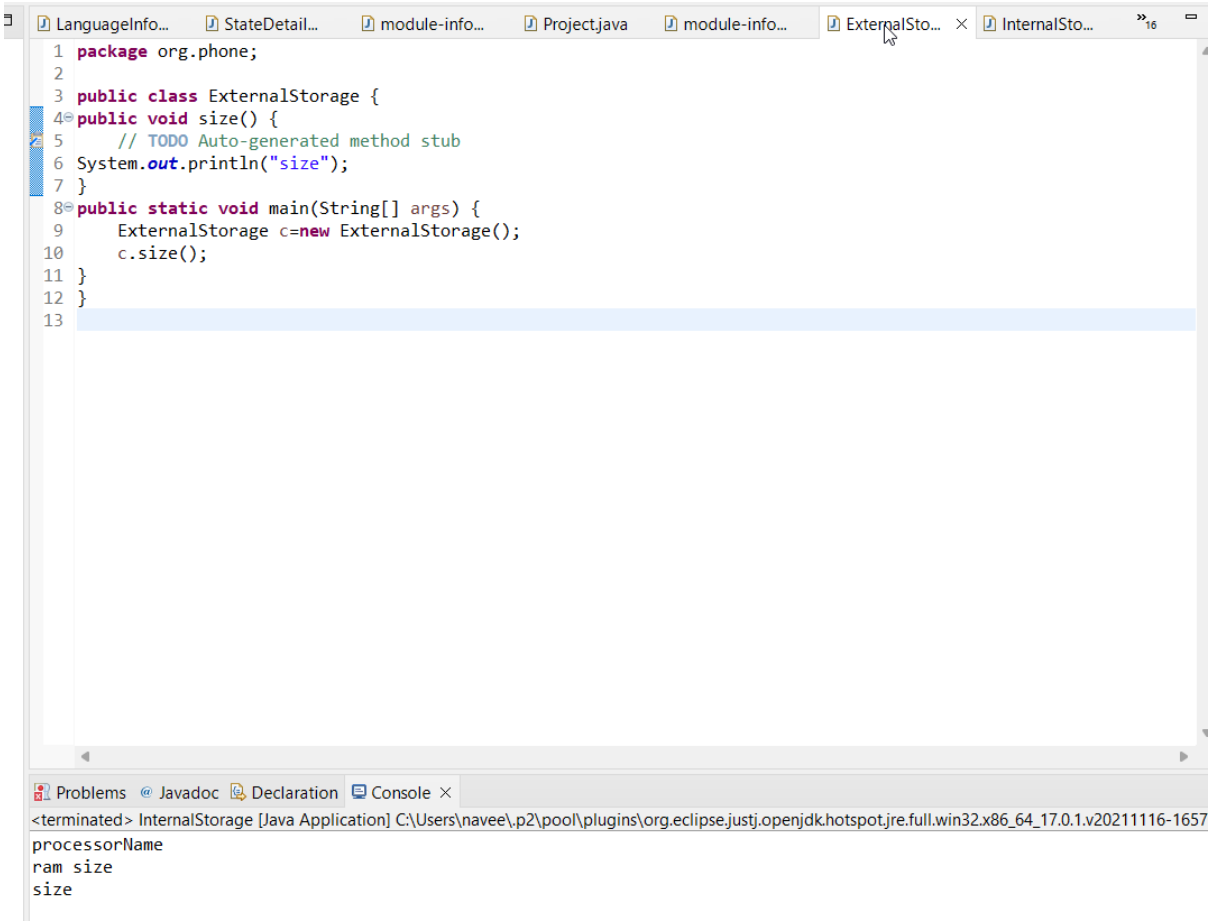

eclipse-workspace - EmployeeInformation/src/org/company/Company.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
DayOneJava
DayOneJava1
DayOneJava12
DayTwoJava
EmployeeInformation
JRE System Library [JavaSE-17]
src
org.client
Client.java
org.company
Company.java
org.emp
Employee.java
org.project
Project.java
module-info.java
LanguageDetails
JRE System Library [JavaSE-17]
src
org.lang
LanguageInfo.java
StateDetails.java
module-info.java
OneDayJava
LanguageInfo...
StateDetail...
module-info...
Employee.java
Company.j
1 package org.company;
2
3 public class Company {
4 public void companyName() {
5 // TODO Auto-generated method stub
6 System.out.println("companyName");
7 }
8 }
9


eclipse-workspace - EmployeeInformation/src/org/project/Project.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
DayOneJava
DayOneJava1
DayOneJava12
DayTwoJava
EmployeeInformation
JRE System Library [JavaSE-17]
src
org.client
Client.java
org.company
Company.java
org.emp
Employee.java
org.project
Project.java
module-info.java
LanguageDetails
JRE System Library [JavaSE-17]
src
org.lang
LanguageInfo.java
StateDetails.java
module-info.java
OneDayJava
LanguageInfo...
StateDetail...
module-info...
Employee.java
Company.java
Client.java
Project.java
1 package org.project;
2
3 public class Project {
4 public void projectName() {
5 // TODO Auto-generated method stub
6 System.out.println("projectName");
7 }
8 }
9
Problems
Javadoc
Declaration
Console
<terminated> Employee (1) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-
empName
companyName
clientName
projectName
Writable
Smart Insert
9 : 1 : 156

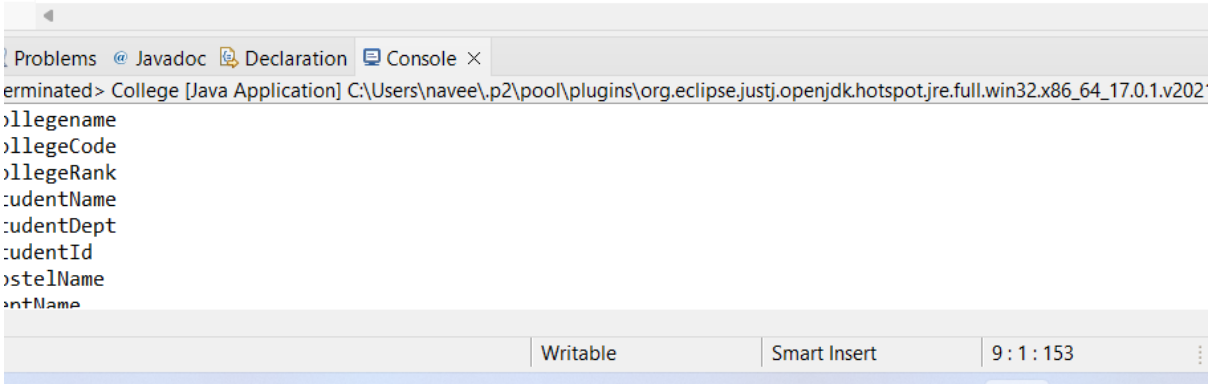

File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
DayOneJava
DayOneJava1
DayOneJava12
DayTwoJava
EmployeeInformation
JRE System Library [JavaSE-17]
src
org.client
Client.java
org.company
Company.java
org.emp
Employee.java
org.project
Project.java
module-info.java
LanguageDetails
JRE System Library [JavaSE-17]
src
org.lang
LanguageInfo.java
StateDetails.java
module-info.java
OneDayJava
LanguageInfo... StateDetail... module-info... Employee.java Company.java Client.java Project.java
1 package org.emp;
2 import org.company.Company;
3 import org.client.Client;
4 import org.project.Project;
5 public class Employee {
6 private void empName() {
7 // TODO Auto-generated method stub
8 System.out.println("empName");
9 }
10 public static void main(String[] args) {
11 Employee e =new Employee();
12 e.empName();
13 Company c = new Company();
14 c.companyName();
15 Client d = new Client();
16 d.clientName();
17 Project f = new Project();
18 f.projectName();
19
20 }
21 }
22
Problems Javadoc Declaration Console
<terminated> Employee (1) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\
empName
companyName
clientName
projectName
Writable
Smart Insert
22 : 1 : 475


eclipse-workspace - EmployeeInformation/src/org/client/Client.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
DayOneJava
DayOneJava1
DayOneJava12
DayTwoJava
EmployeeInformation
JRE System Library [JavaSE-17]
src
org.client
Client.java
org.company
Company.java
org.emp
Employee.java
org.project
Project.java
module-info.java
LanguageDetails
JRE System Library [JavaSE-17]
src
org.lang
LanguageInfo.java
StateDetails.java
module-info.java
OneDayJava
LanguageInfo... StateDetail... module-info... Employee.java Company.java Client.java Project.java
Outline
org.clien
Client
client
1 package org.client;
2
3 public class Client {
4 public void clientName() {
5 // TODO Auto-generated method stub
6 System.out.println("clientName");
7 }
8 }
9
Problems Javadoc Declaration Console
<terminated> Employee (1) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe (22
empName
companyName
clientName
projectName
Writable
Smart Insert
9 : 1 : 152


LanguageInfo...
StateDetail...
module-info...
Project.java
module-info...
ExternalSto...
InternalSto...
package org.phone;
public class InternalStorage {
private void processorName() {
// TODO Auto-generated method stub
System.out.println("processorName");
}
private void ramsize() {
// TODO Auto-generated method stub
System.out.println("ram size");
}
public static void main(String[] args) {
InternalStorage d = new InternalStorage();
d.processorName();
d.ramsize();
ExternalStorage c=new ExternalStorage();
c.size();
}
}
Problems
Javadoc
Declaration
Console
<terminated> InternalStorage [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\
processorName
ram size
size
Writable
Smart Insert
20 : 1 : 455


LanguageInfo...
StateDetail...
module-info...
Project.java
module-info...
ExternalSto...
InternalSto...
1 package org.phone;
2
3 public class ExternalStorage {
4 public void size() {
5 // TODO Auto-generated method stub
6 System.out.println("size");
7 }
8 public static void main(String[] args) {
9 ExternalStorage c=new ExternalStorage();
10 c.size();
11 }
12 }
13
Problems
Javadoc
Declaration
Console
<terminated> InternalStorage [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657
processorName
ram size
size

```java
package org.college;

public class Hostel {
public void hostelName() {
    // TODO Auto-generated method stub
System.out.println("HostelName");
}
}

```

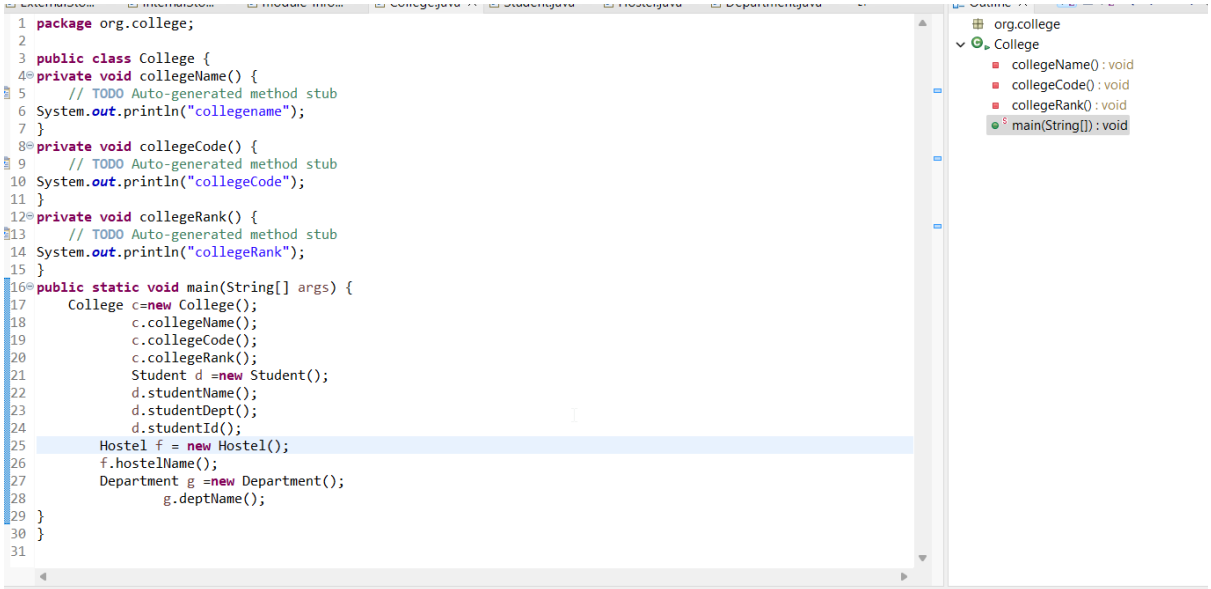


```java
package org.college;

public class Student {
public void studentName() {
    // TODO Auto-generated method stub
System.out.println("studentName");
}
public void studentDept() {
    // TODO Auto-generated method stub
System.out.println("StudentDept");
}
public void studentId() {
    // TODO Auto-generated method stub
System.out.println("StudentId");
}
}

```

Problems Javadoc Declaration Console

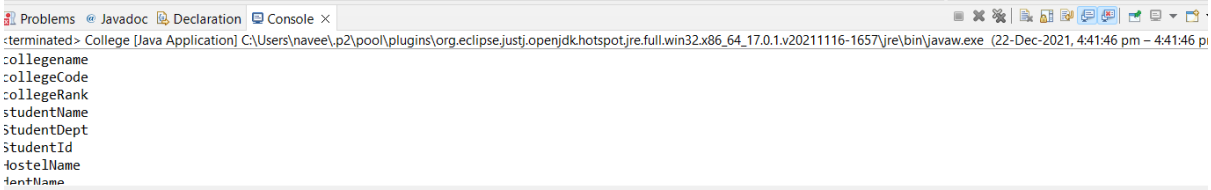
<terminated> College [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211

```
collegename
collegeCode
collegeRank
studentName
StudentDept
StudentId
HostelName
deptName
```

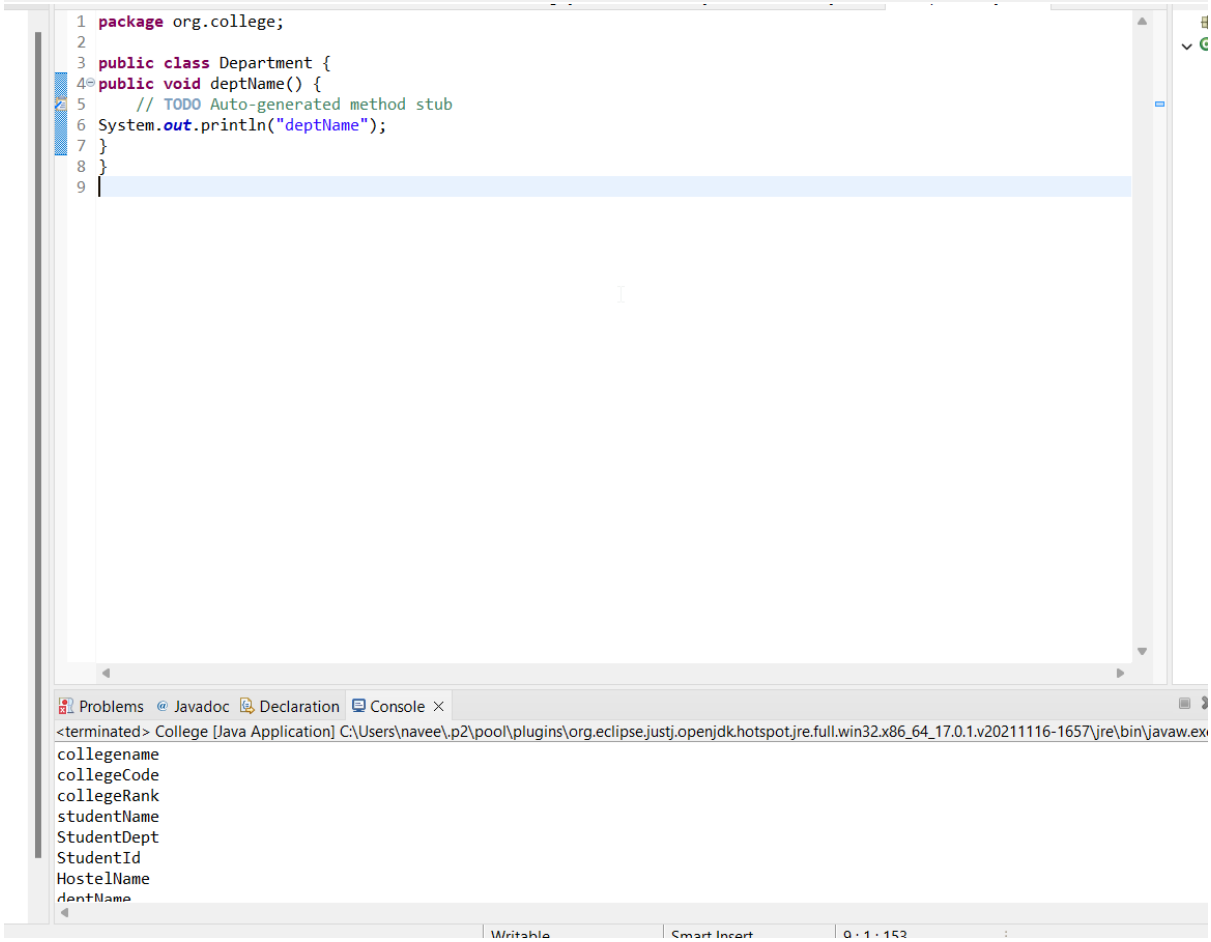


```
package org.college;

public class College {
private void collegeName() {
    // TODO Auto-generated method stub
System.out.println("collegename");
}
private void collegeCode() {
    // TODO Auto-generated method stub
System.out.println("collegeCode");
}
private void collegeRank() {
    // TODO Auto-generated method stub
System.out.println("collegeRank");
}
public static void main(String[] args) {
    College c=new College();
            c.collegeName();
            c.collegeCode();
            c.collegeRank();
            Student d =new Student();
            d.studentName();
            d.studentDept();
            d.studentId();
        Hostel f = new Hostel();
        f.hostelName();
        Department g =new Department();
                g.deptName();
}
}
```

```
org.college
College
  collegeName() : void
  collegeCode() : void
  collegeRank() : void
  main(String[]) : void
```



```
Problems  Javadoc  Declaration  Console
<terminated> College [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe  (22-Dec-2021, 4:41:46 pm – 4:41:46 pm
collegename
collegeCode
collegeRank
studentName
StudentDept
StudentId
HostelName
deptName
```



```
package org.college;

public class Department {
public void deptName() {
    // TODO Auto-generated method stub
System.out.println("deptName");
}
}
```

```
Problems  Javadoc  Declaration  Console
<terminated> College [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe
collegename
collegeCode
collegeRank
studentName
StudentDept
StudentId
HostelName
deptName
```

QUESTION 9:

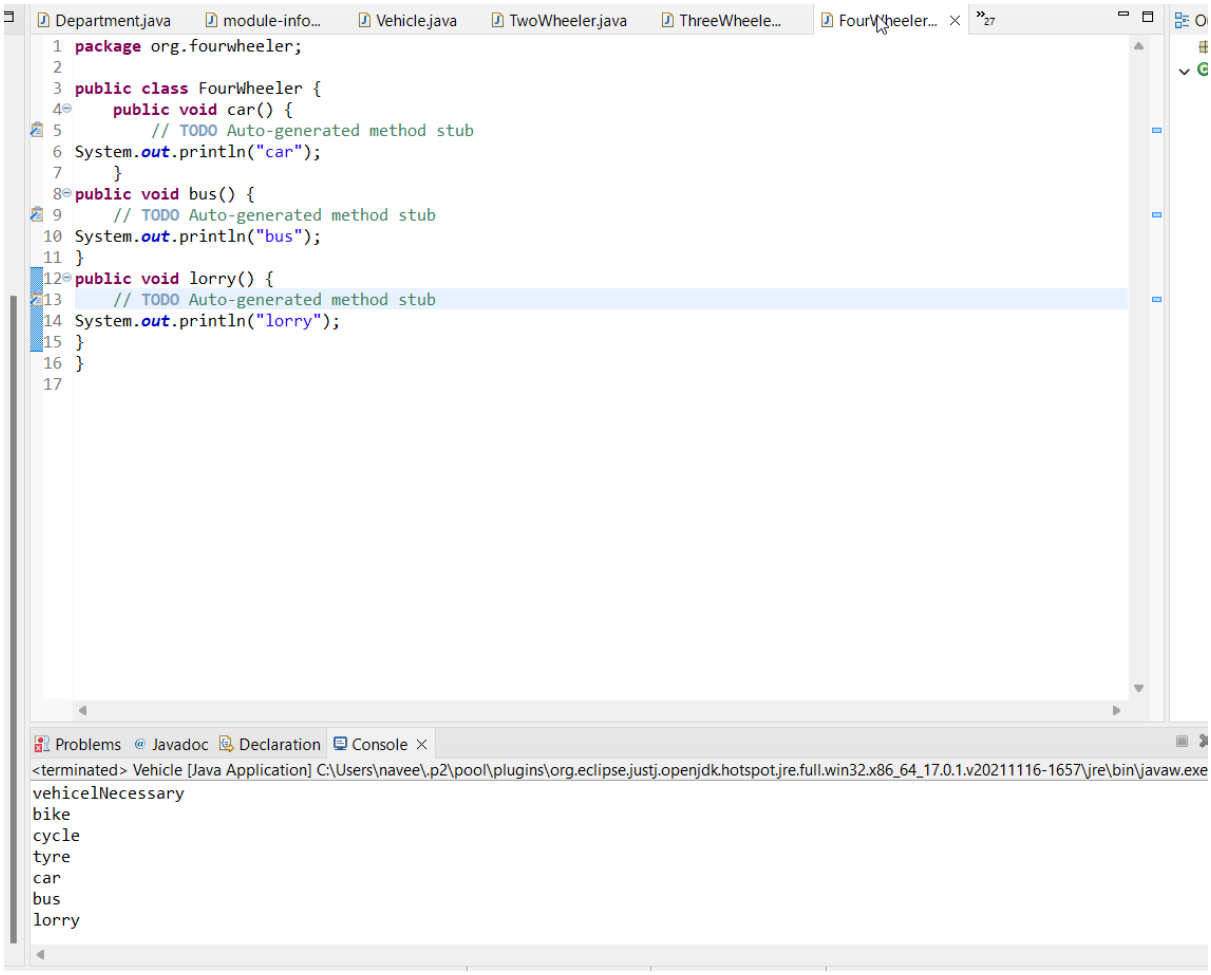
Project :VehicleInformation
Package :org.allvehicle
Class :Vehicle
Methods :VehicleNecessery()

Package :org.twowheeler
Class :TwoWheller
Methods :bike(),cycle()

Package :org.threewheeler
Class :ThreeWheeler
Methods :Auto()

Package :org.fourwheeler
Class :FourWheeler
Methods :car(),bus(),lorry()

Description:
Create an object for all 4 classes inside the Vehicle class and call all classes methods also follow the all coding standards.


Department.java
module-info...
Vehicle.java
TwoWheeler.java
ThreeWheele...
FourWheeler...
27
1 package org.fourwheeler;
2
3 public class FourWheeler {
4 public void car() {
5 // TODO Auto-generated method stub
6 System.out.println("car");
7 }
8 public void bus() {
9 // TODO Auto-generated method stub
10 System.out.println("bus");
11 }
12 public void lorry() {
13 // TODO Auto-generated method stub
14 System.out.println("lorry");
15 }
16 }
17
Problems
Javadoc
Declaration
Console
<terminated> Vehicle [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe
vehicelNecessary
bike
cycle
tyre
car
bus
lorry

```java
package org.threewheeler;

public class ThreeWheeler {
    public void tyre() {
        // TODO Auto-generated method stub
System.out.println("tyre");
    }
}

```

Problems Javadoc Declaration Console

<terminated> Vehicle [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.ex

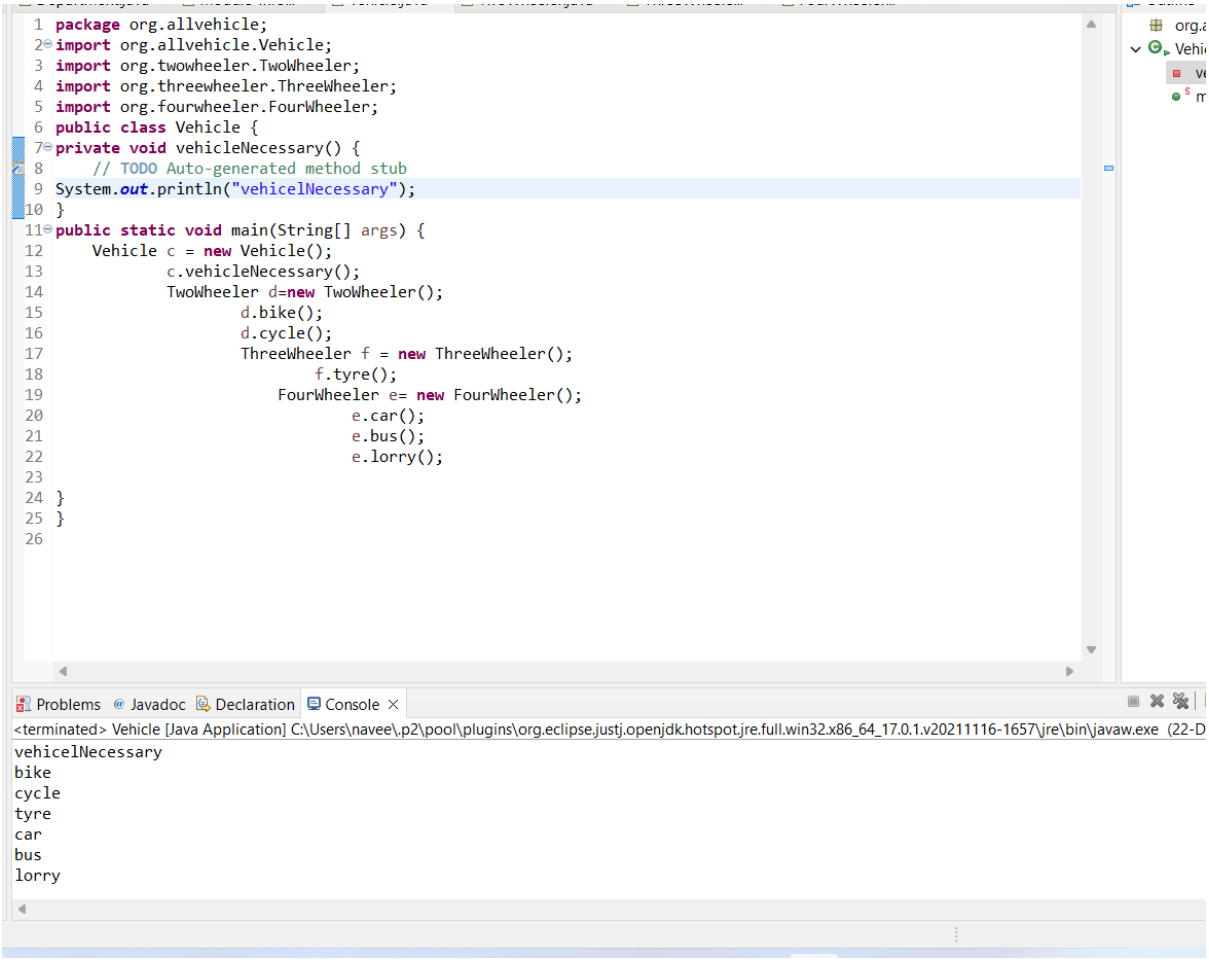
```
vehicelNecessary
bike
cycle
tyre
car
bus
lorry
```

```java
package org.twowheeler;

public class TwoWheeler {
public void bike() {
    // TODO Auto-generated method stub
System.out.println("bike");
}
public void cycle() {
    // TODO Auto-generated method stub
System.out.println("cycle");
}
}

```

Problems Javadoc Declaration Console

<terminated> Vehicle [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\j

```
vehicelNecessary
bike
cycle
tyre
car
bus
lorry
```



```java
package org.allvehicle;
import org.allvehicle.Vehicle;
import org.twowheeler.TwoWheeler;
import org.threewheeler.ThreeWheeler;
import org.fourwheeler.FourWheeler;
public class Vehicle {
private void vehicleNecessary() {
    // TODO Auto-generated method stub
System.out.println("vehicelNecessary");
}
public static void main(String[] args) {
    Vehicle c = new Vehicle();
            c.vehicleNecessary();
            TwoWheeler d=new TwoWheeler();
                    d.bike();
                    d.cycle();
                    ThreeWheeler f = new ThreeWheeler();
                            f.tyre();
                        FourWheeler e= new FourWheeler();
                                e.car();
                                e.bus();
                                e.lorry();

}
}
```

```
Problems  Javadoc  Declaration  Console
<terminated> Vehicle [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe  (22-D
vehicelNecessary
bike
cycle
tyre
car
bus
lorry
```

```
QUESTION 9:
-----------
     Project   :VehicleInformation
     Package   :org.allvehicle
     Class     :Vehicle
     Methods   :VehicleNecessery()

     Package   :org.twowheeler
     Class     :TwoWheller
     Methods   :bike(),cycle()

     Package   :org.threewheeler
     Class     :ThreeWheeler
     Methods   :Auto()

     Package   :org.fourwheeler
     Class     :FourWheeler
     Methods   :car(),bus(),lorry()

Description:
Create an object for all 4 classes  inside the Vehicle class and call all classes methods also follow the all coding standards.
```

```
UESTION 10:
--------------
     Project   :TransportInformation
     Package   :org.transport
     Class     :Transport
     Methods   :TransportForm

     Package   :org.road
     Class     :Road
     Methods   :bike(),cycle(),bus(),car()

     Package   :org.air
     Class     :Air
     Methods   :aeroPlane(),heliCopter()
```

Package :org.water
Class :Water
Methods :boat(),ship()

Description:
Create an object for all 4 classes inside the Transport class and call all classes methods also follow the all coding standards.

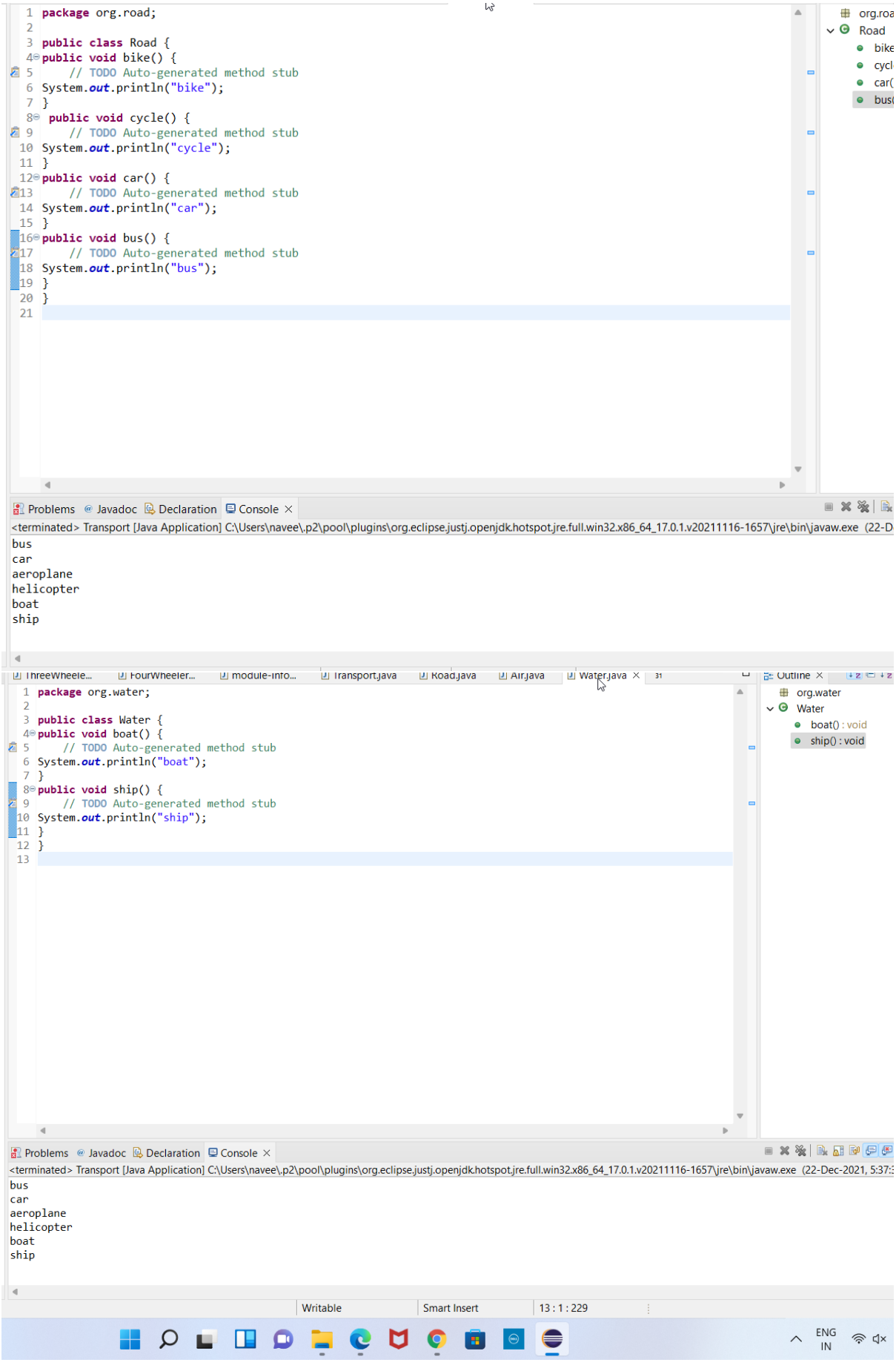
```java
package org.air;

public class Air {
public void aeroPlane() {
    // TODO Auto-generated method stub
System.out.println("aeroplane");
}
public void helicopter() {
    // TODO Auto-generated method stub
System.out.println("helicopter");
}
}

```

Problems Javadoc Declaration Console

<terminated> Transport [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\

```
bus
car
aeroplane
helicopter
boat
ship
```

Writable Smart Insert 13 : 1 : 247


package org.road;
public class Road {
public void bike() {
// TODO Auto-generated method stub
System.out.println("bike");
}
public void cycle() {
// TODO Auto-generated method stub
System.out.println("cycle");
}
public void car() {
// TODO Auto-generated method stub
System.out.println("car");
}
public void bus() {
// TODO Auto-generated method stub
System.out.println("bus");
}
}
org.roa
Road
bike
cycl
car(
bus
Problems
Javadoc
Declaration
Console
<terminated> Transport [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe (22-D
bus
car
aeroplane
helicopter
boat
ship
ThreeWheele...
FourWheeler...
module-info...
Transport.java
Road.java
Air.java
Water.java
package org.water;
public class Water {
public void boat() {
// TODO Auto-generated method stub
System.out.println("boat");
}
public void ship() {
// TODO Auto-generated method stub
System.out.println("ship");
}
}
Outline
org.water
Water
boat() : void
ship() : void
Problems
Javadoc
Declaration
Console
<terminated> Transport [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe (22-Dec-2021, 5:37:
bus
car
aeroplane
helicopter
boat
ship
Writable
Smart Insert
13 : 1 : 229
ENG
IN

UESTION 11:

Project :NetworkInformation
Package :org.network
Class :Wifi
Methods :wifiName()

Class :MobileData
Methods :dataName()

Class :Lan
Methods :lanName()

Class :Wireless
Methods :modamName()

Description:
Create an object for all 4 classes inside the Wifi class and call all classes methods also follow the all coding standards.

```java
package org.network;

public class Lan {
public void lanName() {
    // TODO Auto-generated method stub
System.out.println("lanName");
}
}
```

Problems Javadoc Declaration Console ×

<terminated> Wifi [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.ex

```
wifiName
dataName
lanName
modamName
```

```java
package org.network;

public class Wifi {
    private void wifiName() {
        // TODO Auto-generated method stub
System.out.println("wifiName");
    }
public static void main(String[] args) {
    Wifi c= new Wifi();
    c.wifiName();
    MobileData d =new MobileData();
    d.dataName();
    Lan f =new Lan();
    f.lanName();
    Wireless g = new Wireless();
    g.modamName();
}
}
```

Problems Javadoc Declaration Console ×

<terminated> Wifi [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.e

```
wifiName
dataName
lanName
modamName
```

```java
package org.network;

public class MobileData {
public void dataName() {
    // TODO Auto-generated method stub
System.out.println("dataName");
}
}

```

Problems Javadoc Declaration Console

<terminated> Wifi [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe

```
wifiName
dataName
lanName
modamName
```

```java
package org.network;

public class Wireless {
public void modamName() {
    // TODO Auto-generated method stub
System.out.println("modamName");
}
}

```

Problems Javadoc Declaration Console

<terminated> Wifi [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe

```
wifiName
dataName
lanName
modamName
```

CORE JAVA DAY—2 TASK

QUESTION 1:

Description: Using Scanner class get the below details
empId
empName
empEmail
empPhoneno
empSalary
empGender
empCity

*ScannerClass.java

```
1  package org.wipro;
2
3  import java.util.Scanner;
4
5  public class ScannerClass {
6  public static void main(String[] args) {
7      Scanner o = new Scanner(System.in);
8      System.out.println("emp id is:");
9      int id =o.nextInt();
10     System.out.println("emp name is:");
11     String name=o.next();
12     System.out.println("empEmail is");
13     String mail=o.next();
14     System.out.println("empPhoneno is:");
15     long phone =o.nextLong();
16     System.out.println("empSalary is:");
17     int salary =o.nextInt();
18     System.out.println("empGender is:");
19     String gender=o.next();
20     System.out.println("empCity is:");
21     String city=o.next();
22
23
24
25
26 }
27 }
```

Console

```
<terminated> ScannerClass [Java Applicat
emp id is:
1
emp name is:
NAVEEN
empEmail is
naveen.jan9@gmail.com
empPhoneno is:
8344350473
empSalary is:
56000
empGender is:
male
empCity is:
chennai
```

QUESTION 2:

Description: Using Scanner class get the below details
studentId
studentName
Mark1
Mark2
Mark3
Mark4
Mark5
:Find the total and average of marks

ScannerClass.java

```java
package org.wipro;

import java.util.Scanner;

public class ScannerClass {
public static void main(String[] args) {
    Scanner o = new Scanner(System.in);
    System.out.println(" studentId is:");
    int id =o.nextInt();
    System.out.println("studentName is:");
    String name=o.next();
    System.out.println(" Mark1 is:");
    int m1 =o.nextInt();
    System.out.println(" Mark2 is:");
    int m2 =o.nextInt();
    System.out.println(" Mark3 is:");
    int m3 =o.nextInt();
    System.out.println(" Mark4 is:");
    int m4 =o.nextInt();
    System.out.println(" Mark5 is:");
    int m5 =o.nextInt();
    float total;
    float average;
    total = m1+m2+m3+m4+m5;
    average = (float)(total / 5.0);
    System.out.println("the total is "+total);
    System.out.println("the average is"+average);

}
}
```

Console

```
<terminated> ScannerClass [Jav
 studentId is:
1
studentName is:
NAVEEN
 Mark1 is:
98
 Mark2 is:
90
 Mark3 is:
67
 Mark4 is:
45
 Mark5 is:
67
the total is 367.0
the average is73.4
```

QUESTION 3:

---

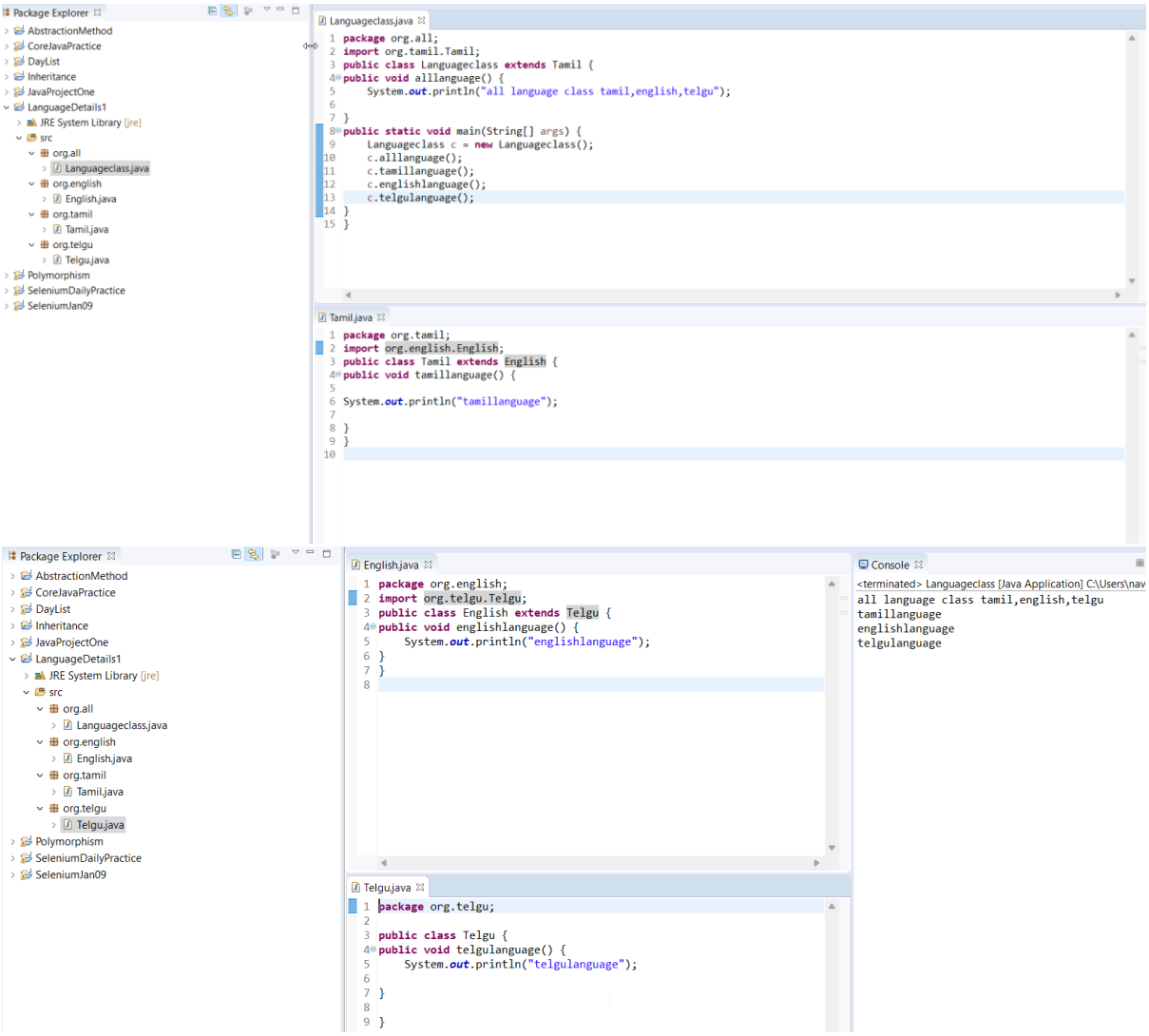
package name: org.all
Project name: LanguageDetails
Class name : Languageclass
Methods : alllanguage

package name: org.tamil
Project name: LanguageDetails
Class name : Tamil
Methods : tamillanguage

package name: org.english
Project name: LanguageDetails
Class name : English
Methods : englishlanguage

package name: org.telgu
Project name: LanguageDetails
Class name : Telgu
Methods : telgulanguage

Description:
create above 4 packages and call all your class methods into the Languageclass using multilevel inheritance.



QUESTION 4:
---

package name: org.india
Project name: SouthIndia
Class name : India
Methods : india

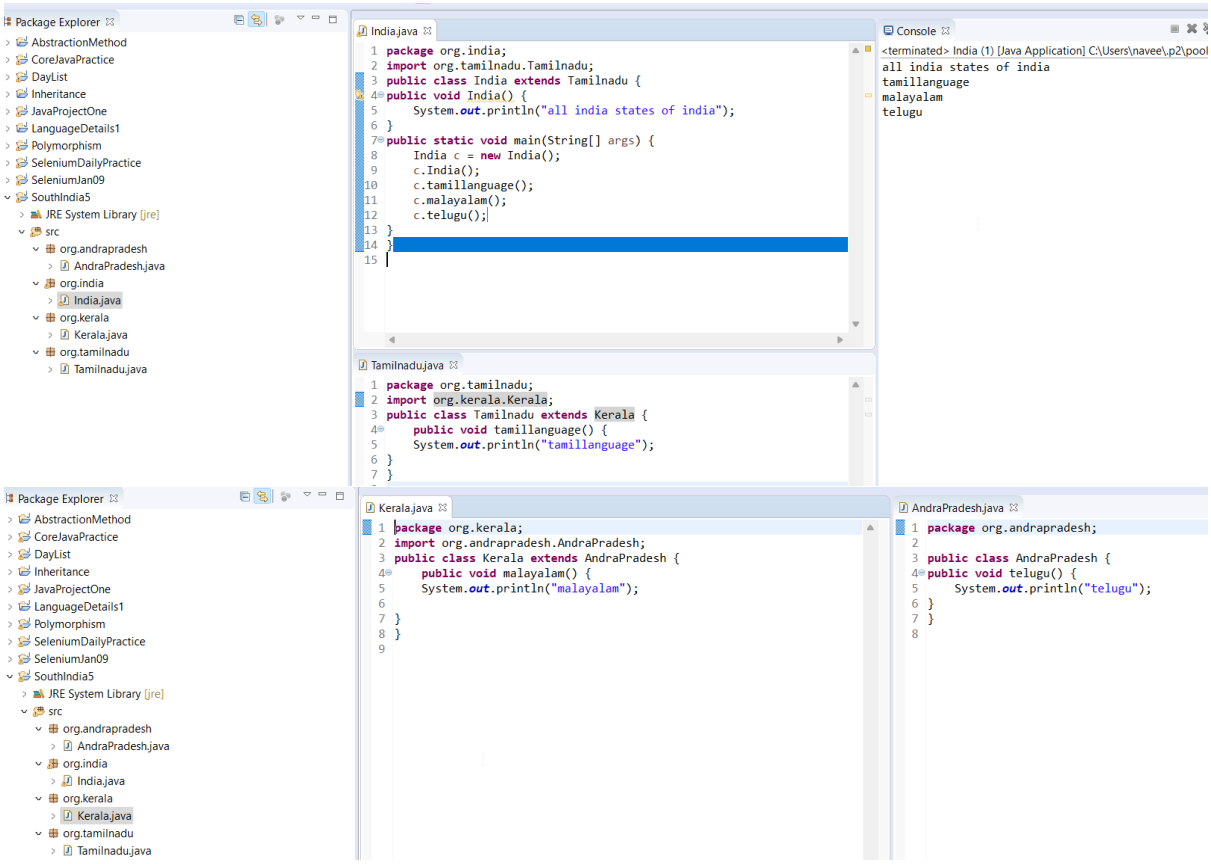
package name: org.tamilnadu
Project name: SouthIndia
Class name : TamiladuN
Methods : tamillanguage

package name: org.kerala
Project name: SouthIndia
Class name : kerala
Methods : malayalam

package name: org.andrapradesh
Project name: SouthIndia
Class name : AndhraPradesh
Methods : telugu

Description:

create above 4 packages and call all your class methods into the India using multilevel inheritance.



QUESTION 5:

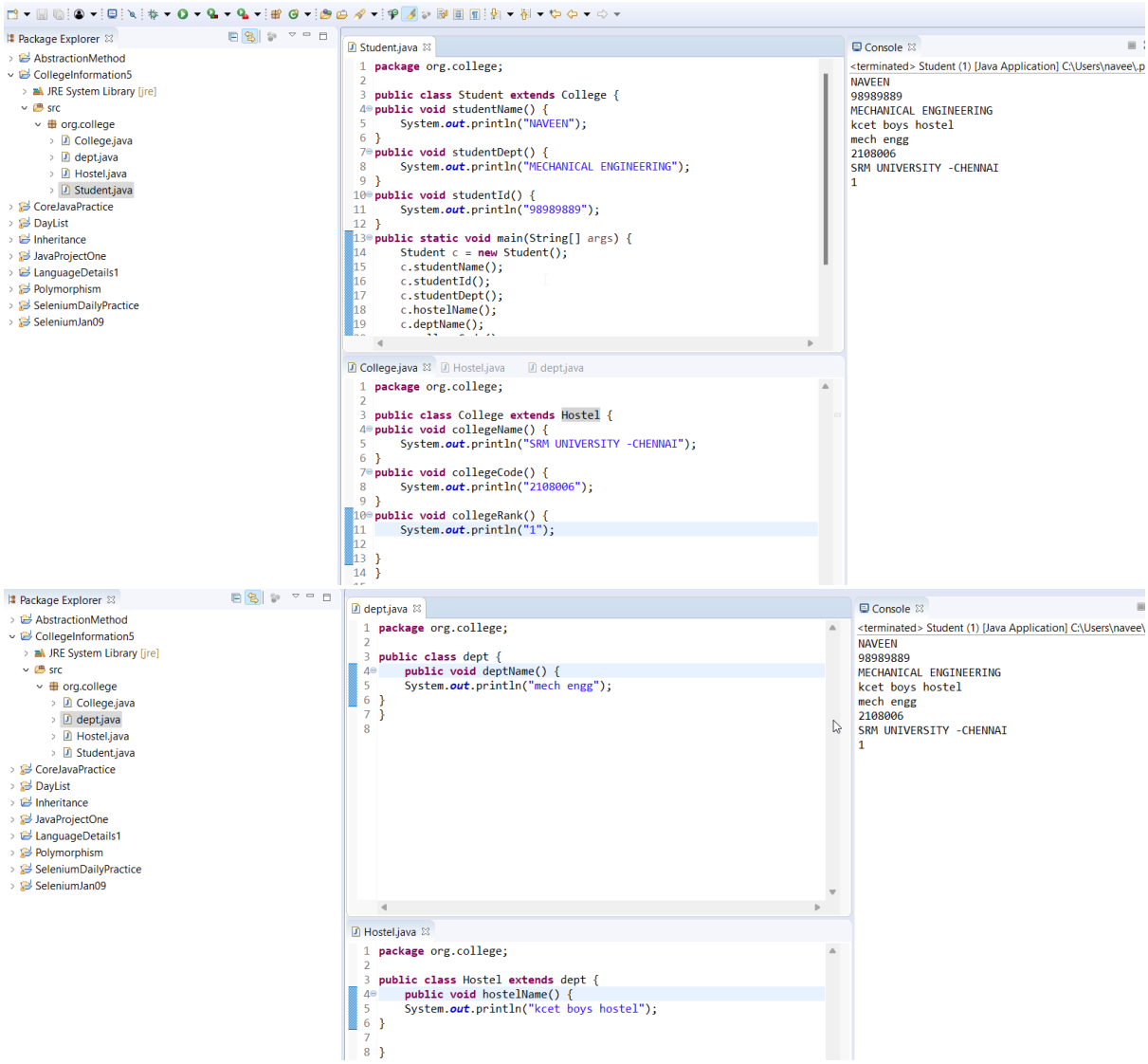
Project :CollegeInformation
Package :org.college
Class :College
Methods :collegeName(),collegeCode(),collegeRank()

Class :Student
Methods :studentName(),studentDept(),studentId()

Class :Hostel
Methods :HostelName()

Class :dept
Methods :deptName()

Description:
create above 4 class and call all your class methods into the Student using multilevel inheritance.



QUESTION 6:

Project :COmputer
Class :Computer
Methods :computerModel()

Class :Desktop
Methods :desktopSize()

Description:
create above 2 class and call all your class methods into the Desktop using single inheritance.

```java
package org.computer;

public class Desktop extends Computer {
public void deskTopSize() {
    // TODO Auto-generated method stub
System.out.println("desktopsize");
}
public static void main(String[] args) {
    Desktop c = new Desktop();
    c.deskTopSize();
    c.computerModel1();
}
}
```

Problems Javadoc Declaration Console

<terminated> Desktop [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17

```
desktopsize
computermodel
```

```java
package org.computer;

public class Computer {
public void computerModel1() {
    // TODO Auto-generated method stub
System.out.println("computermodel");
}
}
```

Problems Javadoc Declaration Console

<terminated> Desktop [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe

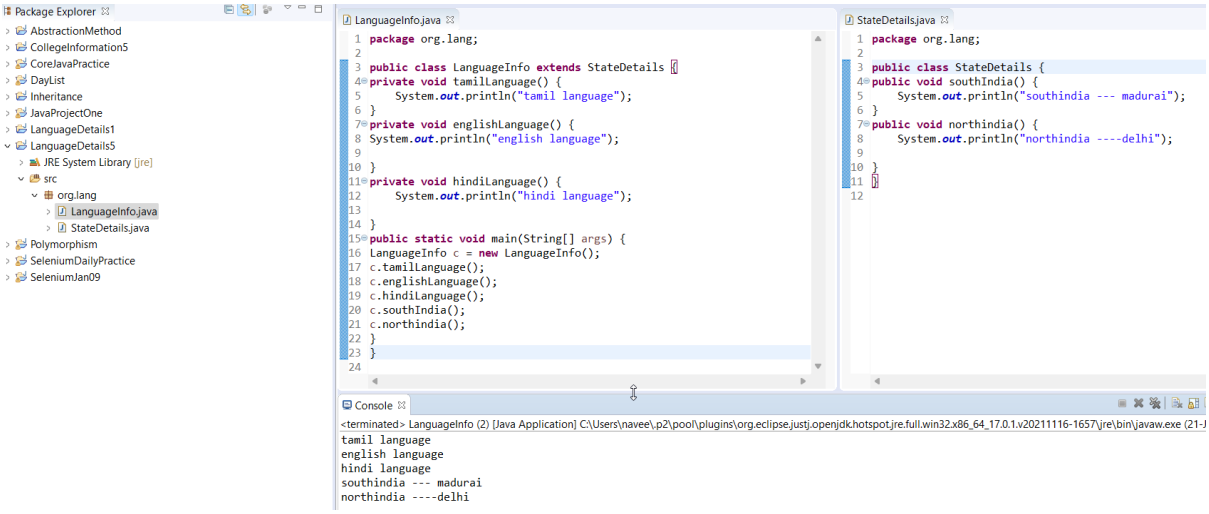
```
desktopsize
computermodel
```

Greenshot

Greensh

```
QUESTION 6:
-----------
     Project   :COmputer
     Class     :Computer
     Methods   :computerModel()

     Class     :Desktop
     Methods   :desktopSize()

Description:
```

```
create above 2 class and call all your class methods into the Desktop using single inheritance.

QUESTION 7:
-----------
     Project   :LanguageDetails
     Package   :org.lang
     Class     :LanguageInfo
     Methods   :tamilLanguage(),englishLanguage(),hindiLanguage()

     Class     :StateDetails
     Methods   :southIndia(),northIndia()

Description:
create above 2 class and call all your class methods into the LanguageInfo using single inheritance.
```

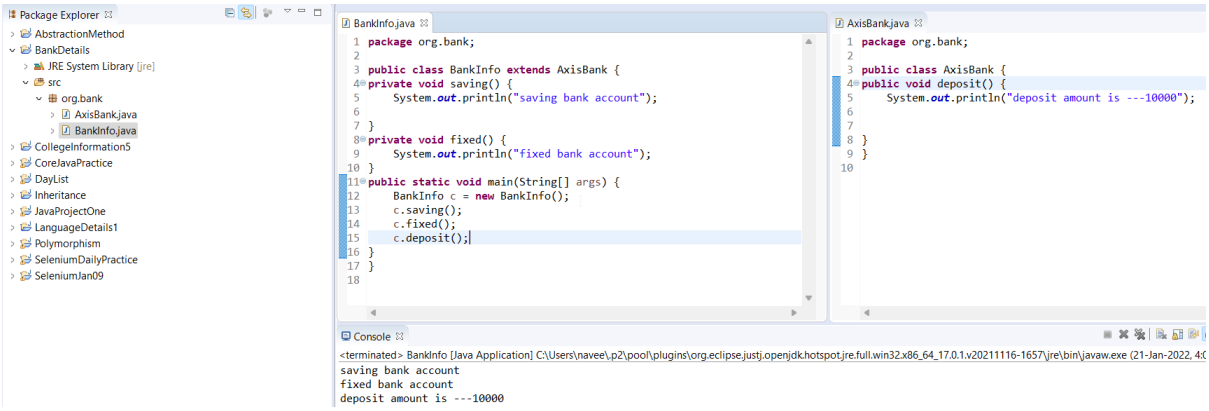


QUESTION 9:

Project :BankDetails
Package :org.bank
Class :BankInfo
Methods :saving(),fixed()

Class :AxisBank
Methods :deposit()

Description:
create above 2 class and call all your class methods into the BankInfo using single inheritan



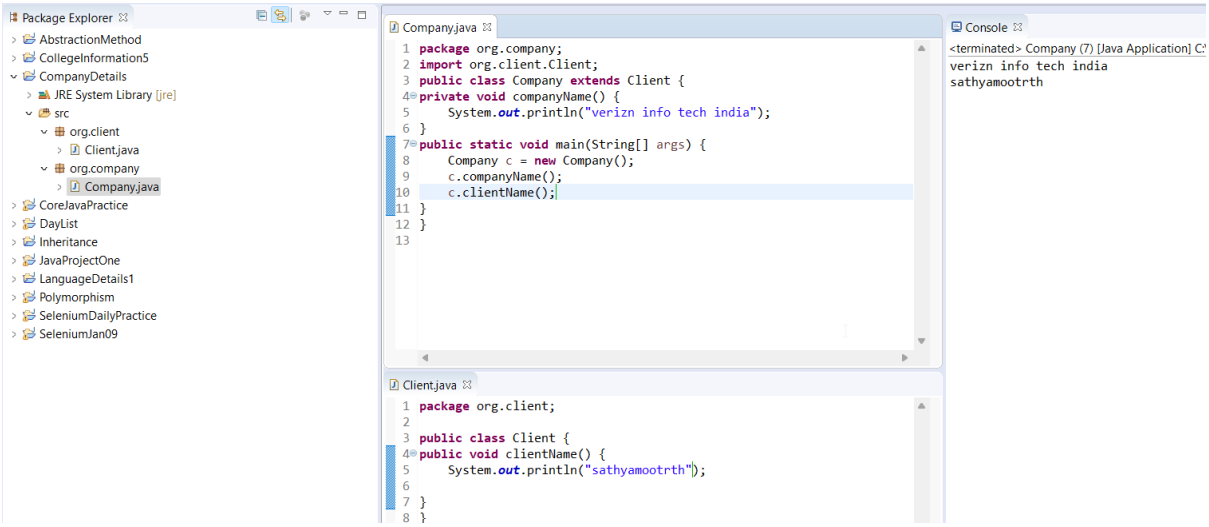
QUESTION 10:

Project :CompanyDetails
Package :org.company
Class :Company
Methods :companyName()

Package :org.client
Class :Client
Methods :clientName()

Description:

create above 2 packages and call all your class methods into the Comapany using single inheritance.



QUESTION 11:

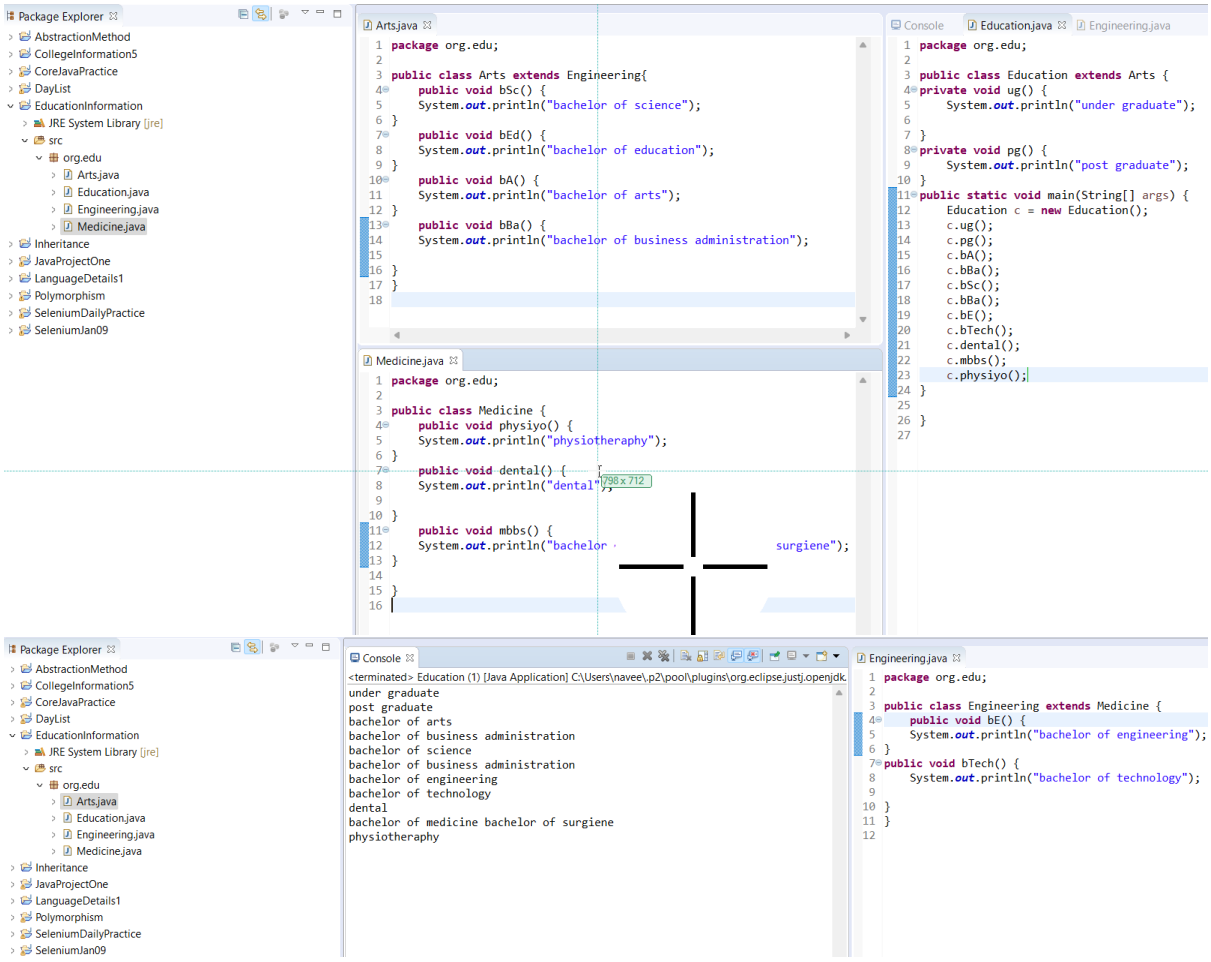
Project :EducationInformation
Package :org.edu
Class :Education
Methods :ug(),pg()

Class :Arts
Methods :bsc(),bEd(),bA(),bBA()

Class :Engineering
Methods :bE(),bTech()

Class :Medicine
Methods :physiyo(),dental(),mbbs()

Description:
create above 4 class and call all your class methods into the Education using multilevel inheritance.


Package Explorer
AbstractionMethod
CollegeInformation5
CoreJavaPractice
DayList
EducationInformation
JRE System Library [jre]
src
org.edu
Arts.java
Education.java
Engineering.java
Medicine.java
Inheritance
JavaProjectOne
LanguageDetails1
Polymorphism
SeleniumDailyPractice
SeleniumJan09
Arts.java
package org.edu;
public class Arts extends Engineering{
public void bSc() {
System.out.println("bachelor of science");
}
public void bEd() {
System.out.println("bachelor of education");
}
public void bA() {
System.out.println("bachelor of arts");
}
public void bBa() {
System.out.println("bachelor of business administration");
}
}
Medicine.java
package org.edu;
public class Medicine {
public void physiyo() {
System.out.println("physiotheraphy");
}
public void dental() {
System.out.println("dental");
}
public void mbbs() {
System.out.println("bachelor surgiene");
}
}
798 x 712
Console
Education.java
Engineering.java
package org.edu;
public class Education extends Arts {
private void ug() {
System.out.println("under graduate");
}
private void pg() {
System.out.println("post graduate");
}
public static void main(String[] args) {
Education c = new Education();
c.ug();
c.pg();
c.bA();
c.bBa();
c.bSc();
c.bBa();
c.bE();
c.bTech();
c.dental();
c.mbbs();
c.physiyo();
}
}

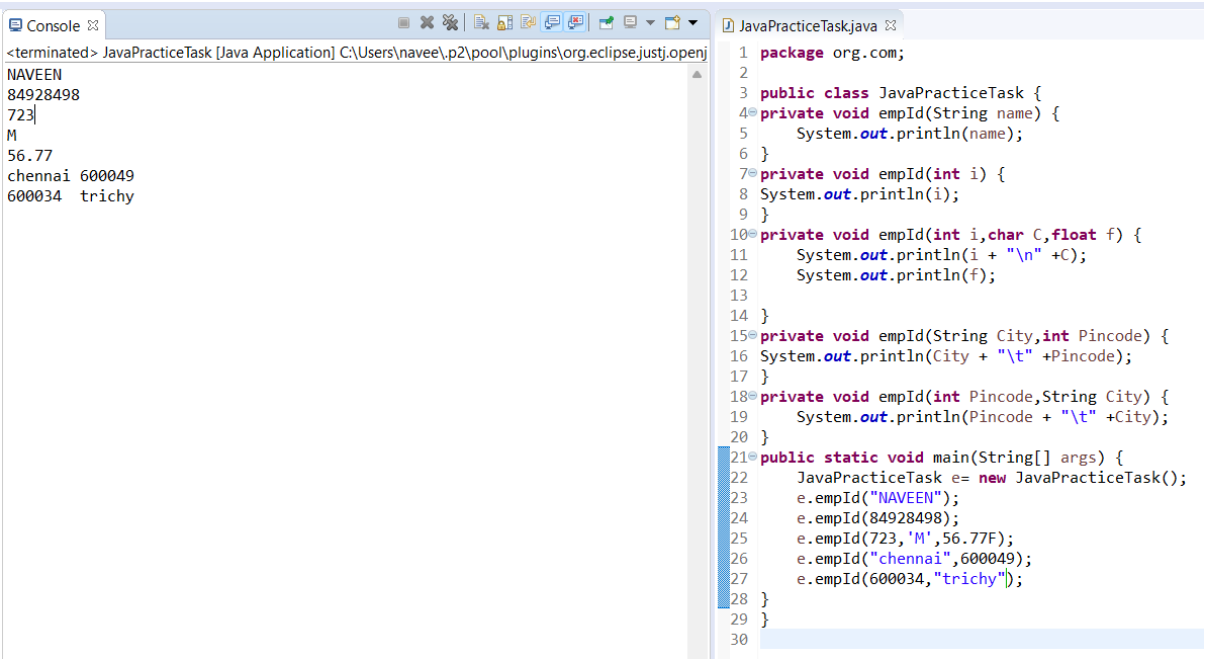
Package Explorer
AbstractionMethod
CollegeInformation5
CoreJavaPractice
DayList
EducationInformation
JRE System Library [jre]
src
org.edu
Arts.java
Education.java
Engineering.java
Medicine.java
Inheritance
JavaProjectOne
LanguageDetails1
Polymorphism
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Education (1) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.
under graduate
post graduate
bachelor of arts
bachelor of business administration
bachelor of science
bachelor of business administration
bachelor of engineering
bachelor of technology
dental
bachelor of medicine bachelor of surgiene
physiotheraphy
Engineering.java
package org.edu;
public class Engineering extends Medicine {
public void bE() {
System.out.println("bachelor of engineering");
}
public void bTech() {
System.out.println("bachelor of technology");
}
}

DAY 3 –JAVA TASK

QUESTION 2:

Project :EmployeeDetails
Package :org.emp
Class :Employee
Methods :empId()

Description
You have to overload the method empId() based on different datatype in arguments.



QUESTION 3:

Project :CompanyDetails
Package :org.company
Class :CompanyInfo
Methods :companyName()

Description
You have to overload the method companyName() based on different Number of arguments.

CompanyInfo.java

```java
package org.company;

public class CompanyInfo {


private void companyName(int g,String k) {
    System.out.println(g+ "\n" + k);
}
private void CompanyName(int i) {
    System.out.println(i);
}
private void CompanyName(int i,String s) {
    System.out.println(i + "\t" +s);
}
private void CompanyName(String j,int o) {
    System.out.println(j + "\t" +o);
}
public static void main(String[] args) {
    CompanyInfo r = new CompanyInfo();
    r.CompanyName(1,"cts");
    r.CompanyName(7877979);
    r.CompanyName(16,"east street,karappak,navalur,chenai");
    r.CompanyName("NAVEEN",676768);

}
}
```

Console

```
<terminated> CompanyInfo [Java Application] C:\Users\nav
1       cts
7877979
16      east street,karappak,navalur,chenai
NAVEEN  676768
```

QUESTION 4:

Project :MyPhone
Package :org.phone
Class :Phone
Methods :phoneInfo()

Description
You have to overload the method phoneInfo() based on different datatype order in arguments.

Phone.java

```java
package org.phone;

public class Phone {
private void phoneInfo(String s) {
System.out.println(s);
}
private void phoneInfo(int i) {
    System.out.println(i);
}
private void phoneInfo(int i,char c) {
    System.out.println(i+"\t"+c);
}
private void phoneInfo(String s,int pincode) {
    System.out.println(s+"\t"+pincode);
}
public static void main(String[] args) {
    Phone c = new Phone();
    c.phoneInfo("REDMI NOTE TEN");
    c.phoneInfo(1);
    c.phoneInfo(1,'R');
    c.phoneInfo("chennai",600019);

}
}
```

Console

```
<terminated> Phone
REDMI NOTE TEN
1
1       R
chennai 600019
```

QUESTION 5:

Project :GreensAddress

Package :org.add
Class :GreensTech
Methods :greensOmr()

Description
You have to overload the method greensOmr() based on order,type,number.

GreensTech.java

```java
package org.add;

public class GreensTech {
private void greensOmr(String name) {
    System.out.println(name);
}
private void greensOmr(int i) {
System.out.println(i);
}
private void greensOmr(int i,String C) {
System.out.println(i+"\n"+C);
}
private void greensOmr(String s,int pincode) {
    System.out.println(s+"\t"+pincode);
}
public static void main(String[] args) {
    GreensTech t = new GreensTech();
    t.greensOmr("GREENS TECHNOLOGY");
    t.greensOmr(1);
    t.greensOmr(1, "selenium");
    t.greensOmr("chennai", 600049);
}
}
```

Console

```
<terminated> GreensTe
GREENS TECHNOLOGY
1
1
selenium
chennai 600049
```

QUESTION 6:

Project :BankDetails
Package :org.bank
Class :BankInfo
Methods :saving(),fixed(),deposit()

Class :AxisBank
Methods :deposit()

Description:
You have to override the method deposit in AxisBank.

AxisBank.java

```java
package org.bank;

public class AxisBank extends BankInfo {
private void deposit() {
    System.out.println("100");

}
public static void main(String[] args) {
    AxisBank c = new AxisBank();
    c.deposit();
}
}
```

Console

<terminated> AxisBank (1) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.h

```
100
```

BankInfo.java

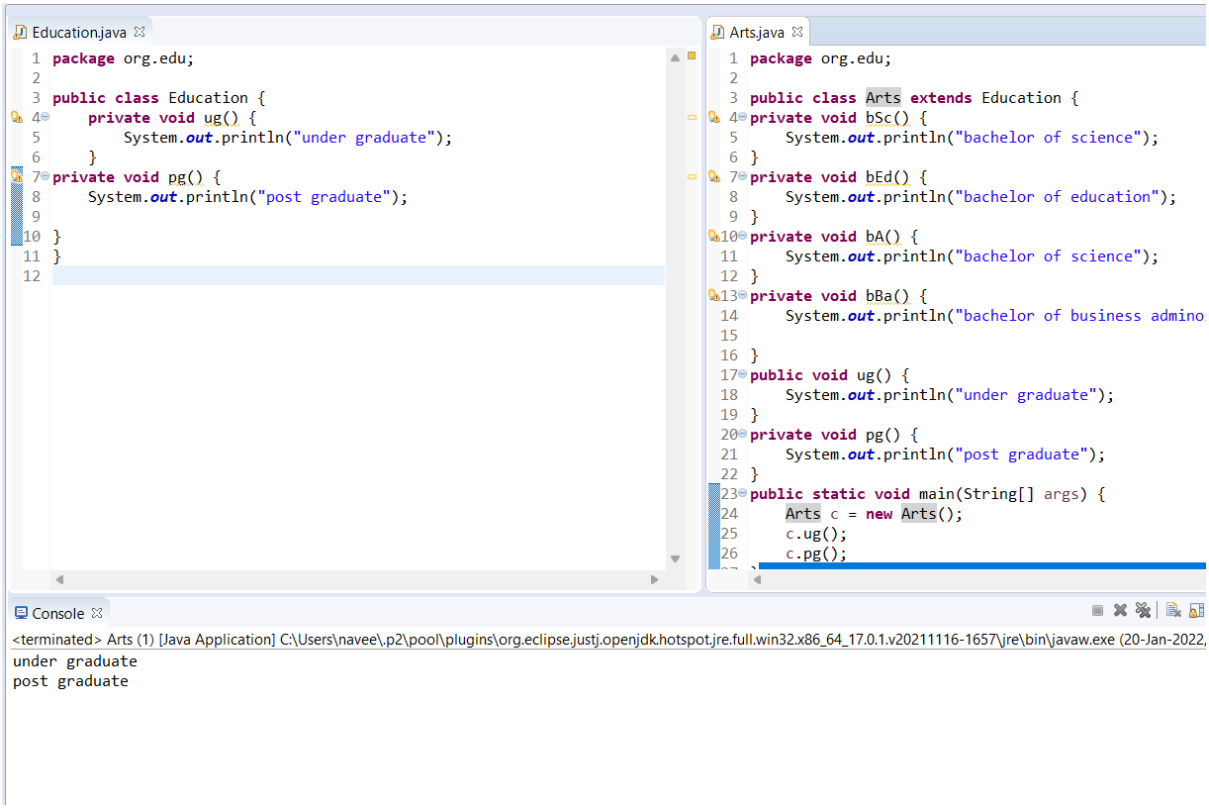
```java
package org.bank;

public class BankInfo {
private void saving() {
    System.out.println("5.54");
}
private void fixed() {
    System.out.println("8.99");

}
private void deposit() {
    System.out.println("100");

}

}
```

QUESTION 7:

Project :EducationInformation
Package :org.edu
Class :Education
Methods :ug(),pg()

Class :Arts
Methods :bSc(),bEd(),bA(),bBA(),ug(),pg()

Description:
You have to override the method ug(),pg() in Arts.



QUESTION 8:

Project :UniversityInformation
Package :org.univ
Class :University
Methods :ug(),pg()

Class :College
Methods :ug(),pg()

Description:
ug(),pg() is just a templete in University class and You have to override the method ug(),pg() in College class.

Education.java

```java
package org.univ;

public class Education {
private void ug() {
    System.out.println("UNDER GRADUATE");

}
private void pg() {
    System.out.println("POST GRADUATE");
}

}
```

Console

```
<terminated> College
UNDER GRADUATE
post graduate
```

College.java

```java
package org.univ;

public class College extends Education {
private void ug() {
    System.out.println("UNDER GRADUATE");
}
private void pg() {
    System.out.println("post graduate");
}
public static void main(String[] args) {


College c = new College();
c.ug();
c.pg();
}
}
```

QUESTION 9:

Project :BikeInformation
Package :org.bike
Interface :Bike
Methods :cost(),speed()

Class :Ktm
Methods :cost(),speed()

Description:
cost(),speed() is just a templete in Bike Interface and You have to override the method cost(),speed() in Ktm class.

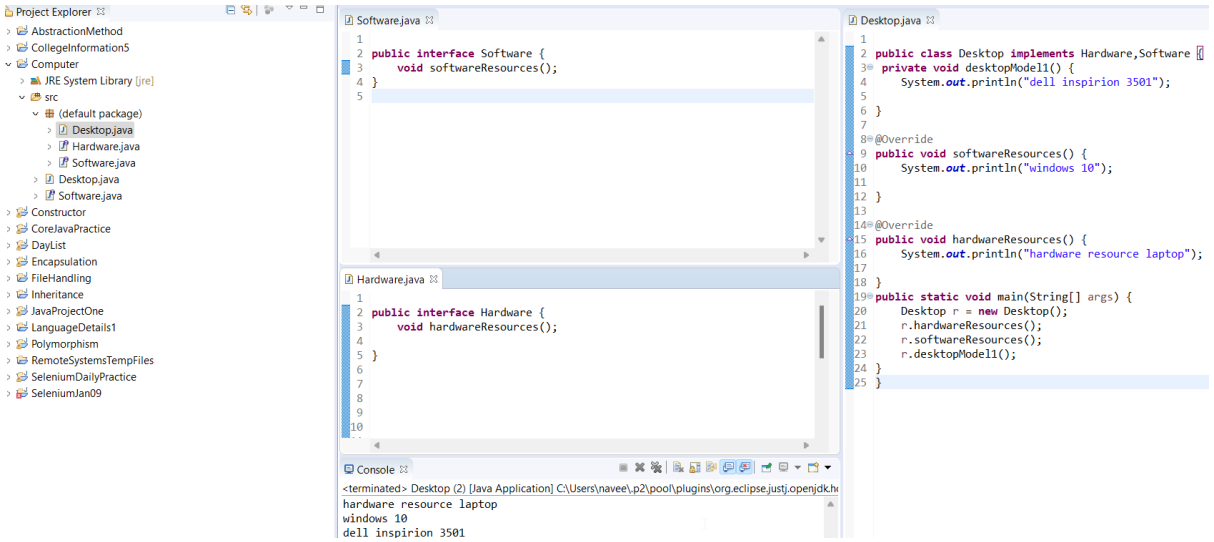
QUESTION 10:

Project :Computer
Interface :HardWare
Methods :hardwareResources()

Interface :Software
Methods :softwareResources()

Class :Desktop
Methods :desktopModel()

Description:
create 2 Interface and archieve multiple inheritance.



DAY4-

Day4_ControlStatements

DAY4:

1.if/else if
2.Loopings(for,while,do-while)
3.Switch case
4.break/continue
QUESTIONS(Find the output)

QUESTION 1:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 100; i++) {
      if (i == 5) {

      }
      System.out.println(i);

    }

  }

}
```

QUESTION 2:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 10; i++) {
      if (i == 5) {
        break;
      }
      System.out.println(i);

    }

  }

}
```

QUESTION 3:

```
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 10; i++) {
      if (i == 5) {
        continue;
      }
      System.out.println(i);
```

```
    }

  }

}
```

QUESTION 4:

```
------------
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = 1; j <= 3; j++) {
        System.out.println(j);
      }

    }

  }

}
```

QUESTION 5:

```
------------
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = 1; j <= 3; j++) {
        System.out.println(i);
      }

    }

  }

}
```

QUESTION 6:

```
-----------
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = 1; j <= i; j++) {
        System.out.println(j);
      }

    }

  }

}
```

QUESTION 7:

```
-----------
package org.test;
```

```
public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = i + 1; j <= 3; j++) {
        System.out.println(j);
      }

    }

  }

}
```

QUESTION 8:

```
------------
package org.test;

public class Hello {
  public static void main(String[] args) {
    for (int i = 1; i <= 3; i++) {
      for (int j = i + 1; j <= i; j++) {
        System.out.println(j);
      }

    }

  }

}
```

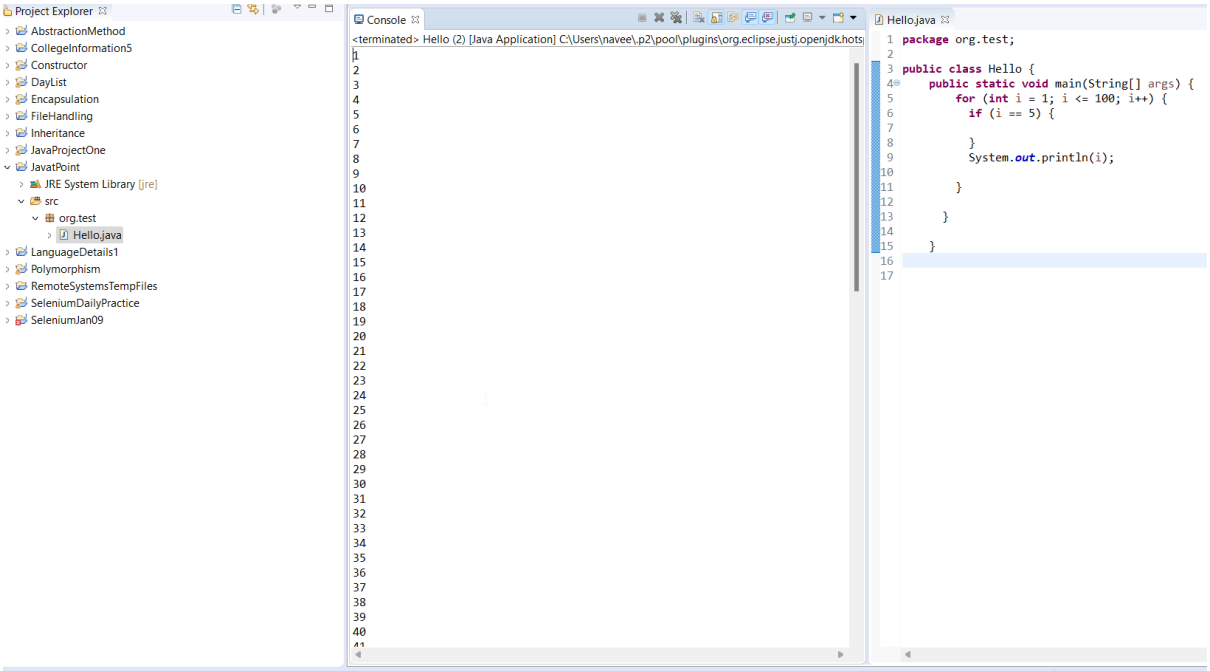
QUESTION 9:

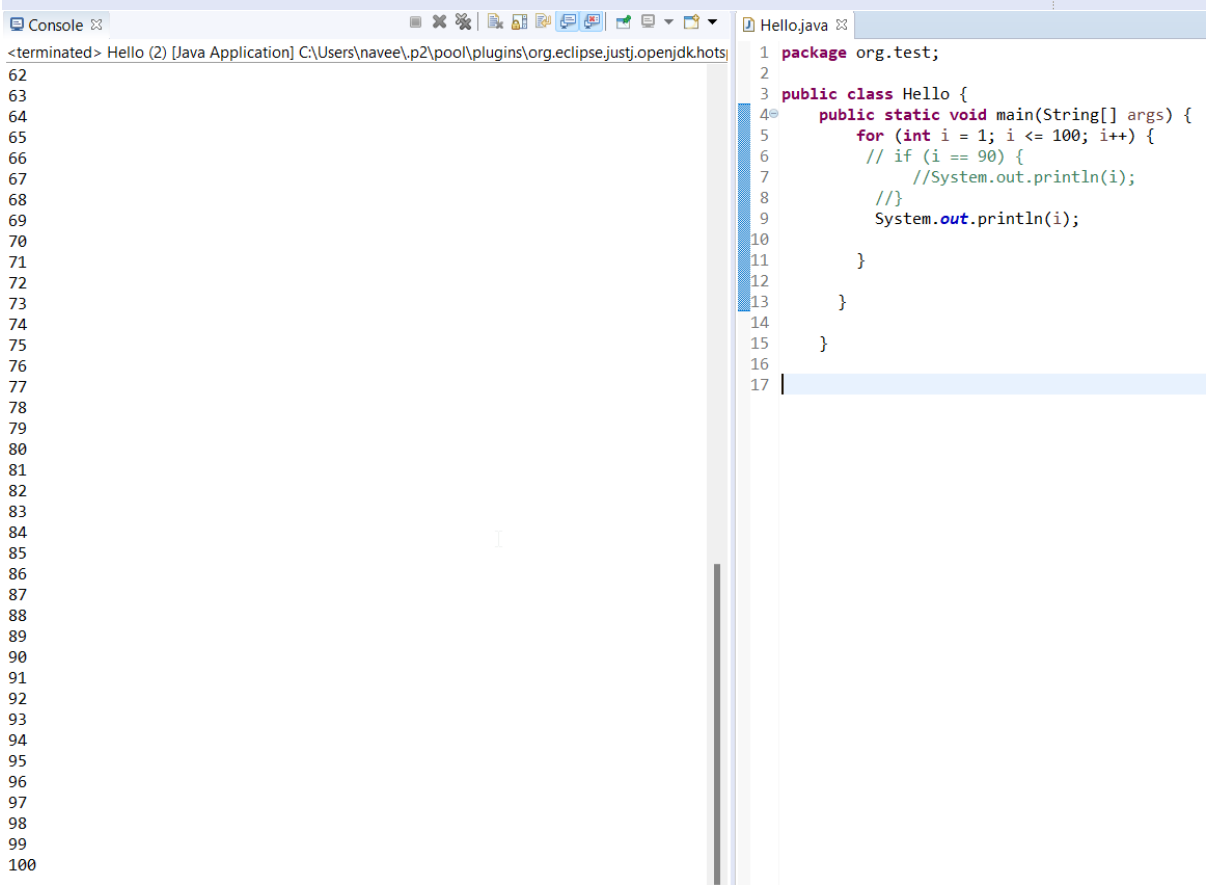
```
------------
package org.test;

public class Hello {
  public static void main(String[] args) {
    int i=5;
      if (i == 5) {
        break;
      }
      System.out.println(i);
  }

}
```

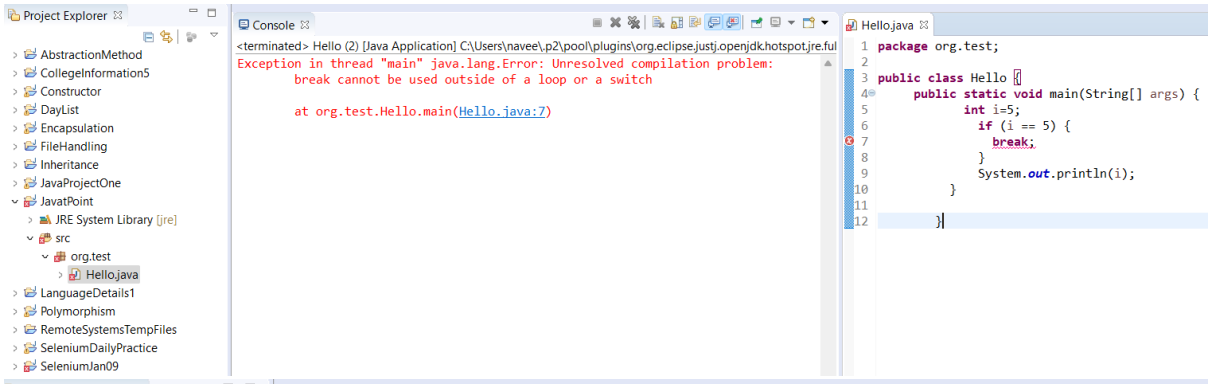
QUESTION 10:

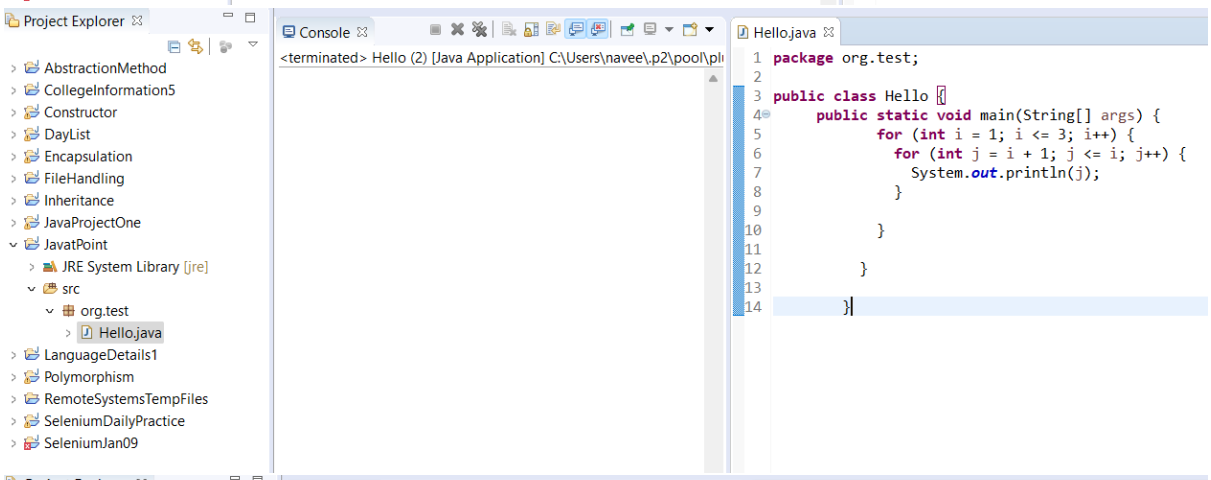
```
------------
package org.test;

public class Hello {
  public static void main(String[] args) {
    int i=5;
      if (i == 5) {
        continue;
      }
      System.out.println(i);
  }

}
```

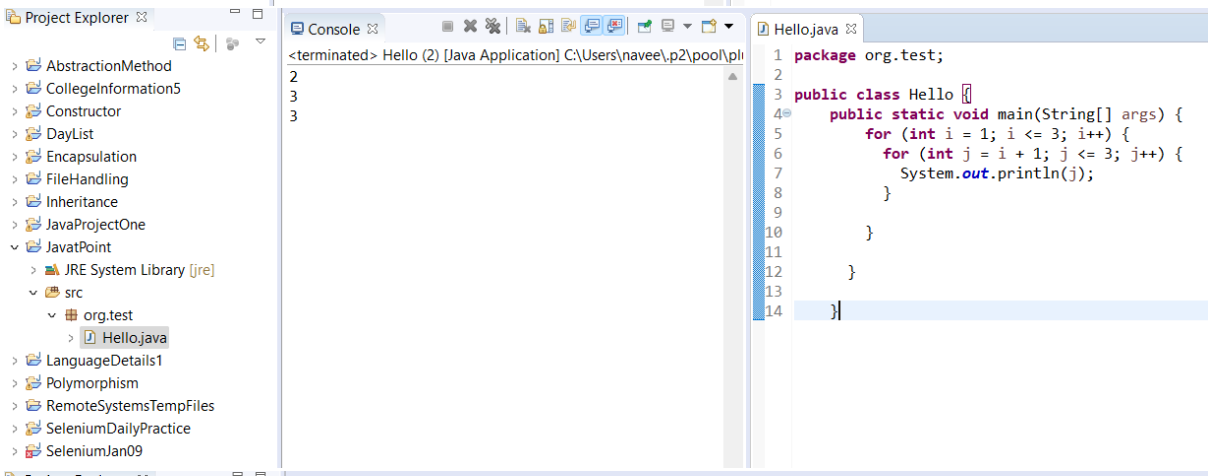
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

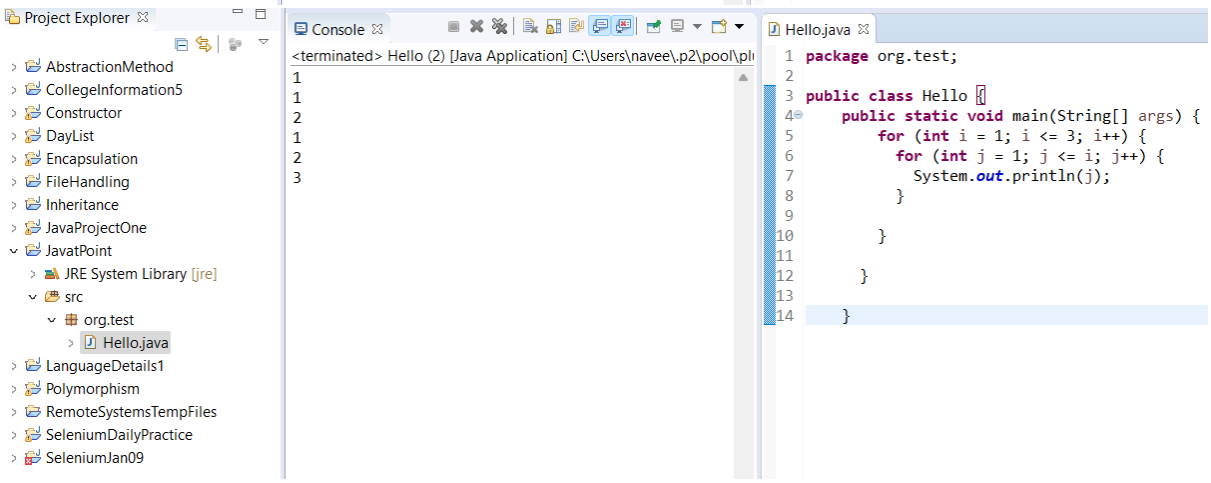

Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hots
Hello.java
package org.test;
public class Hello {
public static void main(String[] args) {
for (int i = 1; i <= 100; i++) {
if (i == 5) {
}
System.out.println(i);
}
}
}

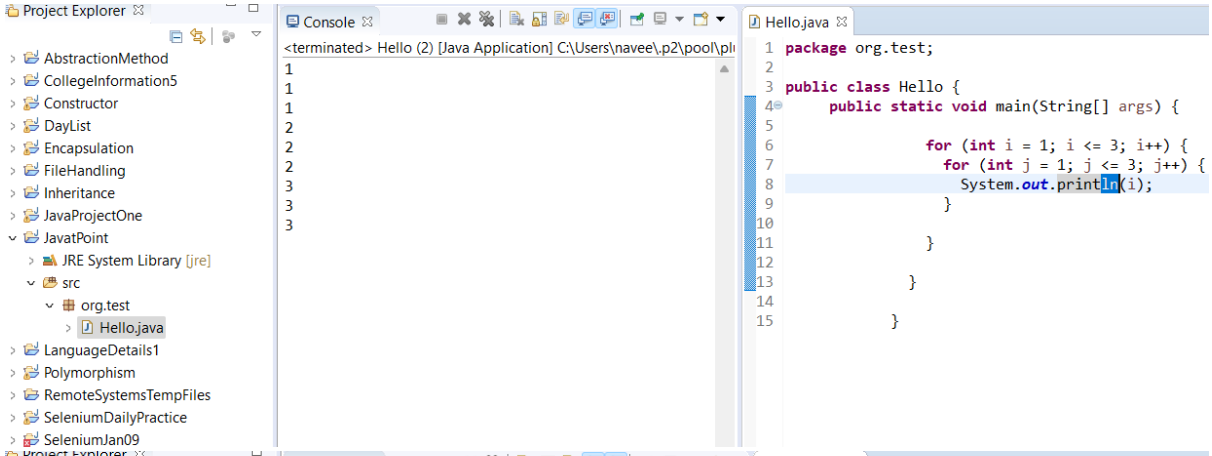

Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hots
Hello.java
package org.test;
public class Hello {
public static void main(String[] args) {
for (int i = 1; i <= 100; i++) {
// if (i == 90) {
//System.out.println(i);
//}
System.out.println(i);
}
}
}

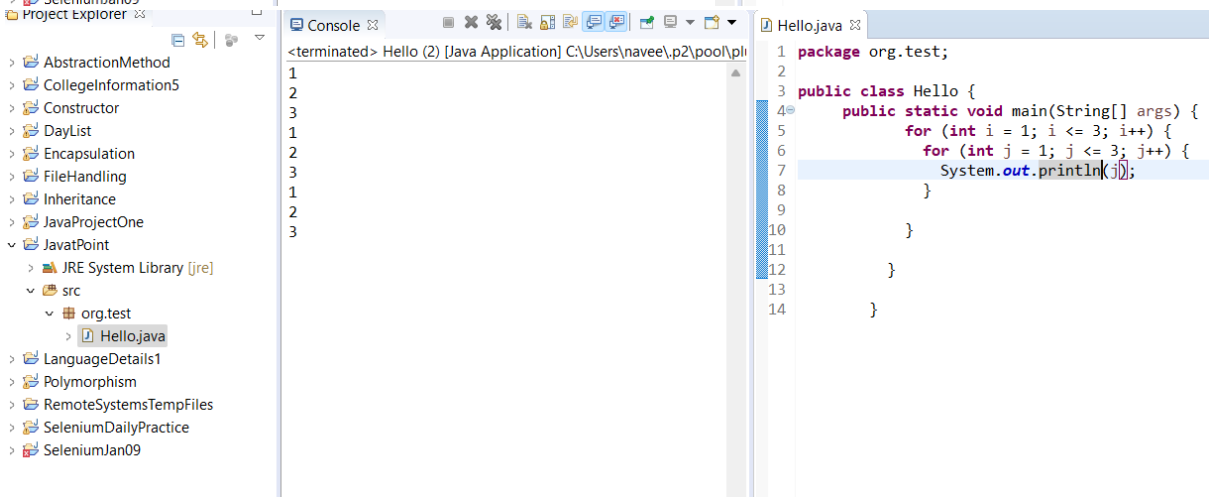

Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.ful
Exception in thread "main" java.lang.Error: Unresolved compilation problem:
break cannot be used outside of a loop or a switch
at org.test.Hello.main(Hello.java:7)
Hello.java
package org.test;
public class Hello {
public static void main(String[] args) {
int i=5;
if (i == 5) {
break;
}
System.out.println(i);
}
}

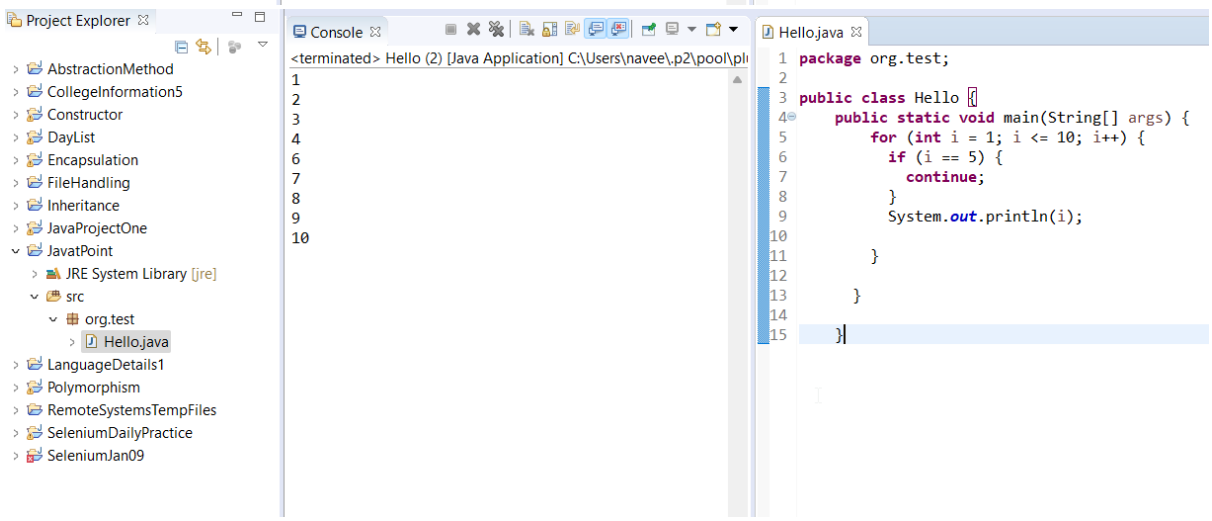

Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\pl
Hello.java
package org.test;
public class Hello {
public static void main(String[] args) {
for (int i = 1; i <= 3; i++) {
for (int j = i + 1; j <= i; j++) {
System.out.println(j);
}
}
}
}

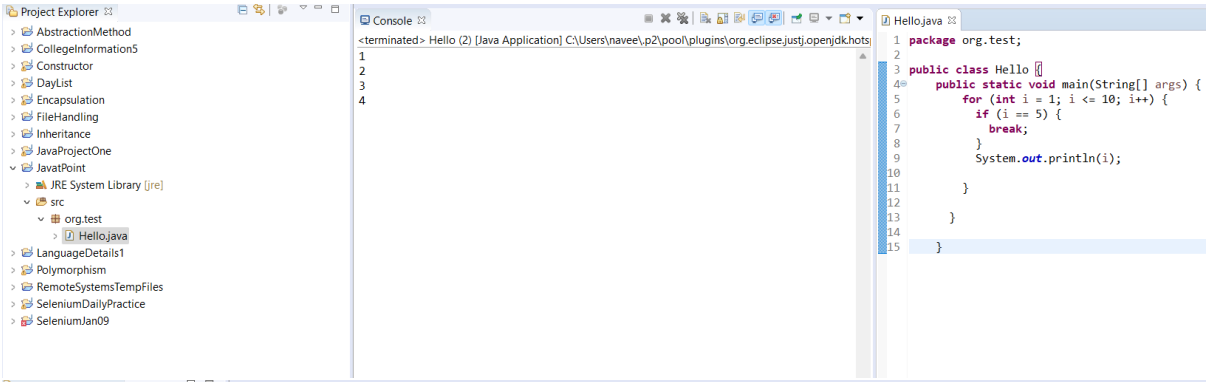

Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\pl
2
3
3
Hello.java
package org.test;
public class Hello {
public static void main(String[] args) {
for (int i = 1; i <= 3; i++) {
for (int j = i + 1; j <= 3; j++) {
System.out.println(j);
}
}
}
}

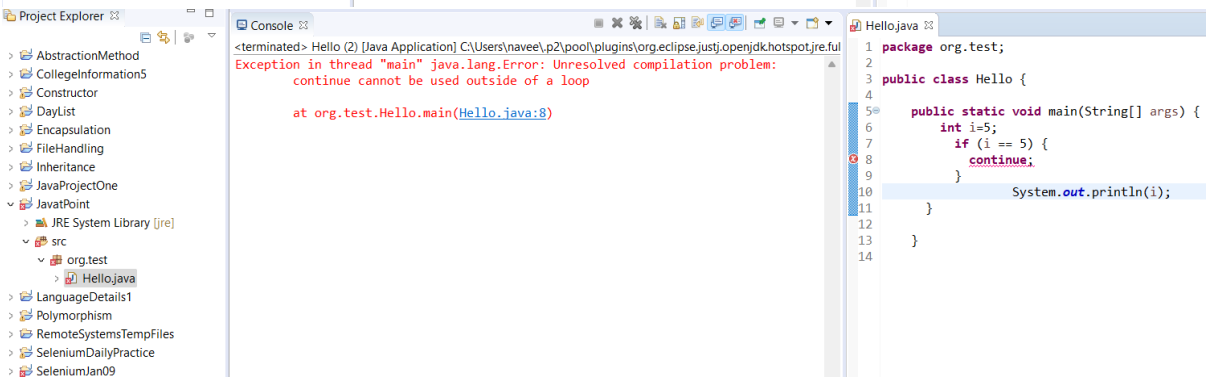

Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\pl
1
1
2
1
2
3
Hello.java
package org.test;
public class Hello {
public static void main(String[] args) {
for (int i = 1; i <= 3; i++) {
for (int j = 1; j <= i; j++) {
System.out.println(j);
}
}
}
}


Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\pl
1
1
1
2
2
2
3
3
3
Hello.java
1 package org.test;
2
3 public class Hello {
4 public static void main(String[] args) {
5
6 for (int i = 1; i <= 3; i++) {
7 for (int j = 1; j <= 3; j++) {
8 System.out.println(i);
9 }
10
11 }
12
13 }
14
15 }


Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\pl
1
2
3
1
2
3
1
2
3
Hello.java
1 package org.test;
2
3 public class Hello {
4 public static void main(String[] args) {
5 for (int i = 1; i <= 3; i++) {
6 for (int j = 1; j <= 3; j++) {
7 System.out.println(j);
8 }
9
10 }
11
12 }
13
14 }


Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\pl
1
2
3
4
6
7
8
9
10
Hello.java
1 package org.test;
2
3 public class Hello {
4 public static void main(String[] args) {
5 for (int i = 1; i <= 10; i++) {
6 if (i == 5) {
7 continue;
8 }
9 System.out.println(i);
10
11 }
12
13 }
14
15 }



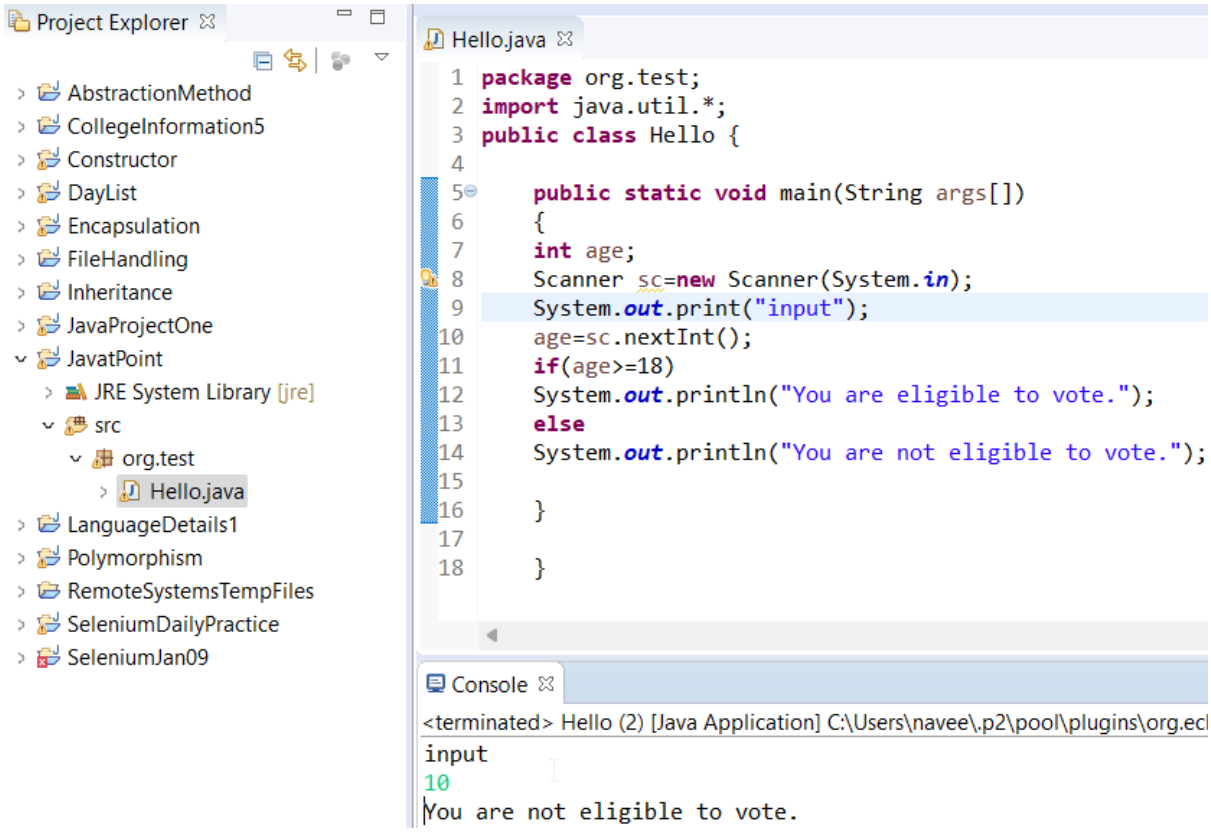


QUESTIONS(Programs)

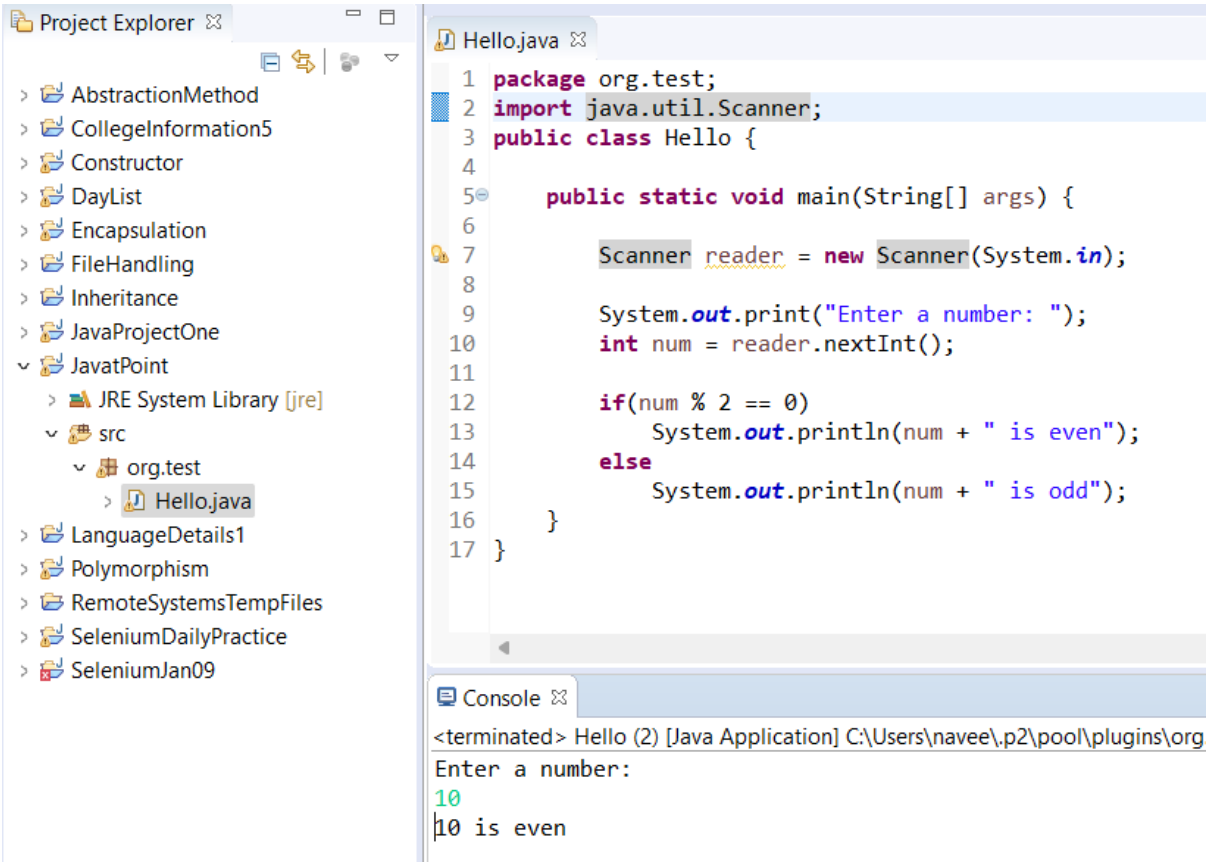
QUESTION 1:

Description: Write Java program to allow the user to input his/her age.
Then the program will show if the person is eligible to vote.
A person who is eligible to vote must be older than or equal 1 to 18 years old.
Example:

Input = 10
Output = print not eligible.




```
QUESTION 2:
-----------
Description: Write a program to find even or odd number

Example:
---------
Input  = 10
Output = Even
```

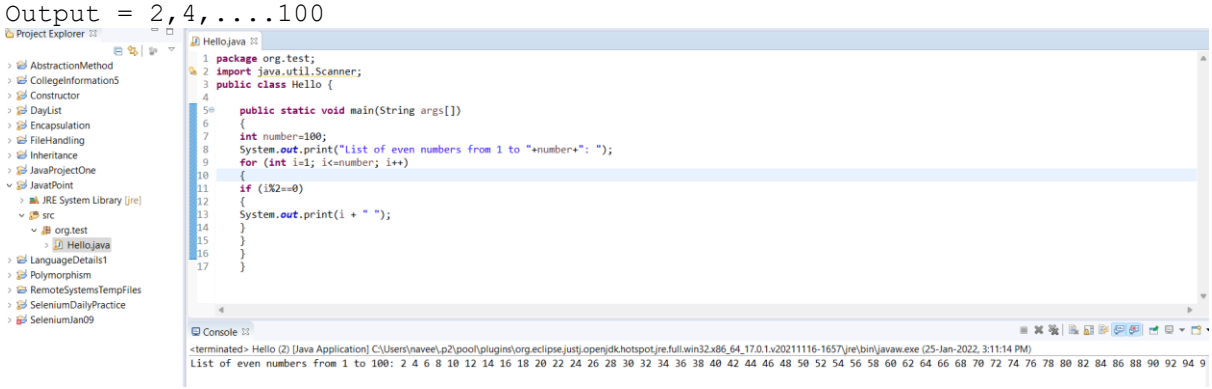


QUESTION 3:

Description: Write a program to print even number from 1 to 100

Example:

Output = 2,4,....100

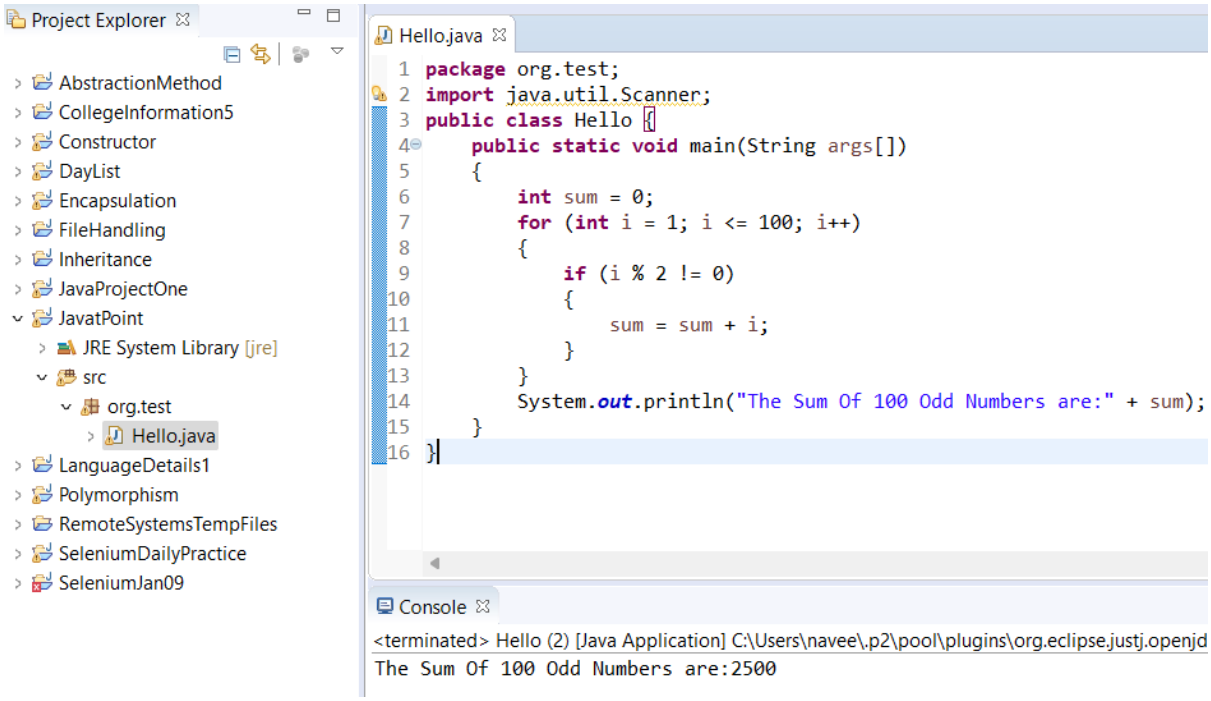


QUESTION 4:

Description: Find the sum of odd number 1 to 100

Example:

Output = 2500

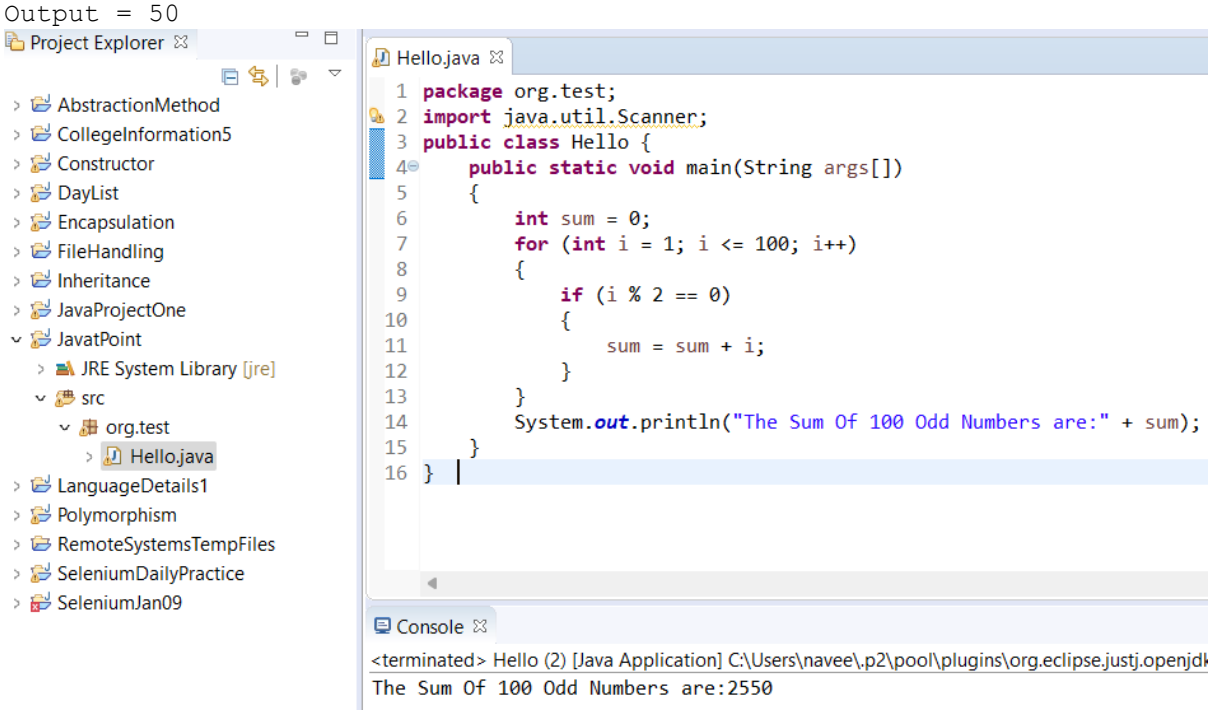


QUESTION 5:

Description: Count of even number 1 to 100

Example:

Output = 50

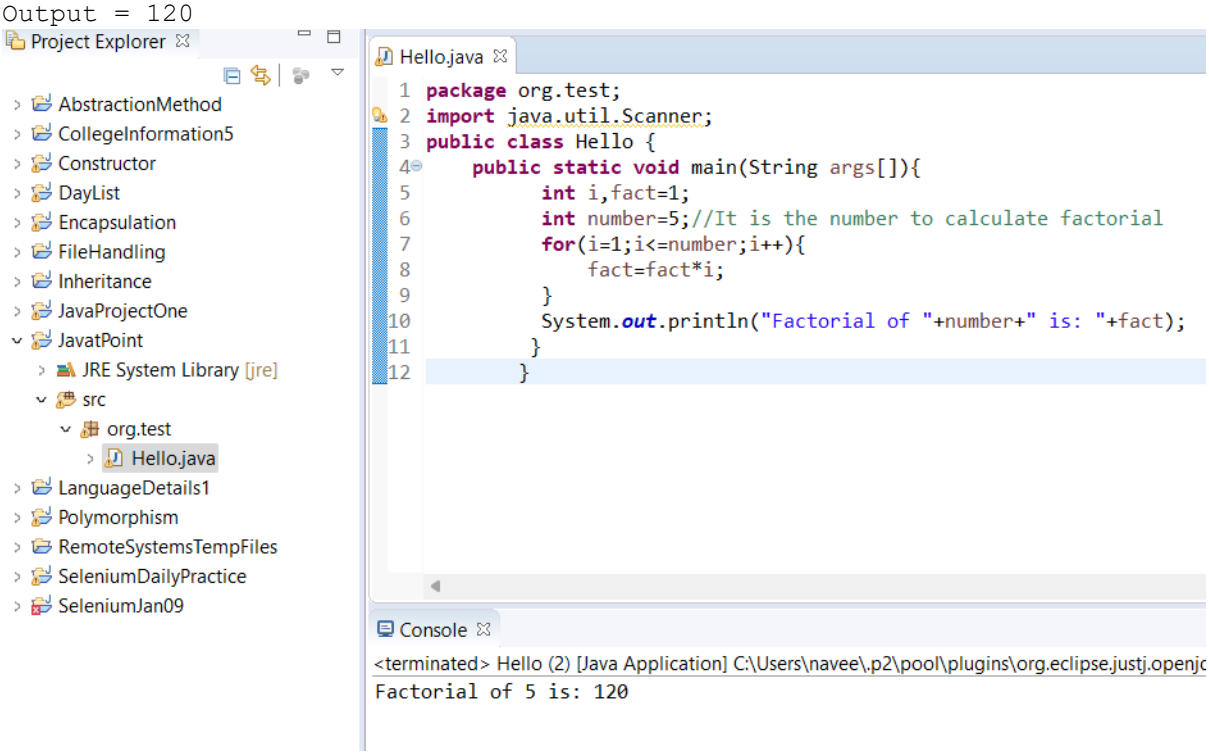


QUESTION 6:

Description: Write a program to find the factorial of a number.

Example:

Input = 5
Output = 120

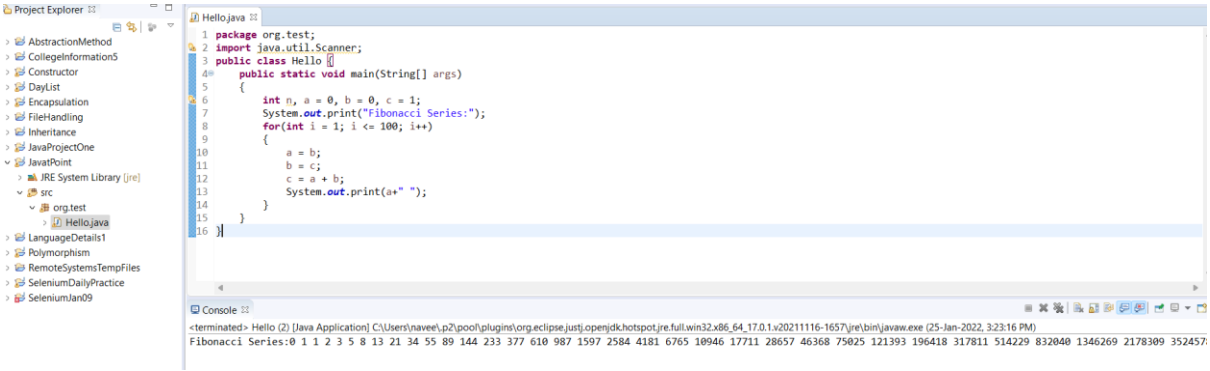


QUESTION 7:

Description: Write a program to print the fibonacci series of a number 1 to 100.

Example:

Output = 0,1,1,2,3,5.....

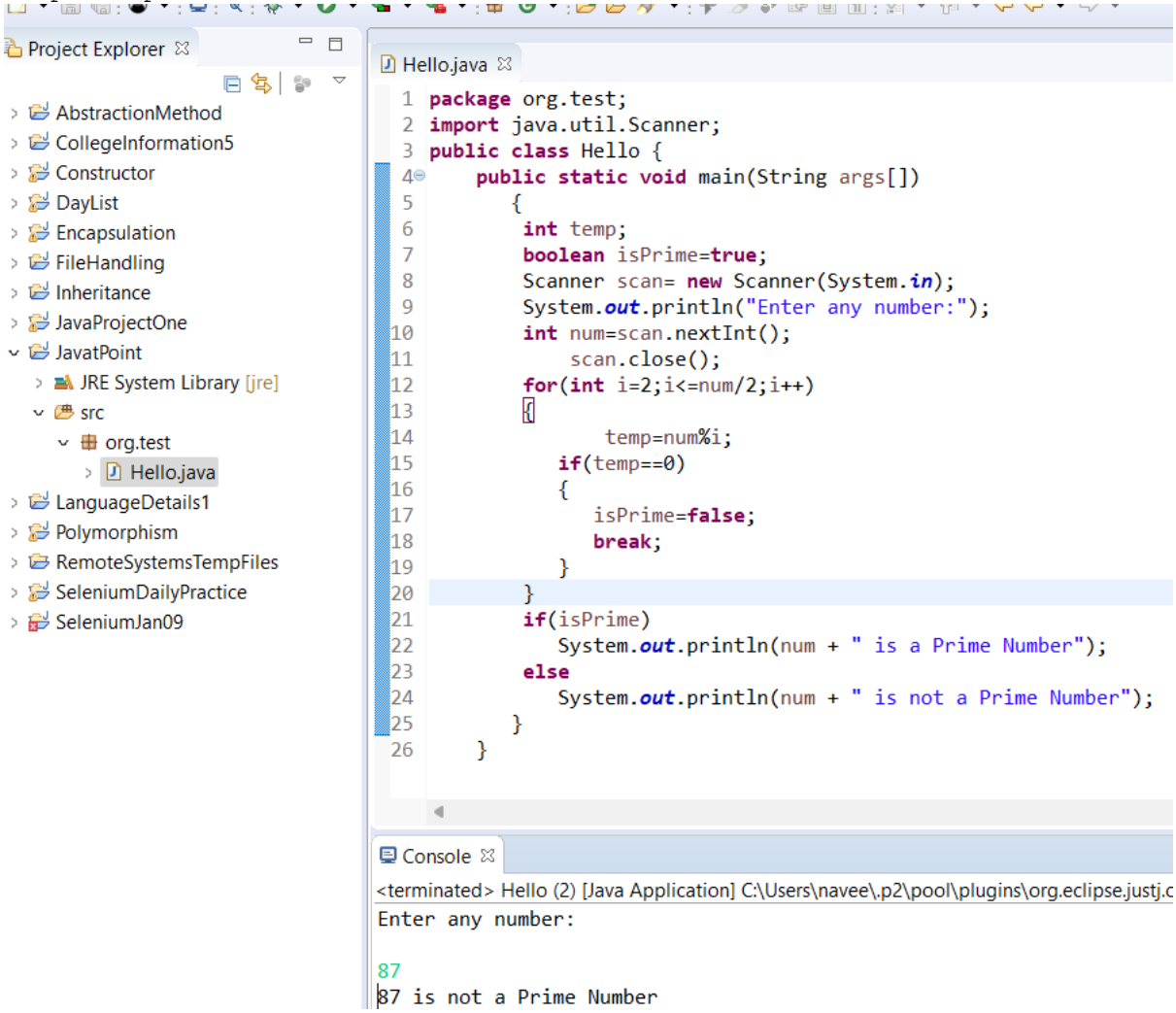


QUESTION 8:

Description: Find prime number or not.

Example:

Input = 11
Output = prime number



QUESTION 9:

Description : Print the below patterns using for loop.

Output:

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7
-----------------------
          *
        *   *
      *   *   *
    *   *   *   *
  *   *   *   *   *
-------------------------
```

Test.java

```java
package org.test;


public class Test {
    public static void main(String[] args)
    {

        for (int i = 1; i <= 7; i++)
        {
            for (int j = 1; j <= i; j++)
            {
                System.out.print(j+" ");
            }

            System.out.println();
        }
    }
}
```

Console

<terminated> Test

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7
```

Test.java

```java
package org.test;


public class Test {
    public static void main(String[] args)
    {

            int rows = 5;

            for (int i = 1; i <= rows; ++i) {
              for (int j = 1; j <= i; ++j) {
                System.out.print("* ");
              }
              System.out.println();
            }
          }
}
```

Console

<terminated> Tes

```
*
* *
* * *
* * * *
* * * * *
```

*Test.java

```java
package org.test;


public class Test {
    public static void main(String[] args) {
    for (int i = 0; i < 5; i++) {
        for (int j = 0; j < 5 - i; j++)
        {
            System.out.print(" ");
            }
        for (int k = 0; k <= i; k++)
        {
            System.out.print("* ");
            }
        System.out.println();
        }
    }
}
```

Console

<terminated> Te

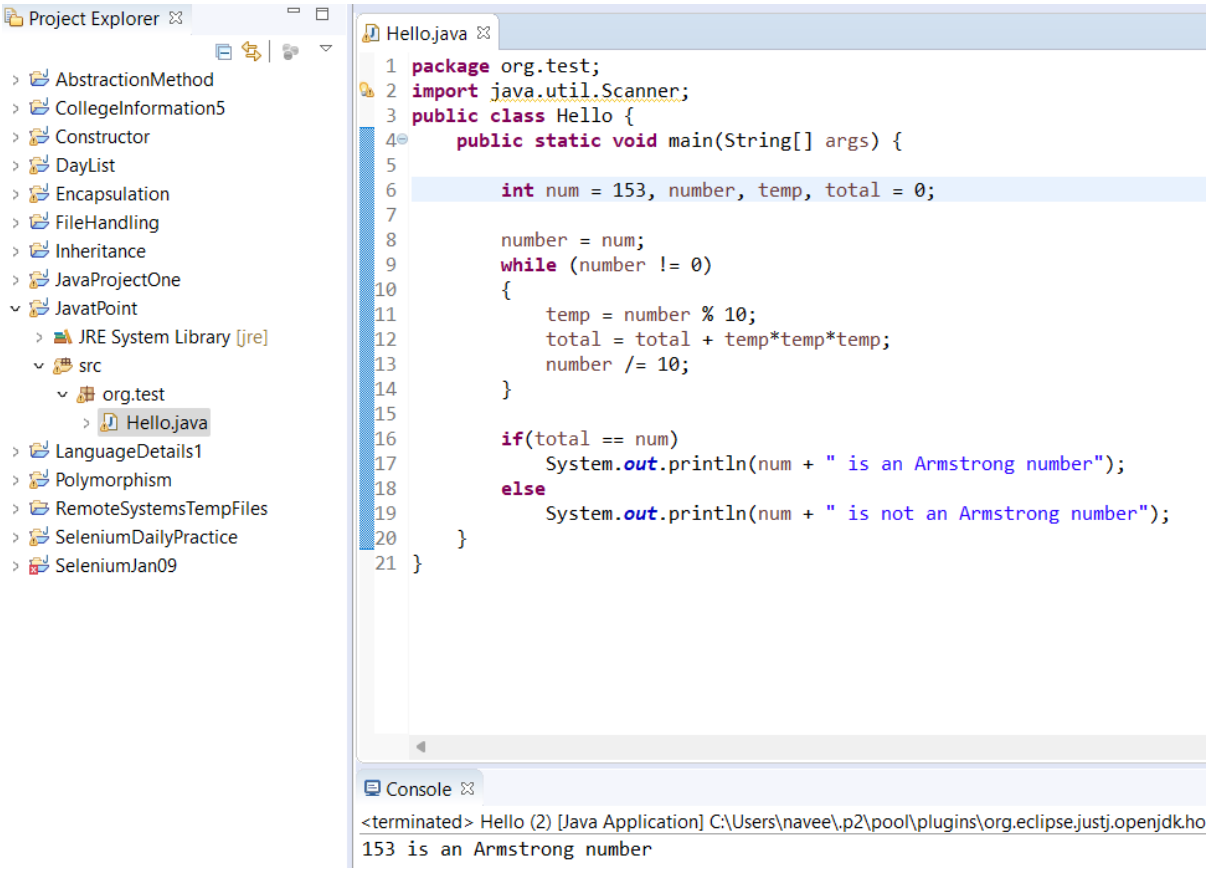
```
    *
   * *
  * * *
 * * * *
* * * * *
```

QUESTION 10:

Description: Find Amstrong number or not

Example:

Input = 153
Output = Amstrong number

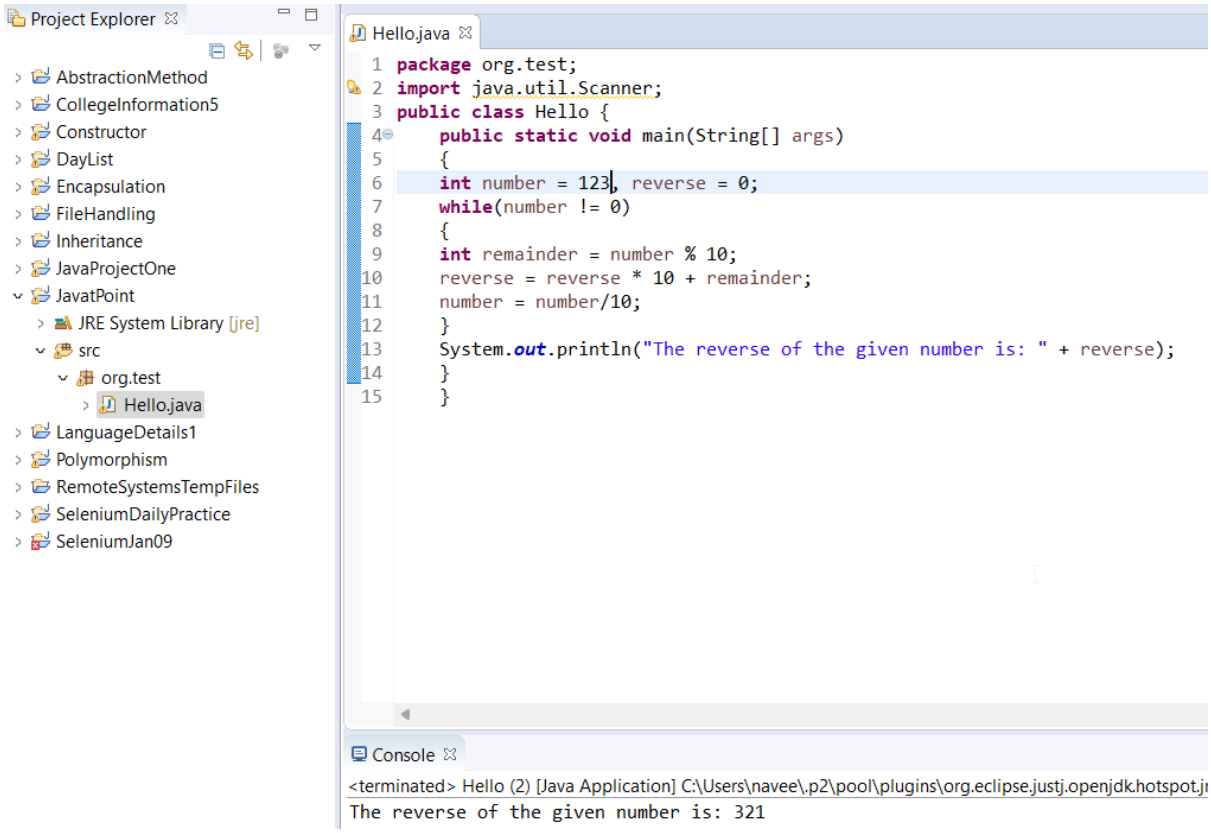


QUESTION 11:

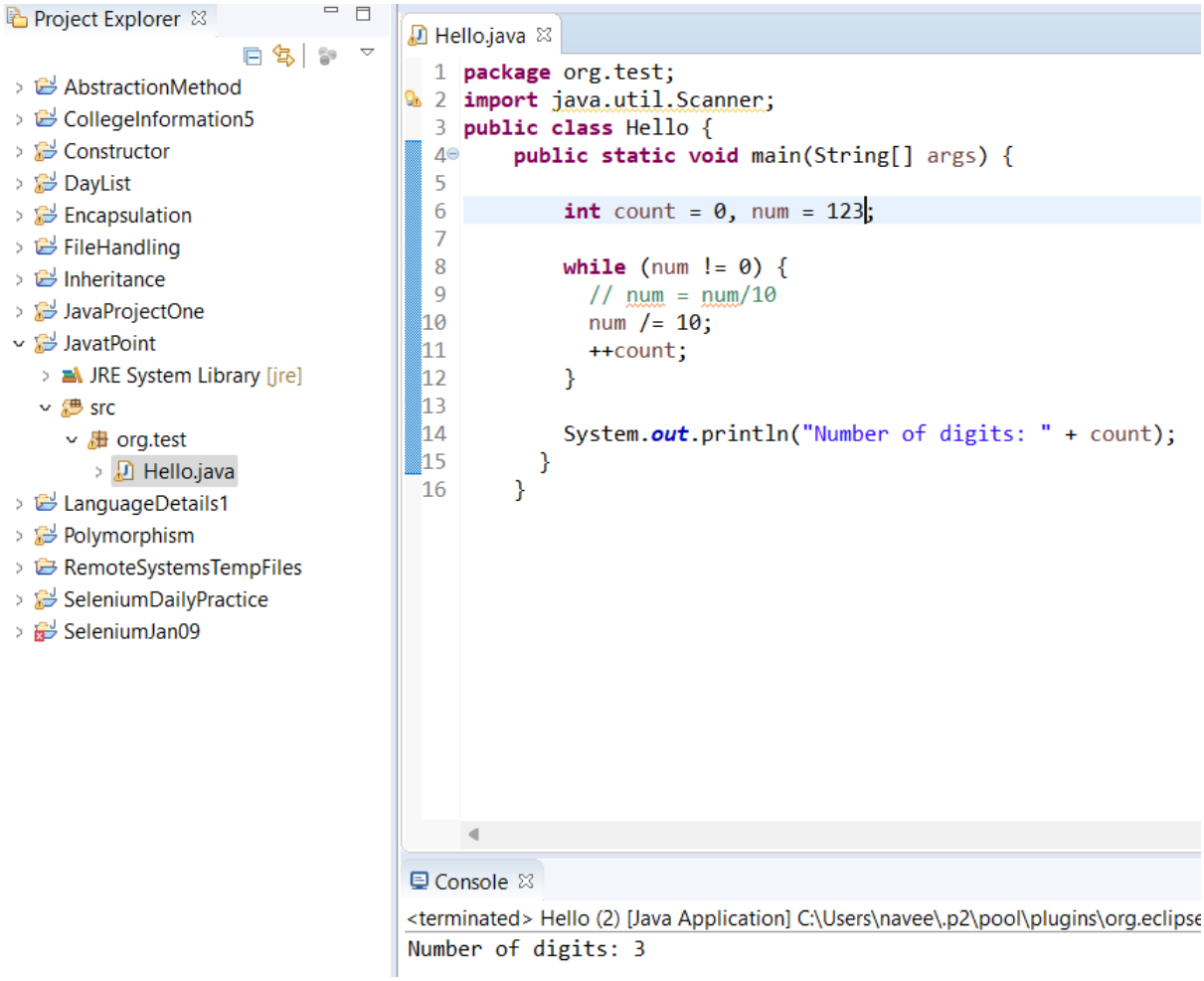
Description: Reverse the number

Example:

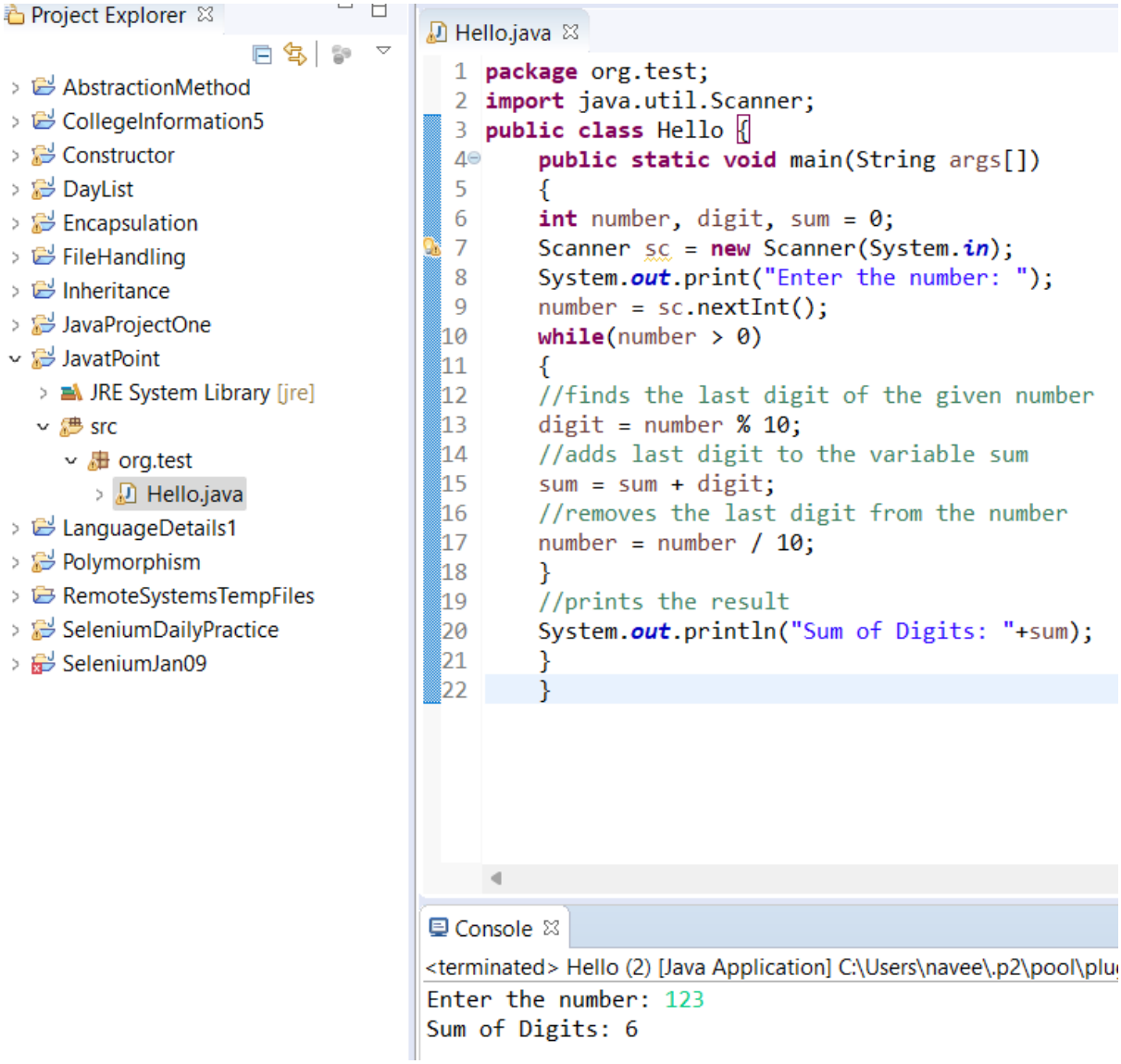
Input = 123
Output = 321




```
QUESTION 12:
-------------
Description: Count of the number

Example:
--------
Input  = 123
Output = 3
```




```
QUESTION 13:
-------------
Description: Sum of the number

Example:
--------
Input  = 123
Output = 6
```

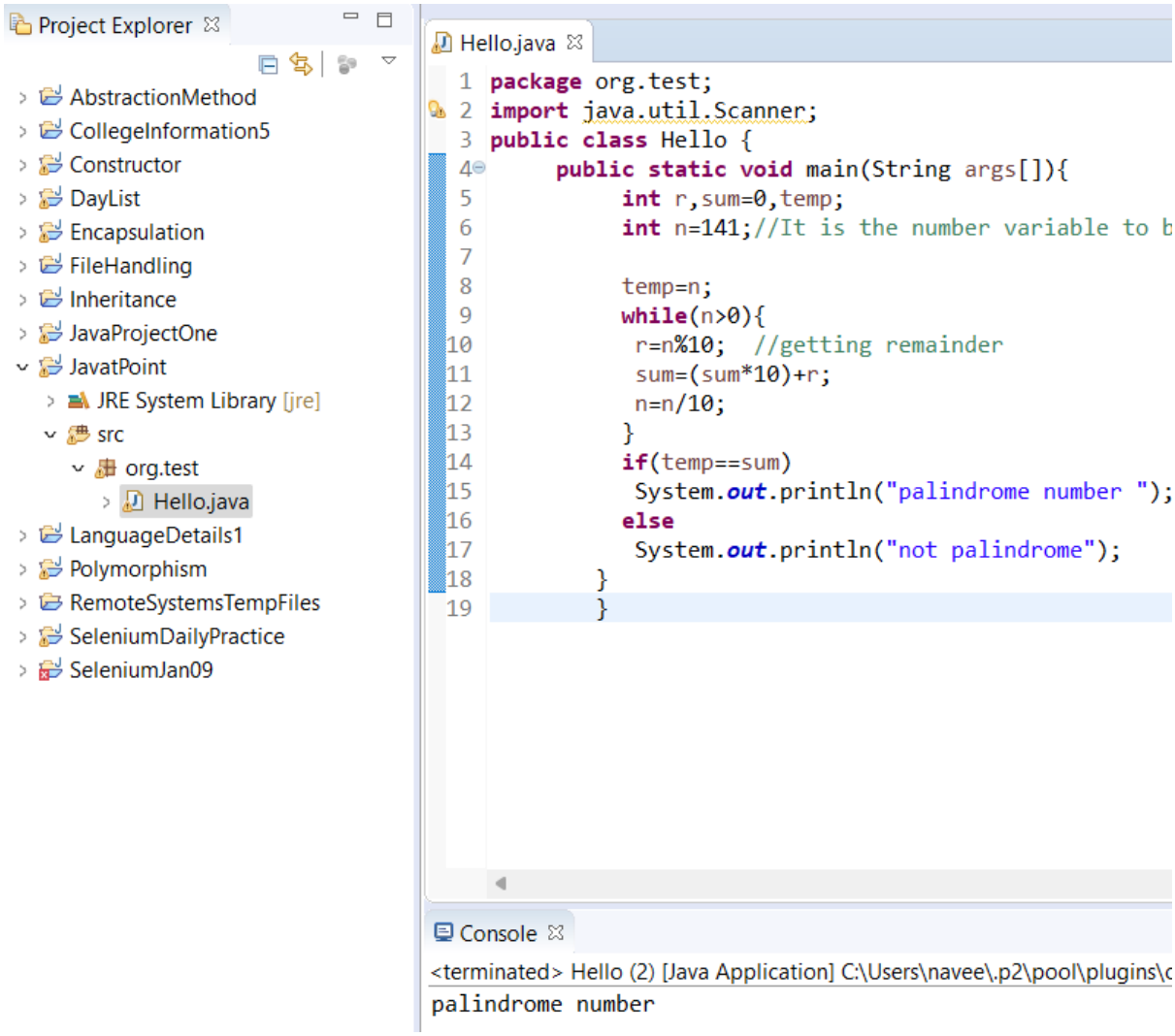


QUESTION 14:

Description: Verify the number is palindrome number not

Example:

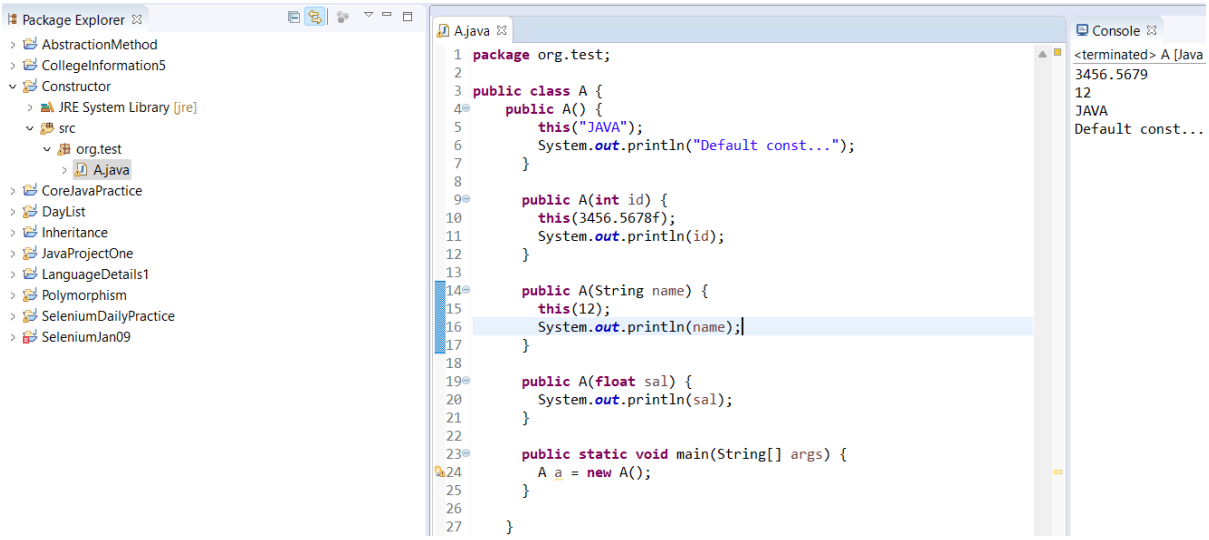
Input = 141
Output = Palindrome


Project Explorer
AbstractionMethod
CollegeInformation5
Constructor
DayList
Encapsulation
FileHandling
Inheritance
JavaProjectOne
JavatPoint
JRE System Library [jre]
src
org.test
Hello.java
LanguageDetails1
Polymorphism
RemoteSystemsTempFiles
SeleniumDailyPractice
SeleniumJan09
Hello.java
1 package org.test;
2 import java.util.Scanner;
3 public class Hello {
4 public static void main(String args[]){
5 int r,sum=0,temp;
6 int n=141;//It is the number variable to b
7
8 temp=n;
9 while(n>0){
10 r=n%10; //getting remainder
11 sum=(sum*10)+r;
12 n=n/10;
13 }
14 if(temp==sum)
15 System.out.println("palindrome number ");
16 else
17 System.out.println("not palindrome");
18 }
19 }
Console
<terminated> Hello (2) [Java Application] C:\Users\navee\.p2\pool\plugins\c
palindrome number

QUESTIONS(Find the below Output)

QUESTION 1:

```
package org.test;

public class A {
  public A() {
    this("JAVA");
    System.out.println("Default const...");
  }

  public A(int id) {
    this(3456.5678f);
    System.out.println(id);
  }

  public A(String name) {
    this(12);
    System.out.println(name);
  }

  public A(float sal) {
    System.out.println(sal);
  }

  public static void main(String[] args) {
    A a = new A();
  }

}
```

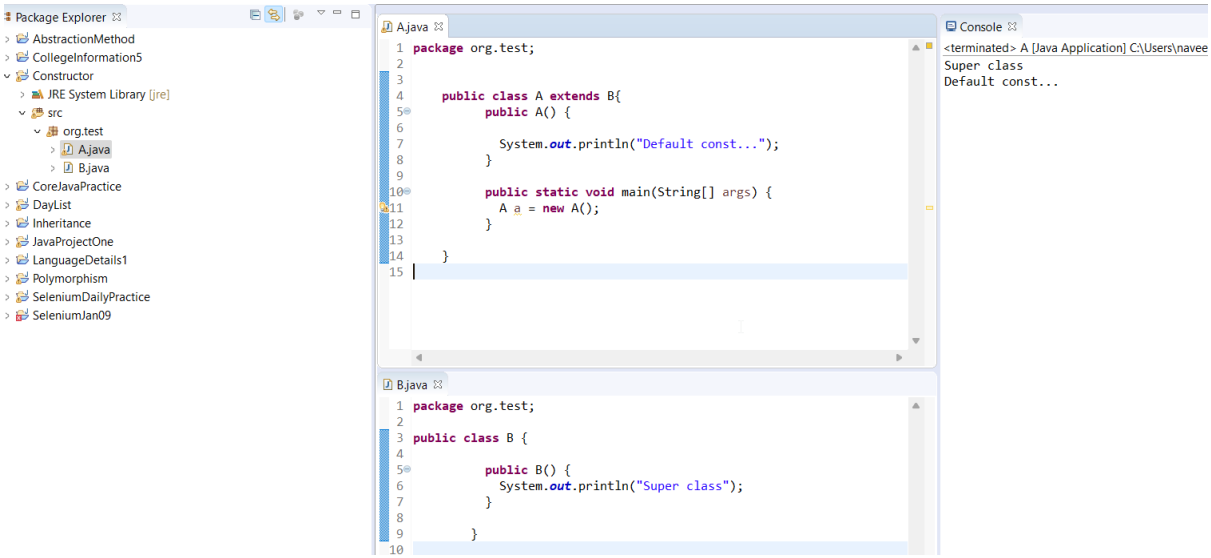


QUESTION 2:

```
package org.test;

public class A extends B{
  public A() {

    System.out.println("Default const...");
```

```
  }

  public static void main(String[] args) {
    A a = new A();
  }

}
```

```
package org.test;

public class B {
  public B() {
    System.out.println("Super class");
  }

}
```



QUESTION 3:

```
package org.test;

public class A extends B{
  public A() {

    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }

}
```

```
package org.test;

public class B {
  public B() {
    System.out.println("Super class");
  }
```

```
  public B(int id) {
    System.out.println(id);
  }

}
```

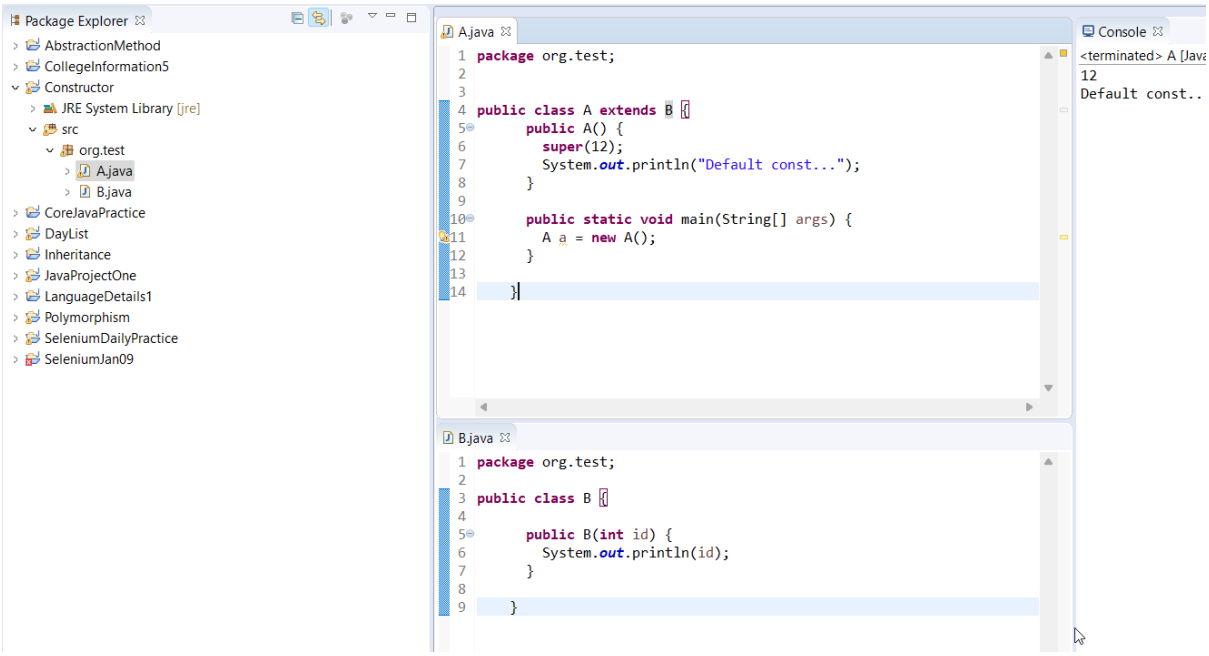
QUESTION 4:

```
package org.test;

public class A extends B {
  public A() {
    super(12);
    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }

}

package org.test;

public class B {
  public B() {
    System.out.println("Super class");
  }

  public B(int id) {
    System.out.println(id);
  }

}
```

QUESTION 5:

```
package org.test;

public class B {

  public B(int id) {
    System.out.println(id);
  }

}
```

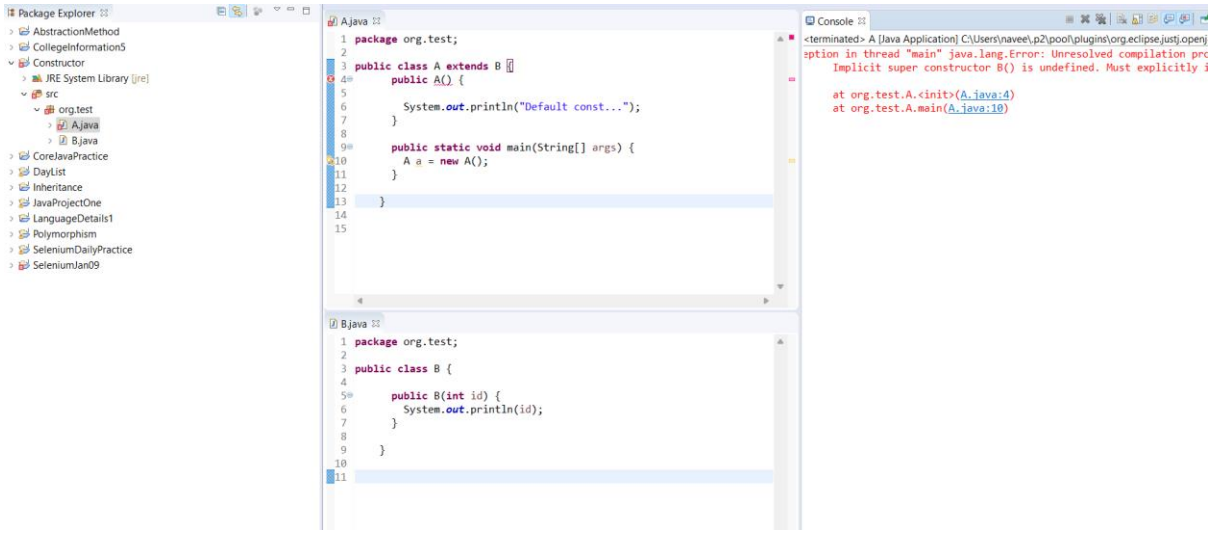
```
package org.test;

public class A extends B {
  public A() {
    super(12);
    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }

}
```



QUESTION 6:

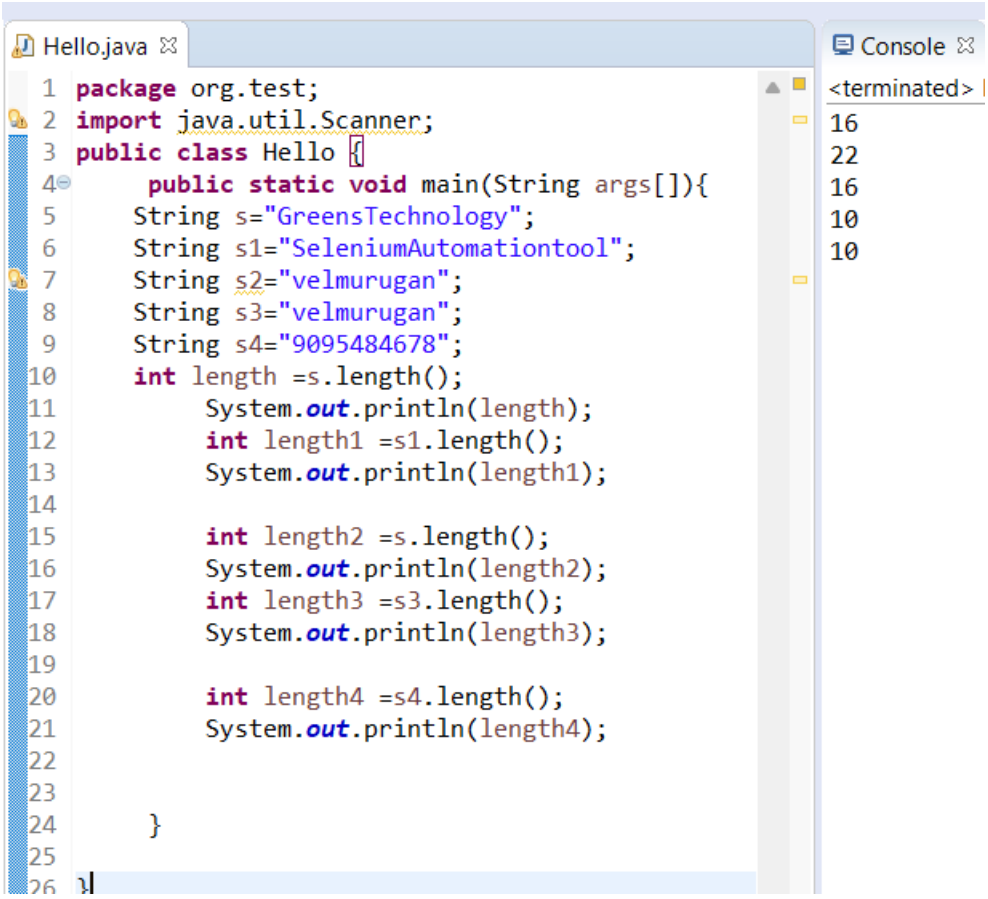
```
package org.test;

public class A extends B {
  public A() {

    System.out.println("Default const...");
  }

  public static void main(String[] args) {
    A a = new A();
  }

}
```

```
package org.test;

public class B {

  public B(int id) {
    System.out.println(id);
  }

}
```


Package Explorer
AbstractionMethod
CollegeInformation5
Constructor
JRE System Library [jre]
src
org.test
A.java
B.java
CoreJavaPractice
DayList
Inheritance
JavaProjectOne
LanguageDetails1
Polymorphism
SeleniumDailyPractice
SeleniumJan09
A.java
1 package org.test;
2
3 public class A extends B {
4 public A() {
5
6 System.out.println("Default const...");
7 }
8
9 public static void main(String[] args) {
10 A a = new A();
11 }
12
13 }
14
15
B.java
1 package org.test;
2
3 public class B {
4
5 public B(int id) {
6 System.out.println(id);
7 }
8
9 }
10
11
Console
<terminated> A [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openj
eption in thread "main" java.lang.Error: Unresolved compilation prc
Implicit super constructor B() is undefined. Must explicitly i
at org.test.A.<init>(A.java:4)
at org.test.A.main(A.java:10)

QUESTION 1:

Description: Find the length of the below string
   String 1: GreensTechnology
   String 2: SeleniumAutomationtool
   String 3: velmurugan
   String 4: j a v a p r o g r a m
   String 5: 9095484678




```java
package org.test;
import java.util.Scanner;
public class Hello {
     public static void main(String args[]){
    String s="GreensTechnology";
    String s1="SeleniumAutomationtool";
    String s2="velmurugan";
    String s3="velmurugan";
    String s4="9095484678";
    int length =s.length();
         System.out.println(length);
         int length1 =s1.length();
         System.out.println(length1);

         int length2 =s.length();
         System.out.println(length2);
         int length3 =s3.length();
         System.out.println(length3);

         int length4 =s4.length();
         System.out.println(length4);


     }

}
```

QUESTION 2:

Description: Find the particular character index in the given string
   String 1: GreensTechnology
             Find the last index of o

   String 2: SeleniumAutomationtool
             Find the index of o

   String 3: Velmurugan
             Find the index of n

   String 4: j a v a p r o g r a m
             Find the last index of (emptyspace)

   String 5: 9095484678
             Find the index of 8

Hello.java

```java
package org.test;
import java.util.Scanner;
public class Hello {
     public static void main(String args[]){
    String s="GreensTechnology";
    String s1="SeleniumAutomationtool";
    String s2="velmurugan";
    String s3="j a v a p r o g r a m";
    String s4="9095484678";
         int length =s.lastIndexOf('o');
         System.out.println(length);

         int length1 =s1.indexOf('o');
         System.out.println(length1);

         int length2 =s2.indexOf('n');
         System.out.println(length2);

         int length3 =s3.lastIndexOf(' ');
         System.out.println(length3);

         int length4 =s4.indexOf('8');
         System.out.println(length4);


     }

}
```

Console

<terminated> H

```
13
11
9
19
5
```

Question 3:

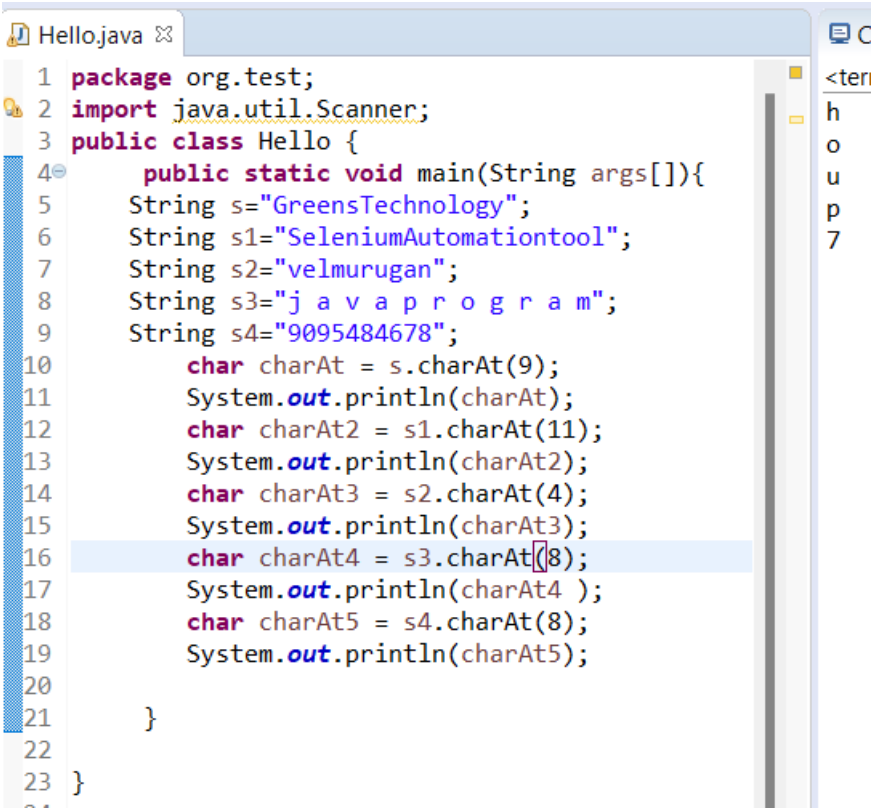
Description: Find the particular character in the given string
   String 1: GreensTechnology
             print the character h

   String 2: SeleniumAutomationtool
             print the character o

   String 3: velmurugan
             print the character u

   String 4: j a v a p r o g r a m
             print the character p

   String 5: 9095484678
             print the character 7



QUESTION 4:

Description : Get two input from user and check the equality
: print in the output whether it is Equal or not
Example:

Input :
String 1 : Java
String 2 : Java

Example:

Input :
String 1 : Java
String 2 : java

Example:

Input :
String 1 : Green Technology
String 2 : GreenTechnology.

Example(use equalsIgnoreCase):

Input :
String 1 : Java
String 2 : java

Example(use equalsIgnoreCase) :

Input :
String 1 : Nisha
String 2 : nisha

Hello.java

```java
package org.test;
import java.util.Scanner;
public class Hello {
     public static void main(String args[]){
    String s="Java";
    String s1="Java";
    boolean equals = s.equals(s1);
    System.out.println(equals);
    String s2="Java";
    String s3="java";
    boolean equals2 = s2.equals(s3);
    System.out.println(equals2);
    String s4="Green Technology";
    String s5="GreenTechnology  ";
    boolean equals3 = s4.equals(s5);
    System.out.println(equals3);
    String s6="Java";
    String s7="java";
    boolean equalsIgnoreCase = s6.equalsIgnoreCase(s7);
    System.out.println(equalsIgnoreCase);
    String s8="Java";
    String s9="java";
    boolean equalsIgnoreCase2 = s8.equalsIgnoreCase(s9);
    System.out.println(equalsIgnoreCase2);
     }

}
```

Console

<terminat

```
true
false
false
true
true
```

QUESTION 5:

QUESTION 5.1:

Description: Get the email id from the user and verify '@' is present or not?

Example:

Input = velmurugank451@gmail.com
Output = valid email id

QUESTION 5.2:

Description:Get the address from the user and verify "pincode" is present or not?

Example:

Input = 5-35-2a,venkatesh nivas,Aruppukottai
Output = invalid address

QUESTION 5.3:

Description:Get the email from the user and verify '@' is present or not and return true or false?

Example:

Input = Nishakerala24@gmail.com
Output = True/False

QUESTION 5.4:

Description:Get the phonenumber from the user and verify any character is present or not .
If character is present return invalid number

Example:

Input = 90954a6o78
Output = False

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
        String s="velmurugank451@gmail.com";
        boolean contains = s.contains("@");
        if(contains == true) {
            System.out.println("valid email id");
        }
        String s1="5-35-2a,venkatesh nivas,Aruppukottai";
        boolean contains2 = s1.contains("pincode");
        if(contains2 == false) {
            System.out.println("invalid address");
        }
        String s2="Nishakerala24@gmail.com";
        boolean contains3 = s2.contains("@");
        System.out.println(contains3);
        String s4="90954a6o78";
        boolean matches = s4.matches("%[a-zA-Z]%");
        System.out.println(matches);

    }

}
```

Console

```
<terminated> HellloJ
valid email id
invalid address
true
false
```

QUESTION 6:

Description:Get the phonenumber from the user .
If phonenumber exceeds greater than 10 then return invalid number

Example:

Input = 89034256972365
output = invalid
Example 2:

Input = 9095484678

Output = valid

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
        String s="89034256972365";
        int length = s.length();
        if(length>10) {
            System.out.println("invalid");
        }
        String s2="9095484678";
        int length2 = s2.length();
        if(length2 == 10) {
            System.out.println("valid id");
        }

    }

}
```

Console

```
<terminated:
invalid
valid id
```

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
        String s="89034256972365";
        int length = s.length();
        if(length>10) {
            System.out.println("invalid");
        }
        String s2="9095484678";
        int length2 = s2.length();
        if(length2 == 10) {
            System.out.println("valid id");
        }

    }

}
```

Console

```
<terminated:
invalid
valid id
```

QUESTION 7:

QUESTION 7.1:

Description:Given string as "Welcome to java class" and replace java into sql.

Example:

Input = Welcome to class java
output = Welcome to class sql

QUESTION 7.2:

Description:Given string as "Greens Adayar"and replace Adayar into Omr.

Example:

Input = Greens Adayar
Output = Greens Omr

QUESTION 7.3:

Description: Given String as "Welcome to java class" and Replace space into '#'

Example:
---
input:Welcome to java class
output:Welcome#to#java#class

QUESTION 7.4:

Description:Get the email from the user and verify "gmail" is present or not.
If present replace that gmail into yahoo

Example:

Input = Nishakerala24@gmail.com
Output = Nishakerala24@yahoo.com

QUESTION 7.5:

Description:Get the address from the user and verify "pincode" is present or not.
If present replce the pincode with empty space

Example:

Input = 5-35-2a,venkatesh nivas,Aruppukottai,pincode-626101
Output = 5-35-2a,venkatesh nivas,Aruppukottai

HelllloJava.java

```java
package org.test;

public class HelllloJava {
    public static void main(String[] args) {
        String s="Welcome to java class";
        String replace = s.replace("java", "sql");
        System.out.println(replace);

        String s1="Greens Adayar";
        String replace2 = s1.replace("Adayar", "Omr");
        System.out.println(replace2);

        String s2="Welcome to java class";
        String replace3 = s2.replace(" ", "#");
        System.out.println(replace3);

        String s3="Nishakerala24@gmail.com";
        String replace4 = s3.replace("gmail", "yahoo");
        System.out.println(replace4);
        String s4="5-35-2a,venkatesh nivas,Aruppukottai,pincode-626101";
        String replace5 = s4.replace("5-35-2a,venkatesh nivas,Aruppukottai,pincode-626101", "5-35-2a,venkatesh nivas,Aruppukottai");
        System.out.println(replace5);
    }

}
```

Console

```
<terminated> HelllloJava [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe (27-Jan-2022, 4
Welcome to sql class
Greens Omr
Welcome#to#java#class
Nishakerala24@yahoo.com
5-35-2a,venkatesh nivas,Aruppukottai
```

QUESTION 8

QUESTION 8.1

Description:Get the input from the user and print that word in lowercase

Example:

Input = NISHANTHI
Output = nishanthi

QUESTION 8.2

Description:Get the input from the user and print that word in Uppercase

Example:

Input = nishanthi
Output = NISHANTHI

QUESTION 8.3

Description:Convert all small letter and into capital letter

Example:

Input = WelcomE
Output = wELCOMe

QUESTION 8.4

Description:Find the number of uppercase count and lowercase count in the given String

Example:

Input = WelComeToJava
Output:

UpperCase=4
LowerCase=9

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
        String s="NISHANTHI";
        String replace = s.toLowerCase();
        System.out.println(replace);

        String s1="nishanthi";
        String replace2 = s1.toLowerCase();
        System.out.println(replace2);

        String s2="WelcomE";
        String replace3 = s2.substring(0,1).toLowerCase()+s2.substring(1, 6).toUpperCase()+s2.substring(6).toLowerCase();
        System.out.println(replace3);

        String s3="WelComeToJava";
        int upper = 0,lower=0;
        for(int i=0;i<s3.length();i++) {
            char ch=s3.charAt(i);
            if(ch>='A' && ch<='Z') {
                upper++;
            }
            else if(ch>='a' && ch<='z') {
                lower++;
            }
        }
        System.out.println("uppercasecount " +upper);
        System.out.println("lowercasecount:" +lower);
    }

}
```

Console

```
<terminated> HellloJava [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-1657\jre\bin\javaw.exe (27-
nishanthi
nishanthi
wELCOMe
uppercasecount 4
lowercasecount:9
```

QUESTION 9

QUESTION 9.1

Description: Given String as "Welcome to java class" and verify whether the given string startsWith welcome

Example:

Input = Welcome to class java
output = True

QUESTION 9.2

Description: Given String as "Hai i am nisha" and verify whether the given string startsWith welcome

Example:

```
-----------
Input  = Hai i am nisha
output = False
```

QUESTION 9.3

```
--------------
```

Description: Given String as "Welcome to java class" and verify whether the given string endsWith class

```
Example:
-----------
Input  = Welcome to java class
output = True
```

QUESTION 9.4

```
--------------
```

Description: Given String as "Welcome to java class" and verify whether the given string endsWith java

```
Example:
-----------
Input  = Welcome to java class
output = False
```

QUESTION 9.5

```
--------------
```

Description: Given String as "Welcome to java class" and verify whether the string is empty or not

```
Example:
-----------
Input  = Welcome to java class
output = False
```

QUESTION 9.6

```
--------------
```

Description: Given String as "" and verify whether the string is empty or not

```
Example:
-----------
Input  = ""
Output = False
```

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
        String s="Welcome to class java";
        boolean startsWith = s.startsWith("Welcome ");
        System.out.println(startsWith);

        String s1=" Hai i am nisha";
        boolean startsWith2 = s1.startsWith("Welcome");
        System.out.println(startsWith2);

        String s2="Welcome to class java";
        boolean endsWith = s2.endsWith("java");
        System.out.println(endsWith);

        String s3="Welcome to class java";
        boolean endsWith2 = s3.endsWith("class");
        System.out.println(endsWith2);

        String s4="Welcome to class java";
        boolean empty = s4.isEmpty();
        System.out.println(empty);

        String s5="";
        boolean empty2 = s5.isEmpty();
        System.out.println(empty2);


    }

}
```

Console

```
<terminated>
true
false
true
false
false
true
```

QUESTION 10

Description : Get two input from user and Compare

Example

String 1 : Nisha
String 2 : nisha

Example

String 1 : Nia
String 2 : nisha

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
        String s1 ="Nisha";
        String s2= "nisha";
        boolean equals = s1.equals(s2);
        System.out.println(equals );

        String s3  ="Nia";
        String s4="nisha";
boolean equals2 = s3.equals(s4);
System.out.println(equals2 );

    }

}
```

Console

```
<terminated
false
false
```

QUESTION 11

QUESTION 11.1

Description : Generate the two literal string and find the identityHashCose()

Example

String 1 : Nisha
String 2 : Nisha

QUESTION 11.2

Description : Generate the two non literal string and find the identityHashCose()

Example

String 1 : Nisha
String 2 : Nisha

QUESTION 11.3

Description : Generate the three non literal string and find the identityHashCose()

Example

String 1 : Nisha
String 2 : Rengan
String 3 : NishaRengan

QUESTION 11.4

Description : Generate the three literal string and find the identityHashCose()

Example

String 1 : Nisha
String 2 : Rengan

String 3 : NishaRengan

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
        String s1 ="Nisha";
        String s2= "nisha";
        int identityHashCode = System.identityHashCode(s1);
        System.out.println(identityHashCode);
        int identityHashCode2 = System.identityHashCode(s2);
        System.out.println(identityHashCode2);

        String s3  ="Nisha";
        String s4="Nisha";
        int identityHashCode3 = System.identityHashCode(s3);
        System.out.println(identityHashCode3);
        int identityHashCode4 = System.identityHashCode(s4);
        System.out.println(identityHashCode4);


    }

}
```

Console

```
<terminated> He
1279149968
2096057945
1279149968
1279149968
```

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
        //String s1 ="Nisha";
        //String s2= "Rengan";
        //String s3  ="NishaRengan";
        System.out.println("non literal string");
        String s1=new String("Nisha");
        String s2=new String("Rengan");
        String s3=new String("NishaRengan");
        int identityHashCode = System.identityHashCode(s1);
        System.out.println(identityHashCode);
        int identityHashCode2 = System.identityHashCode(s2);
        System.out.println(identityHashCode2);
        int identityHashCode3 = System.identityHashCode(s3);
        System.out.println(identityHashCode3);


        System.out.println("literal string");
        String s4 ="Nisha";
        String s5= "Rengan";
        String s6  ="NishaRengan";
        int identityHashCode4 = System.identityHashCode(s4);
        System.out.println(identityHashCode4);
        int identityHashCode5 = System.identityHashCode(s5);
        System.out.println(identityHashCode5);
        int identityHashCode6 = System.identityHashCode(s6);
        System.out.println(identityHashCode6);
    }


}
```

Console

```
<terminated> HellloJava [
non literal string
434176574
2096057945
1689843956
literal string
766572210
1020391880
977993101
```

QUESTION 12

QUESTION 12.1

Description: Given String as "Welcome to java class" and split it by space.

Example:

Input :Welcome to java class
Output:

Welcome
to
java
class

QUESTION 12.2

Description: Given String as "Welcome to java class" and split it by l

Example:

Input :Welcome to java class
Output:

We
come to java c
ass

HelIloJava.java

```
1  package org.test;
2
3  public class HellloJava {
4      public static void main(String[] args) {
5
6          String s1="Welcome to java class";
7          String[] splits1=s1.split(" ");
8          for (String a : splits1) {
9              System.out.println(a);
10         }
11         System.out.println("--------------------------------------");
12         String[] s2=s1.split("1");
13         for (String b : s2) {
14             System.out.println(b);
15         }
16         }
```

Console

<terminated> HellloJava [Java Application] C:\Users\nav

```
Welcome
to
java
class
--------------------------------------
Welcome to java class
```

QUESTION 13

QUESTION 13.1

Description: Given String as "Welcome to java class" and generate a substring.

Example:

Input :Welcome to java class
Output:

Welcome

HelllloJava.java

```java
package org.test;

public class HelllloJava {
    public static void main(String[] args) {

        String s1="Welcome to java class";
        String substring = s1.substring(0, 8);
        System.out.println(substring );
    }
}
```

Console

```
<terminated> H
Welcome
```

QUESTION 14

Example:

Description: Given String as "Welcome" and the number of consonant count and vowels count

Example:

Input = Welcome
output:
---
vowels   = 3
consonant = 4

HelllloJava.java

```java
package org.test;

public class HelllloJava {
    public static void main(String[] args) {

        String s1="Welcome";
        int vowels=0,consonants=0;
        s1 =s1.toLowerCase();
        for (int i = 0; i < s1.length(); i++) {
        char ch=s1.charAt(i);
        if(ch == 'a' || ch == 'e' || ch == 'i'||ch == 'o' || ch == 'u'  ) {
            vowels++;
        }
        else if ((ch >= 'a' && ch <= 'z'))
        {
            ++consonants;
    }
        }
        System.out.println("vowels:" +vowels);
        System.out.println("consonants:" +consonants);

    }
}
```

Console

```
<terminated> HelllloJava [J
vowels:3
consonants:4
```

QUESTION 15:

Description: Find the count of caps,small,number and special character in given string

Example:

Input : Welcome To Java class @123
Output
---
caps count :3
small count :15
number count:3
Special char:5

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {

        String s1="Welcome To Java class @123";
        int upper=0,lower=0,number=0,specialChar=0,space=0;
        char ch;
        for (int i = 0; i < s1.length(); i++) {
        ch =s1.charAt(i);
        if(ch>='A' && ch<='Z') {
            upper++;
        }
        else if (ch>='a' && ch<='z') {
            lower++;
        }
        else if(ch>='0' && ch<='9') {
            number++;
        }
        else if(ch== ' ') {
            space++;
        }
        else {
            specialChar++;
        }
    }

        System.out.println("the no of capital letters is:"+upper);
        System.out.println("the no of small letters is:"+lower);
        System.out.println("the no of numerical letters is:"+number);
        System.out.println("the no of specialchar is:"+specialChar);
}
}
```

Console

```
<terminated> HellloJava [Java Application] C:\
the no of capital letters is:3
the no of small letters is:15
the no of numerical letters is:3
the no of specialchar is:1
```

QUESTION 16

Description: Replace all vowels char into '@'

Example:

Input = Welcome
Output = W@lc@m@

HellloJava.java

```java
package org.test;

public class HellloJava {
    public static void main(String[] args) {
    String s="Welcome";
    System.out.println(s);
    char ch[]=s.toCharArray();
    for(int i =0;i<ch.length;i++) {
        if(ch[i] == 'a' ||ch[i] == 'e'||  ch[i] == 'i'|| ch[i] == 'o'|| ch[i] == 'u') {
            ch[i] ='@';
        }
    }
    for (int i = 0; i < ch.length; i++) {
        System.out.print(ch[i]);
    }

    }

}
```

Console

```
<terminated> H
Welcome
W@lc@m@
```

DAY7:

1.Arrays
2.Collections Introduction
3.List(ArrayList only)
QUESTIONS(Find the below Output)

QUESTION 1:

Description : Write a Java program to sum values of an array
        Input a[]= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
        Output   = 55

CourseDetails.java

```java
package org.tcs.com;

import java.util.HashMap;
public class CourseDetails {
    public static void main(String[] args) {
         int [] arr = new int [] {1, 2, 3, 4, 5,6,7,8,9,10};
            int sum = 0;
            //Loop through the array to calculate sum of elements
            for (int i = 0; i < arr.length; i++) {
               sum = sum + arr[i];
            }
            System.out.println("Sum of all the elements of an array: " + sum);
        }
    }
```

Console

```
<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.j
Sum of all the elements of an array: 55
```

QUESTION 2:

Description : Write a Java program to calculate the average value of array elements.
        Input a[]= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
        Output   = average = 7.0

CourseDetails.java

```java
package org.tcs.com;

public class CourseDetails {
    public static void main(String[] args) {
        int[] numbers = new int[]{1,2,3,4,5,6,7,8,9,10};


         double sum = 0;

         for(int i=0; i<numbers.length ; i++)
         sum = sum + numbers[i];


         double average = sum / numbers.length;

         System.out.println("Average value of array elements is : " + average);
         }
        }
```

Console

```
<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotsp
Average value of array elements is : 5.5
```

QUESTION 3:

Description : Write a Java program to remove duplicates from array
Input a[]= {10,10,20,50,60,80,60,50}
Output a[]= {10,20,50,60,80}

Test.java

```java
package org.test;

import java.util.ArrayList;

public class Test {//Input  a[]= {10,10,20,50,60,80,60,50}
    public static void main(String[] args) {
    ArrayList a =new ArrayList();
        a.add(10);
        a.add(10);
        a.add(20);
        a.add(50);
        a.add(60);
        a.add(80);
        a.add(60);
        a.add(50);
        int length=a.size();
        for(int i=0;i<length;i++) {
            for(int j=i+1;j<length;j++) {
                if(a.get(i) == a.get(j)) {
                    a.remove(i);
                    i=0;
                    j=0;
                    length=length-1;
                }
            }
        }
        System.out.println(a);

    }
}
```

Console

```
<terminated> Test (6) [Java A
[10, 20, 80, 60, 50]
```

QUESTION 4:

QUESTION 4.1:

Description : Create a new ArrayListlist with values and find the length of it

Input : List = 10,20,30,90,10,10,40,50

QUESTION 4.2:

Description : Create a new LinkedListlist with values and find the length of it

Input : List = 100,200,300,400,500,600,700

QUESTION 4.3:

Description : Create a new vector with values and find the length of it

Input : List = 105,205,305,405,505,605,705,805

QUESTION 4.4:

Description : Create a new LinkedListlist with values and find the size() of it.

Input : List = 100,200,300,400,500,600,700

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Vector;

public class Test {//105,205,305,405,505,605,705,805
    public static void main(String[] args) {
    Vector <Integer> a =new Vector<Integer>();
        a.add(105);
        a.add(205);
        a.add(305);
        a.add(405);
        a.add(505);
        a.add(605);
        a.add(705);
        a.add(805);
        int length=a.size();
        System.out.println(length);
    }
}
```

Console

<terminated> T

8

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;
import java.util.Vector;

public class Test {//105,205,305,405,505,605,705,805
    public static void main(String[] args) {
    List <Integer> a =new LinkedList();
        a.add(100);
        a.add(200);
        a.add(300);
        a.add(400);
        a.add(500);
        a.add(600);
        a.add(700);
        int length=a.size();
        System.out.println(length);
    }
}
```

Console

<terminated> Test

7

Test.java

```java
package org.test;

import java.util.ArrayList;

public class Test {//10,20,30,90,10,10,40,50
    public static void main(String[] args) {
    ArrayList a =new ArrayList();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(90);
        a.add(10);
        a.add(10);
        a.add(40);
        a.add(50);
        int length=a.size();
        System.out.println(length);
    }
}
```

Console

```
<terminated> Tes
8
```

List1.java

```java
package org.com;

import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;


public class List1 {
public static void main(String[] args) {


    List<Integer>l = new LinkedList();
    l.add(100);
    l.add(200);
    l.add(300);
    l.add(400);
    l.add(500);
    l.add(600);
    l.add(700);
    int size = l.size();
    System.out.println(size);

}
}
```

Console

```
<terminated> L
7
```

QUESTION 5:

QUESTION 5.1:

Description : Get the first index value of 10
              Input: List = 10,20,30,90

QUESTION 5.2:

Description : Get the last index value of 10
              Input: List = 10,20,30,90,10,10,40,50

QUESTION 5.3:

```
--------------
Description : Get the index value of 50
              Input:   List = 10,20,30,90,10,10,40,50

QUESTION 5.4:
--------------
Description : Get the index value of 90
              Input:   List = 10,20,30,90,10,10,40,50

QUESTION 5.5:
--------------
Description : Get the each index value of 10 present in below list
              Input:   List = 10,20,30,90,10,10,40,50,10

QUESTION 5.6:
--------------
Description : Get the each index value of 70 present in below list
              Input:   List = 10,20,30,90,10,10,40,50,10
```

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //10,20,30,90,10,10,40,

    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);
    int indexOf =li.indexOf(50);
    System.out.println(indexOf);
    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plugins\org.ecli

7

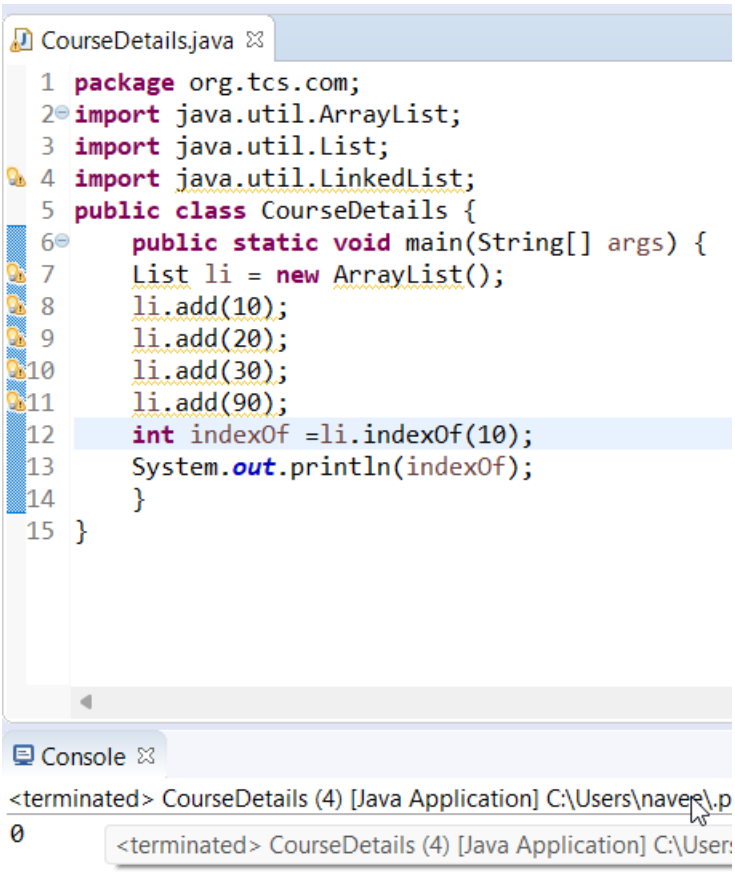
CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //10,20,30,90,10,10,40,50

    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);
    int indexOf =li.indexOf(10);
    System.out.println(indexOf);
    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.ju

```
0
```


CourseDetails.java
1 package org.tcs.com;
2 import java.util.ArrayList;
3 import java.util.List;
4 import java.util.LinkedList;
5 public class CourseDetails {
6 public static void main(String[] args) {
7 List li = new ArrayList();
8 li.add(10);
9 li.add(20);
10 li.add(30);
11 li.add(90);
12 int indexOf =li.indexOf(10);
13 System.out.println(indexOf);
14 }
15 }
Console
<terminated> CourseDetails (4) [Java Application] C:\Users\naver\.p
0
<terminated> CourseDetails (4) [Java Application] C:\User

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //10,20,30,90,10,10,40,50

    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);
    int indexOf =li.indexOf(90);
    System.out.println(indexOf);
    }
}
```

Console

```
<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.j
3
```

```
QUESTION 6:
------------
QUESTION 6.1:
--------------
Description : Get the value present at 2nd index
              Input:   List = 10,20,30,40,50,60

QUESTION 6.2:
--------------
Description : Get the value present at 4th index
              Input:   List = 100,200,300,400,500,600,700

QUESTION 6.3:
--------------
Description : Get the value present at 8th index
              Input:   List = 105,205,305,405,505,605,705,805

QUESTION 6.4:
--------------
Description : Get the each value of list by using normal for loop
```

Input: List = 105,205,305,405,505,605,705,805

QUESTION 6.5:

Description : Get the each value of list by using enhanced for loop
Input: List = 105,205,305,405,505,605,705,805

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //105,


    li.add(105);
    li.add(205);
    li.add(305);
    li.add(405);
    li.add(505);
    li.add(605);
    li.add(705);
    li.add(805);

    for (int i = 0; i < li.size(); i++) {
    System.out.println(li.get(i));
    }
    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p

```
105
205
305
405
505
605
705
805
```

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //105,


    li.add(105);
    li.add(205);
    li.add(305);
    li.add(405);
    li.add(505);
    li.add(605);
    li.add(705);
    li.add(805);
    Object object = li.get(7);
    System.out.println(object);
    }
}

```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p

```
805
```

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //100,200,300,400,500


    li.add(100);
    li.add(200);
    li.add(300);
    li.add(400);
    li.add(500);
    li.add(600);
    li.add(700);

    Object object = li.get(4);
    System.out.println(object);
    }
}
```

Console

```
<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plugins\org.e
500
```

*CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //10,20,30,90,
    
    li.add(10);
    li.add(20);
    li.add(30);
    li.add(40);
    li.add(50);
    li.add(60);

    Object object = li.get(2);
    System.out.println(object);
    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plug

```
30
```

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //105,205,305


    li.add(105);
    li.add(205);
    li.add(305);
    li.add(405);
    li.add(505);
    li.add(605);
    li.add(705);
    li.add(805);

    for (Object x : li) {
    System.out.println(x);
    }
    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plu

```
105
205
305
405
505
605
705
805
```

QUESTION 7:

QUESTION 7.1:

Description : Remove the value present at 2nd index
Input: List = 10,20,30,40,50,60

QUESTION 7.2:

Description : Remove the value present at 10th index
Input: List = 10,20,30,90,10,10,40

QUESTION 7.3:

Description : Remove the last value of 10 present in the list
              Input: List = 10,20,30,90,10,10,40

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          //10,20,3

    li.add(10);
    li.add(20);
    li.add(30);
    li.add(40);
    li.add(50);
    li.add(60);
    li.remove(2);
    System.out.println(li);


    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\po

```
[10, 20, 40, 50, 60]
```

QUESTION 8:

QUESTION 8.1:

Description : Add a value 50 in the 2nd index and display the list after adding.
              Input : List = 10,20,30,90,10,10,40,50

QUESTION 8.2:

Description : Add a value 70 at the end of the list
              Input : List = 10,20,30,90,10,10,40,50

QUESTION 8.3:

Description : Add a value 80 at the 8th index of list

Input : List = 10,20,30,90,10,10,40,50

QUESTION 8.4:

Description : Add a value 100 at the last index of 10 in the list
Input : List = 10,20,30,90,10,10,40,50

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          // 1

    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);
    li.add(8,70);
    System.out.println(li);
    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\

```
[10, 20, 30, 90, 10, 10, 40, 50, 70]
```

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          // 10,2


    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);
    li.add(2,50);
    System.out.println(li);
    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\

```
[10, 20, 50, 30, 90, 10, 10, 40, 50]
```

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          // 10,20,30,


    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);
    li.add(8,80);
    System.out.println(li);
    }
}
```

Console

```
<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\p
[10, 20, 30, 90, 10, 10, 40, 50, 80]
```

```
QUESTION 9:
-----------
QUESTION 9.1:
-------------
Description : Replace the value 300 into 350 in the list
              Input : List = 100,200,300,400,500,600,700

QUESTION 9.2:
--------------
Description : Replace the value present in 7th index as 90
              Input:   List = 10,20,30,90,10,10,40,50,10

QUESTION 9.3:
--------------
Description : Replace the 10 into 100 in List
        Input:   List = 10,20,30,90,10,10,40,50,30
        Output:  List = 100,20,30,90,100,100,40,50,30
```

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;
import java.util.Vector;

public class Test {//10,20,30,90,10,10,40,50,10
    public static void main(String[] args) {
    List <Integer> a =new ArrayList();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(90);
        a.add(10);
        a.add(10);
        a.add(40);
        a.add(50);
        a.add(10);

        a.remove(7);
        a.add(7, 90);
        System.out.println(a);
    }
}
```

Console

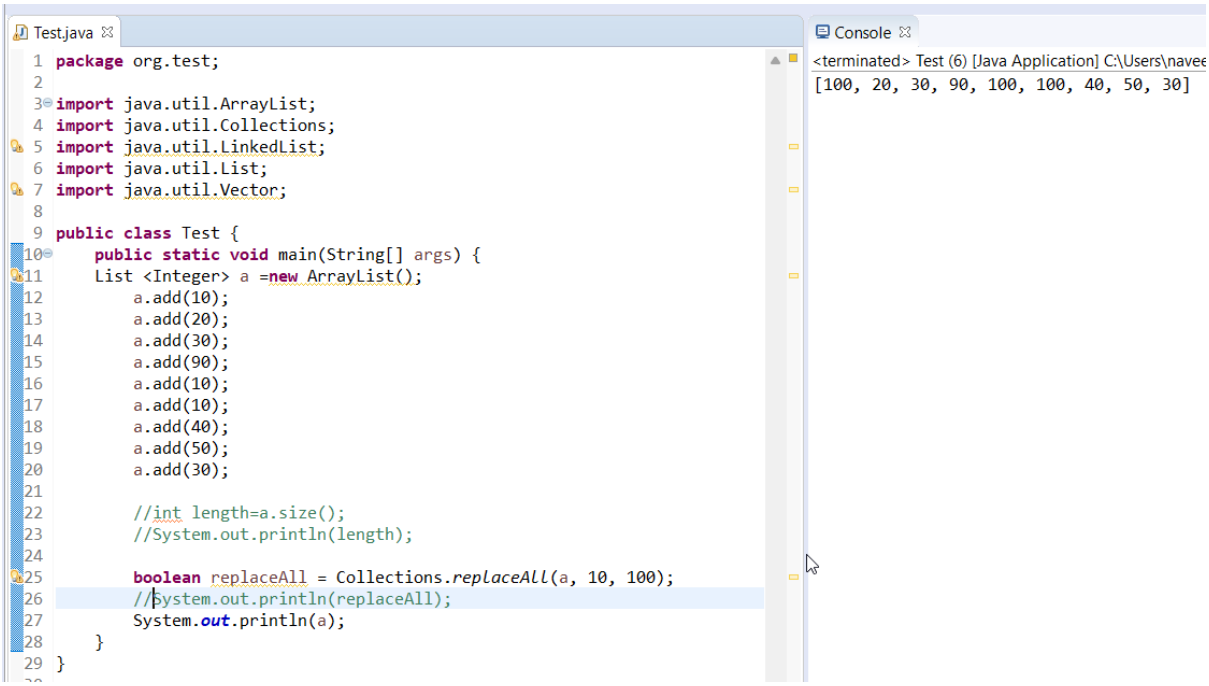
```
<terminated> Test (6) [Java Application] C:\Users\nave
[10, 20, 30, 90, 10, 10, 40, 90, 10]
```

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
    List <Integer> a =new ArrayList();
        a.add(100);
        a.add(200);
        a.add(300);
        a.add(400);
        a.add(500);
        a.add(600);
        a.add(700);
        //int length=a.size();
        //System.out.println(length);
        a.remove(2);
        a.set(2, 350);
        System.out.println(a);
    }
}
```

Console

```
<terminated> Test (6) [Java Application] C:\
[100, 200, 350, 500, 600, 700]
```




```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.LinkedList;
import java.util.List;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
    List <Integer> a =new ArrayList();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(90);
        a.add(10);
        a.add(10);
        a.add(40);
        a.add(50);
        a.add(30);

        //int length=a.size();
        //System.out.println(length);

        boolean replaceAll = Collections.replaceAll(a, 10, 100);
        //System.out.println(replaceAll);
        System.out.println(a);
    }
}
```

QUESTION 10:

QUESTION 10.1:

Description : Create a new ArrayListlists with values and return the common values

Input : List = 10,20,30,90,10,10,40,50
Input : List = 30,40,50,60,80

QUESTION 10.2:

Description : Create a new ArrayListlists with values and return the common values

Input : List = 10,20,30,90,10,10,40,50
Input : List = 10,20,60,50,40,70,80,90

QUESTION 10.3:

Description : Create a new ArrayListlists with values and return the common values

Input : List = 10,20,30,40,50,60,70,80
Input : List = 100,200,300,400,500,600,700,8000

List1.java

```java
package org.com;

import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;


public class List1 {
public static void main(String[] args) {
    List li =new ArrayList();
    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(80);

    List<Integer>l = new LinkedList();
    l.add(10);
    l.add(20);
    l.add(60);
    l.add(50);
    l.add(40);
    l.add(70);
    l.add(80);
    l.add(90);
    li.retainAll(l);
    System.out.println(li);

}
}
```

Console

<terminated> List1 [Java Application]

```
[10, 20, 90, 10, 10, 40, 80]
```

*List1.java

```java
package org.com;

import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;


public class List1 {
public static void main(String[] args) {
    List li =new ArrayList();
    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);

    //System.out.println(li);
    //System.out.println("-------------------------------------");
    List<Integer>l = new LinkedList();
    l.add(30);
    l.add(40);
    l.add(50);
    l.add(60);
    l.add(80);
    li.retainAll(l);
    System.out.println(li);

}
}
```

Console

<terminated> List

```
[30, 40, 50]
```

Test.java

```
1  package org.test;
2
3  import java.util.ArrayList;
4  import java.util.Collections;
5  import java.util.LinkedList;
6  import java.util.List;
7  import java.util.Vector;
8
9  public class Test {
10     public static void main(String[] args) {
11     List <Integer> a =new ArrayList();
12         a.add(10);
13         a.add(20);
14         a.add(30);
15         a.add(90);
16         a.add(10);
17         a.add(10);
18         a.add(40);
19         a.add(50);
20         a.add(30);
21
22         //int length=a.size();
23         //System.out.println(length);
24
25         boolean replaceAll = Collections.replaceAll(a, 10, 100);
26         //System.out.println(replaceAll);
27         System.out.println(a);
28     }
29 }
```

Console

```
<terminated> Test (6) [Java Application] C:\Users\navee
[100, 20, 30, 90, 100, 100, 40, 50, 30]
```

Test.java

```
1  package org.test;
2
3  import java.util.ArrayList;
4  import java.util.LinkedList;
5  import java.util.List;
6  import java.util.Vector;
7
8  public class Test {//10,20,30,90,10,10,40,50,10
9      public static void main(String[] args) {
10     List <Integer> a =new ArrayList();
11         a.add(10);
12         a.add(20);
13         a.add(30);
14         a.add(90);
15         a.add(10);
16         a.add(10);
17         a.add(40);
18         a.add(50);
19         a.add(10);
20
21         a.remove(7);
22         a.add(7, 90);
23         System.out.println(a);
24     }
25 }
```

Console

```
<terminated> Test (6) [Java Application] C:\Users\nave
[10, 20, 30, 90, 10, 10, 40, 90, 10]
```

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
    List <Integer> a =new ArrayList();
        a.add(100);
        a.add(200);
        a.add(300);
        a.add(400);
        a.add(500);
        a.add(600);
        a.add(700);
        //int length=a.size();
        //System.out.println(length);
        a.remove(2);
        a.set(2, 350);
        System.out.println(a);
    }
}
```

Console

```
<terminated> Test (6) [Java Application] C:\
[100, 200, 350, 500, 600, 700]
```

List1.java

```java
package org.com;

import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;


public class List1 {
public static void main(String[] args) {
    List li =new ArrayList();
    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);                    //| 100,200,300,400,500,600,700,8000

    li.add(10);
    li.add(40);
    li.add(80);

    List<Integer>l = new LinkedList();
    l.add(100);
    l.add(200);
    l.add(300);
    l.add(400);
    l.add(500);
    l.add(600);
    l.add(700);
    l.add(800);
    li.retainAll(l);
    System.out.println(li);

}
}
```

C

```
<ter
[]
```

QUESTION 11:

QUESTION 11.1:

Description : Create a new ArrayListlists with values and perform removeAll() function
Input : List = 10,20,30,90,10,10,40,50
Input : List = 30,40,50,60,80

QUESTION 11.2:

Description : Create a new ArrayListlists with values and perform removeAll() function
Input : List = 10,20,30,90,10,10,40,50
Input : List = 10,20,60,50,40,70,80,90

QUESTION 11.3:

Description : Create a new ArrayListlists with values and perform removeAll() function
Input : List = 10,20,30,40,50,60,70,80
Input : List = 100,200,300,400,500,600,700,8000

List1.java

```java
package org.com;

import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;


public class List1 {
public static void main(String[] args) {
    List li =new ArrayList();
    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);

    //System.out.println(li);
    //System.out.println("-------------------------------------");
    List<Integer>l = new LinkedList();
    l.add(30);
    l.add(40);
    l.add(50);
    l.add(60);
    l.add(8);
    li.removeAll(l);
    System.out.println(li);

}
}

```

Console

<terminated> List1 [Java Applic

```
[10, 20, 90, 10, 10]
```

List1.java

```java
package org.com;

import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;


public class List1 {
public static void main(String[] args) {
    List li =new ArrayList();
    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);                    // 100,200,300,400,500,600,700,8000

    li.add(10);
    li.add(40);
    li.add(80);

    List<Integer>l = new LinkedList();
    l.add(100);
    l.add(200);
    l.add(300);
    l.add(400);
    l.add(500);
    l.add(600);
    l.add(700);
    l.add(800);
    li.removeAll(l);
    System.out.println(li);

}
}
```

Console

<terminated> List1 [Java Application] C:\User

```
[10, 20, 30, 90, 10, 10, 40, 80]
```

CourseDetails.java

```java
package org.tcs.com;
import java.util.ArrayList;
import java.util.List;
import java.util.LinkedList;
public class CourseDetails {
    public static void main(String[] args) {
    List li = new ArrayList();          // 10,20,30,90,10,10,40,50
    li.add(10);
    li.add(20);
    li.add(30);
    li.add(90);
    li.add(10);
    li.add(10);
    li.add(40);
    li.add(50);
    List <Integer> l = new LinkedList();
    l.add(30);
    l.add(40);
    l.add(50);
    l.add(60);
    l.add(80);
    li.removeAll(l);
    System.out.println(li);
    }
}
```

Console

<terminated> CourseDetails (4) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.ju

```
[10, 20, 90, 10, 10]
```

DAY8:

1.Userdefine List
2.LinkedList vs Vector
3.Set and types

QUESTIONS:

QUESTION 1.1:

Description : Create a HashSet for the below values
              Input : List = 10,20,30,40,50,60,70,80,90,10,20

QUESTION 1.2:

Description : Create a LinkedHashSet for the below values
              Input : List = 10,20,30,40,50,60,70,80,90,10,20

QUESTION 1.3:

Description : Create a TreeSet for the below values
              Input : List = 10,20,30,40,50,60,70,80,90,10,20

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
    Set <Integer> a =new LinkedHashSet();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(40);
        a.add(50);
        a.add(60);
        a.add(70);
        a.add(80);
        a.add(90);
        a.add(10);
        a.add(20);


        System.out.println(a);


    }
}
```

Console

<terminated> Test (6) [Java Application] C:\Users\nav

```
[10, 20, 30, 40, 50, 60, 70, 80, 90]
```

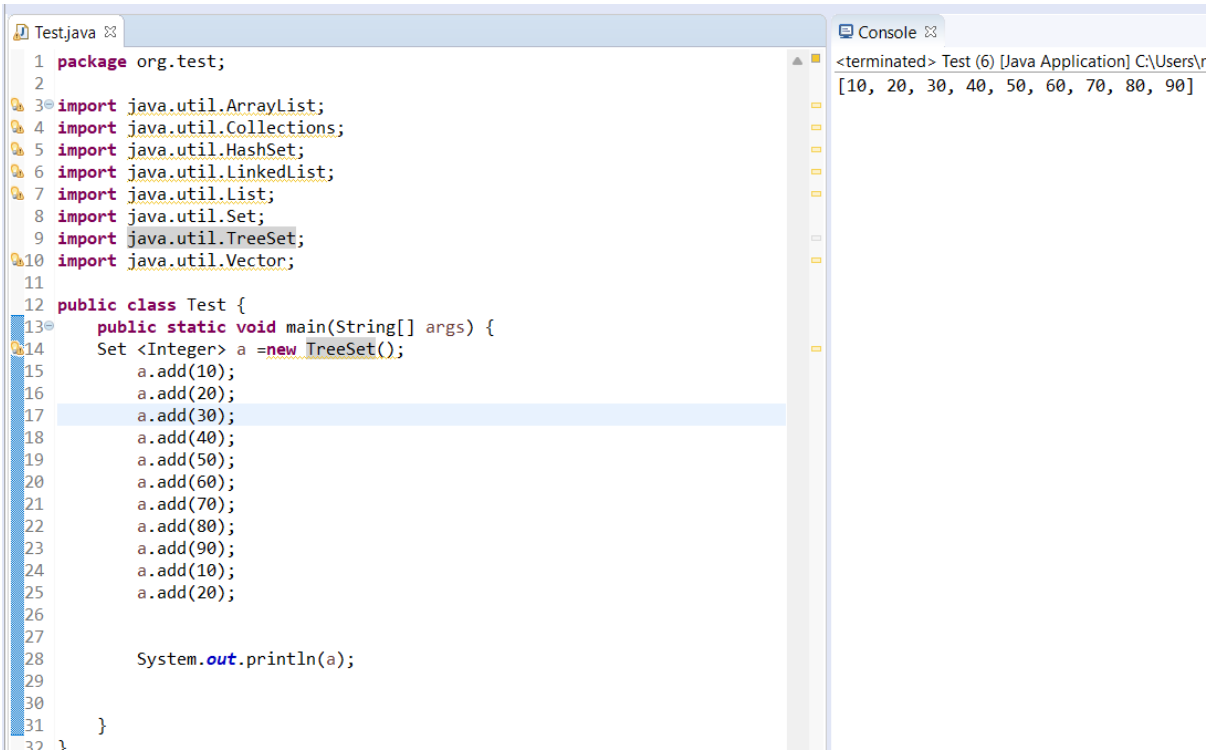
Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
    Set <Integer> a =new HashSet();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(40);
        a.add(50);
        a.add(60);
        a.add(70);
        a.add(80);
        a.add(90);
        a.add(10);
        a.add(20);


        System.out.println(a);


    }
}
```

Console

<terminated> Test (6) [Java Application] C:\Users\na

```
[80, 50, 20, 70, 40, 10, 90, 60, 30]
```



QUESTION 2:

QUESTION 2.1:

Description : Convert the below list in to set(use addAll()) method
Input : List = 10,20,30,90,10,10,40,50

QUESTION 2.2:

Description : Convert the below list in to set(use addAll()) method
Input : List = 105,205,305,405,505,605,705,805,505,605

QUESTION 2.3:

Description : Convert the below list in to set(use addAll()) method
Input : List = 100,200,300,400,500,600,700,100,300,500

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {

        List<Integer>a =new ArrayList();
        a.add(105);
        a.add(205);
        a.add(305);
        a.add(405);
        a.add(505);
        a.add(605);
        a.add(705);
        a.add(805);
        a.add(505);
        a.add(605);

        Set <Integer> z =new TreeSet();
        z.addAll(a);
        System.out.println(z);
    }

    }
```

Console

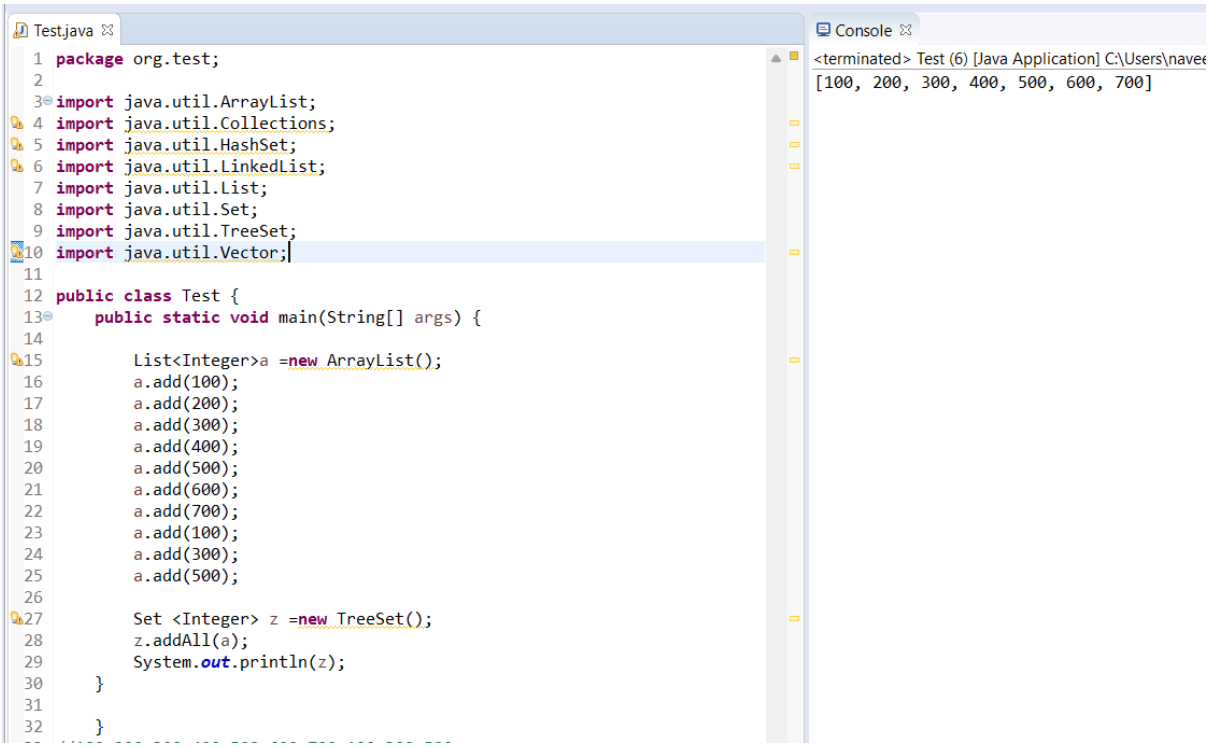
```
<terminated> Test (6) [Java Application] C:\Users\navee\
[105, 205, 305, 405, 505, 605, 705, 805]
```

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {

        List<Integer>a =new ArrayList();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(90);
        a.add(10);
        a.add(10);
        a.add(40);
        a.add(50);
        Set <Integer> z =new TreeSet();
        z.addAll(a);
        System.out.println(z);
    }

    }
```

Console

```
<terminated> Test (6) [Java Applica
[10, 20, 30, 40, 50, 90]
```



QUESTION 3:

QUESTION 3.1:

Description : Get the each value of set by using enhanced for loop
Input: List = 105,205,305,405,505,605,705,805

QUESTION 3.2:

Description : Create a TreeSet and iterate each value in the set by using enhanced for loop
Input : List = 10,20,30,40,50,60,70,80,90,10,20

QUESTION 3.3:

Description : Create a HashSet and iterate each value in the set by using enhanced for loop
Input : List = 10,20,30,40,50,60,70,80,90,10,20

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
        Set<Integer> a =new LinkedHashSet();
        a.add(105);
        a.add(205);
        a.add(305);
        a.add(405);
        a.add(505);
        a.add(605);
        a.add(705);
        a.add(805);
        a.add(505);
        a.add(605);
        for (Integer i: a) {
            System.out.println(i);

        }

    }

```

Console

<terminated> Test (

```
105
205
305
405
505
605
705
805
```

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
        Set<Integer> a =new HashSet();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(40);
        a.add(50);
        a.add(60);
        a.add(70);
        a.add(80);
        a.add(90);
        a.add(10);
        a.add(20);



        for (Integer i: a) {
            System.out.println(i);

        }

    }

    }
```

Console

```
<terminated> Te
80
50
20
70
40
10
90
60
30
```

QUESTION 4:

QUESTION 4.1:

Description : Create a new HashSet with values and return the common values
Input : List = 10,20,30,90,10,10,40,50
Input : List = 30,40,50,60,80

QUESTION 4.2:

Description : Create a new LinkedHashSet with values and return the common values
Input : List = 10,20,30,90,10,10,40,50
Input : List = 10,20,60,50,40,70,80,90

QUESTION 4.3:

Description : Create a new TreeSet with values and return the common values
Input : List = 10,20,30,40,50,60,70,80
Input : List = 100,200,300,400,500,600,700,8000

Test.java

```
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
        Set<Integer> a =new LinkedHashSet();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(90);
        a.add(10);
        a.add(10);
        a.add(40);
        a.add(50);

Set<Integer>a1 =new LinkedHashSet();
a1.add(10);
a1.add(20);
a1.add(60);
a1.add(50);
a1.add(40);
a1.add(70);
a1.add(80);
a1.add(90);
a.retainAll(a1);
System.out.println(a);
}

}
```

Console

<terminated> Test (6) [Java Application]

```
[10, 20, 90, 40, 50]
```

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {

        Set<Integer> a =new HashSet();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(90);
        a.add(10);
        a.add(10);
        a.add(40);
        a.add(50);

Set<Integer>a1 =new HashSet();
a1.add(30);
a1.add(40);
a1.add(50);
a1.add(60);
a1.add(80);

a.retainAll(a1);
System.out.println(a);
}

}
```

Console

<terminated> Te

```
[50, 40, 30]
```

Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
        Set<Integer> a =new TreeSet();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(40);
        a.add(50);
        a.add(60);
        a.add(70);
        a.add(80);

Set<Integer>a1 =new TreeSet();
a1.add(100);
a1.add(200);
a1.add(300);
a1.add(400);
a1.add(500);
a1.add(600);
a1.add(700);
a1.add(8000);
a.retainAll(a1);
System.out.println(a);
}

    }
```

Console

<terminated>

[]

QUESTION 5:

QUESTION 5.1:

Description : Create a new HashSet with values and perform removeAll() function

Input : List = 10,20,30,90,10,10,40,50
Input : List = 30,40,50,60,80

QUESTION 5.2:

Description : Create a new LinkedHashSet with values and perform removeAll() function

Input : List = 10,20,30,90,10,10,40,50
Input : List = 10,20,60,50,40,70,80,90

QUESTION 5.3:

Description : Create a new TreeSet with values and perform removeAll() function

Input : List = 10,20,30,40,50,60,70,80

Input : List = 100,200,300,400,500,600,700,8000

Test.java

```
1  package org.test;
2
3  import java.util.ArrayList;
4  import java.util.Collections;
5  import java.util.HashSet;
6  import java.util.LinkedHashSet;
7  import java.util.LinkedList;
8  import java.util.List;
9  import java.util.Set;
10 import java.util.TreeSet;
11 import java.util.Vector;
12
13 public class Test {
14     public static void main(String[] args) {
15         Set<Integer> a =new LinkedHashSet();
16         a.add(10);
17         a.add(20);
18         a.add(30);
19         a.add(90);
20         a.add(10);
21         a.add(10);
22         a.add(40);
23         a.add(50);
24
25 Set<Integer>a1 =new LinkedHashSet();
26 a1.add(10);
27 a1.add(20);
28 a1.add(60);
29 a1.add(50);
30 a1.add(40);
31 a1.add(70);
32 a1.add(80);
33 a1.add(90);
34 a.removeAll(a1);
35 System.out.println(a);
36 }
37
38     }
```

Console

<terminated> T

```
[30]
```

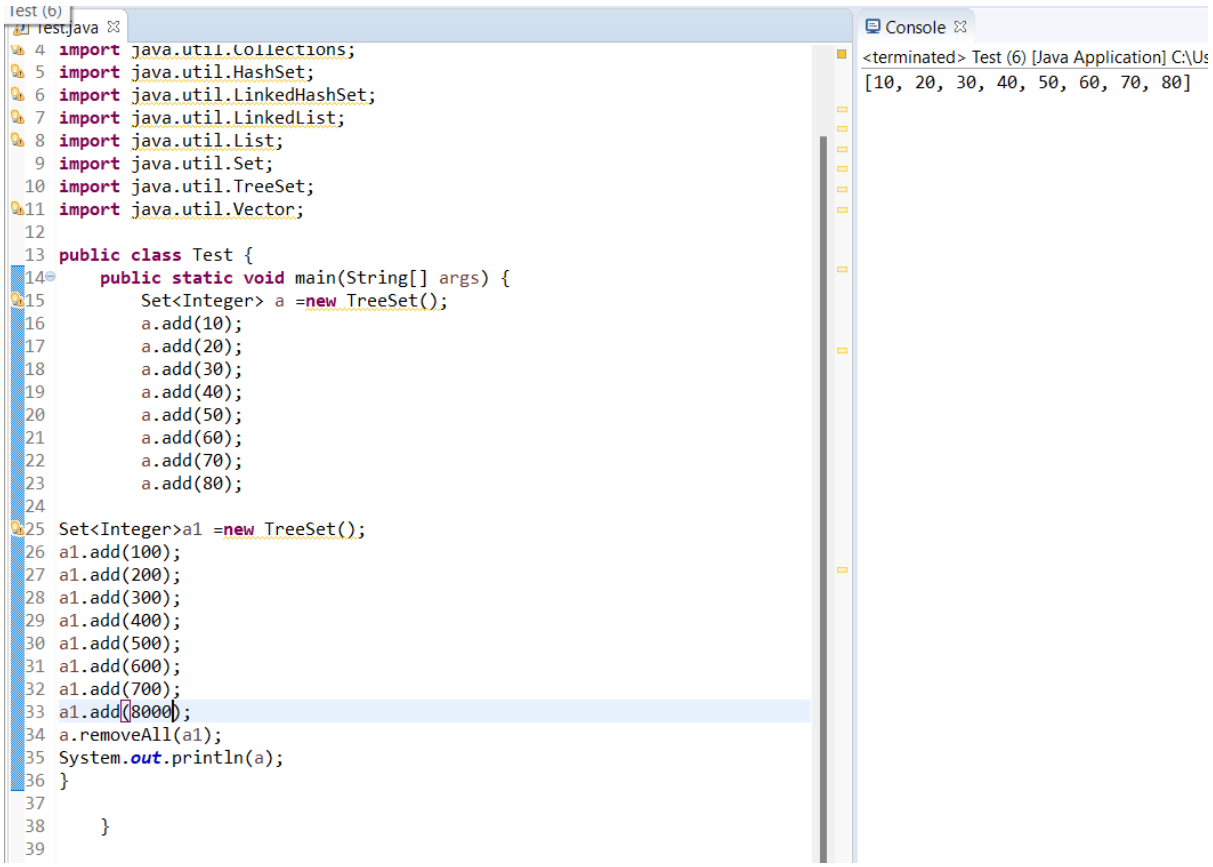
Test.java

```java
package org.test;

import java.util.ArrayList;
import java.util.Collections;
import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;
import java.util.TreeSet;
import java.util.Vector;

public class Test {
    public static void main(String[] args) {
        Set<Integer> a =new HashSet();
        a.add(10);
        a.add(20);
        a.add(30);
        a.add(90);
        a.add(10);
        a.add(10);
        a.add(40);
        a.add(50);

Set<Integer>a1 =new HashSet();
a1.add(30);
a1.add(40);
a1.add(50);
a1.add(60);
a1.add(80);
a.removeAll(a1);
System.out.println(a);
}

    }
```

Console

<terminated> Test (6)

```
[20, 10, 90]
```



QUESTION 6:

Description: Create a userdefine Set and insert the 10 employee details
Input : empId,name,phone,address,dob,doj,eMail,gender,Sal
Output: Using scanner class insert 10 employee details

QUESTION 7:

Description: Create a userdefine Set and insert the 10 Student details
Key : stdId,stdName,stdPhone,address,dOB,eMail,gender
Give the related values for key for each Student

DAY9:

1.Map and types
2.User define Map

QUESTIONS(practical)

QUESTION 1.1:

Description : Create a HashMap with the below key and values
key : 10,20,30,40,50,60,10,50,40
values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

QUESTION 1.2:

Description : Create a LinkedHashMap with the below key and values
key : 10,20,30,40,50,60,10,50,40
Values : 10,20,30,40,50,60,10,50,40

QUESTION 1.3:

Description : Create a TreeHashMap with the below key and values
key : !,@,#,$,%,^,&,*,(,
Values : 10,20,30,40,50,60,10,50,40

QUESTION 1.4:

Description : Create a HashTable with the below key and values
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira

Test3.java | Test2.java

```java
package org.test;

import java.util.Map;
import java.util.TreeMap;

public class Test3 {
    public static void main(String[] args) {


    Map<String,Integer> a =new TreeMap();
    a.put("!", 10);
    a.put("@", 20);
    a.put("#", 30);
    a.put("$", 40);
    a.put("%", 50);
    a.put("^", 60);
    a.put("&", 10);
    a.put("*", 50);
    a.put("(", 40);
    System.out.println(a);
}
}//key     : !,@,#,$,%,^,&,*,(,
  //      Values : 10,20,30,40,50,60,10,50,40

```

Console

<terminated> Test3 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hots

```
{!=10, #=30, $=40, %=50, &=10, (=40, *=50, @=20, ^=60}
```

Test2.java

```java
package org.test;

import java.util.LinkedHashMap;
import java.util.Map;

public class Test2 {
    public static void main(String[] args) {
        Map<Integer,Integer> q =new LinkedHashMap();
        q.put(10, 20);
        q.put(20, 20);
        q.put(30, 30);
        q.put(40, 40);
        q.put(50, 50);
        q.put(60, 60);
        q.put(10, 50);
        q.put(50, 40);
        q.put(40, 40);
        System.out.println(q);

    }

}
```

Console

<terminated> Test2 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.

```
{10=50, 20=20, 30=30, 40=40, 50=40, 60=60}
```

Test.java

```java
package org.test;

import java.util.ArrayList;

public class Test {
    public static void main(String[] args) {

        Map<Integer,String> q =new HashMap();
        q.put(10, "java");
        q.put(20, "sql");
        q.put(30, "oops");
        q.put(40, "sql");
        q.put(50, "oracle");
        q.put(60, "DB");
        q.put(10, "selenium");
        q.put(50, "psql");
        q.put(40, "Hadoop");
        System.out.println(q);
}
}
```

Console

<terminated> Test (6) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.v

```
{50=psql, 20=sql, 40=Hadoop, 10=selenium, 60=DB, 30=oops}
```

Test4.java

```java
package org.test;

import java.util.Hashtable;
import java.util.Map;

public class Test4 {
    public static void main(String[] args) {
        Map<String,String> u= new Hashtable();
        u.put("vel", "Selenium");
        u.put("Ganesh", "framework");
        u.put("Dinesh", "oracle");
        u.put("Venkat", "corejava");
        u.put("subash", "jira");
        System.out.println(u);


    }

}
```

Console

<terminated> Test4 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jr

```
{Ganesh=framework, Venkat=corejava, vel=Selenium, Dinesh=oracle, subash=jira}
```

QUESTION 2:

QUESTION 2.1:

Description : Create a HashMap with the below key and values and get(print) the key only in the map.
key : 10,20,30,40,50,60,10,50,40
values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

QUESTION 2.2:

Description : Create a LinkedHashMap with the below key and values and get(print) the key only in the map.
key : 10,20,30,40,50,60,10,50,40
Values : 10,20,30,40,50,60,10,50,40

QUESTION 2.3:

Description : Create a TreeMap with the below key and values and get(print) the key only in the map.
key : !,@,#,$,%,^,&,*,(,
Values : 10,20,30,40,50,60,10,50,409

QUESTION 2.4:

Description : Create a HashTable with the below key and values and get(print) the key only in the map.
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira

Test2.java *Test3.java

```java
package org.test;
import java.util.Map;
import java.util.Set;
import java.util.TreeMap;
import java.util.Map.Entry;
public class Test3 {
    public static void main(String[] args) {
    Map<String,Integer> a =new TreeMap();
    a.put("!", 10);
    a.put("@", 20);
    a.put("#", 30);
    a.put("$", 40);
    a.put("%", 50);
    a.put("^", 60);
    a.put("&", 10);
    a.put("*", 50);
    a.put("(", 40);
    Set<Entry<String, Integer>> entrySet = a.entrySet();
    for (Entry<String, Integer> entry : entrySet) {
        System.out.println(entry.getKey());
    }
}
}
```

Console

<terminated> Test2 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.ope

```
10
20
30
40
50
60
```

Test2.java

```java
package org.test;

import java.util.LinkedHashMap;

public class Test2 {
    public static void main(String[] args) {
        Map<Integer,Integer> q =new LinkedHashMap();
        q.put(10, 20);
        q.put(20, 20);
        q.put(30, 30);
        q.put(40, 40);
        q.put(50, 50);
        q.put(60, 60);
        q.put(10, 50);
        q.put(50, 40);
        q.put(40, 40);
        Set<Entry<Integer, Integer>> entrySet = q.entrySet();
        for (Entry<Integer, Integer> entry : entrySet) {
            System.out.println(entry.getKey());
        }

    }

}
//key    : 10 20 30 40 50 60 10 50 40
```

Console

<terminated> Test2 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.ho

```
10
20
30
40
50
60
```

Test.java

```java
package org.test;

import java.util.ArrayList;

public class Test {
    public static void main(String[] args) {

        Map<Integer,String> q =new HashMap();
        q.put(10, "java");
        q.put(20, "sql");
        q.put(30, "oops");
        q.put(40, "sql");
        q.put(50, "oracle");
        q.put(60, "DB");
        q.put(10, "selenium");
        q.put(50, "psql");
        q.put(40, "Hadoop");
        Set<Entry<Integer, String>> entrySet = q.entrySet();
        for (Entry<Integer, String> entry : entrySet) {
            System.out.println(entry.getKey());
        }
}
}

```

Console

<terminated> Test (6) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.just

```
50
20
40
10
60
30
```

Test4.java

```java
package org.test;

import java.util.Hashtable;

public class Test4 {
    public static void main(String[] args) {
        Map<String,String> u= new Hashtable();
        u.put("vel", "Selenium");
        u.put("Ganesh", "framework");
        u.put("Dinesh", "oracle");
        u.put("Venkat", "corejava");
        u.put("subash", "jira");
        Set<Entry<String, String>> entrySet = u.entrySet();
        for (Entry<String, String> entry : entrySet) {
            System.out.println(entry.getKey());
        }

    }

}
```

Console

<terminated> Test4 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.h

```
Ganesh
Venkat
vel
Dinesh
subash
```

QUESTION 3:

QUESTION 3.1:

Description : Create a HashMap with the below key and values and get(print) the values only in the map.
key : 10,20,30,40,50,60,10,50,40
values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

QUESTION 3.2:

Description : Create a LinkedHashMap with the below key and values and get(print) the values only in the map.
key : 10,20,30,40,50,60,10,50,40
Values : 10,20,30,40,50,60,10,50,40

QUESTION 3.3:

Description : Create a TreeHashMap with the below key and values and get(print) the values only in the map.
key : !,@,#,$,%,^,&,*,(,

Values : 10,20,30,40,50,60,10,50,409

QUESTION 3.4:

Description : Create a HashTable with the below key and values and get(print) the key only in the map.
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira

Test3.java

```
1  package org.test;
2  import java.util.Map;
6  public class Test3 {
7      public static void main(String[] args) {
8      Map<String,Integer> a =new TreeMap();
9      a.put("!", 10);
10     a.put("@", 20);
11     a.put("#", 30);
12     a.put("$", 40);
13     a.put("%", 50);
14     a.put("^", 60);
15     a.put("&", 10);
16     a.put("*", 50);
17     a.put("(", 40);
18     Set<Entry<String, Integer>> entrySet = a.entrySet();
19     for (Entry<String, Integer> entry : entrySet) {
20         System.out.println(entry.getValue());
21     }
22 }
23 }
```

Console

<terminated> Test3 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hots

```
10
30
40
50
10
40
50
20
60
```

Test2.java

```java
package org.test;

import java.util.LinkedHashMap;

public class Test2 {
    public static void main(String[] args) {
        Map<Integer,Integer> q =new LinkedHashMap();
        q.put(10, 20);
        q.put(20, 20);
        q.put(30, 30);
        q.put(40, 40);
        q.put(50, 50);
        q.put(60, 60);
        q.put(10, 50);
        q.put(50, 40);
        q.put(40, 40);
        Set<Entry<Integer, Integer>> entrySet = q.entrySet();
        for (Entry<Integer, Integer> entry : entrySet) {
            System.out.println(entry.getValue());
        }

    }

}
```

Console

<terminated> Test2 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspo

```
50
20
30
40
40
60
```

Test.java

```
1  package org.test;
2
3  import java.util.ArrayList;
16
17 public class Test {
18     public static void main(String[] args) {
19
20         Map<Integer,String> q =new HashMap();
21         q.put(10, "java");
22         q.put(20, "sql");
23         q.put(30, "oops");
24         q.put(40, "sql");
25         q.put(50, "oracle");
26         q.put(60, "DB");
27         q.put(10, "selenium");
28         q.put(50, "psql");
29         q.put(40, "Hadoop");
30         Set<Entry<Integer, String>> entrySet = q.entrySet();
31         for (Entry<Integer, String> entry : entrySet) {
32             System.out.println(entry.getValue());
33         }
34 }
35 }
36
```

Console

<terminated> Test (6) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.o

```
psql
sql
Hadoop
selenium
DB
oops
```

Test4.java

```java
package org.test;

import java.util.Hashtable;

public class Test4 {
    public static void main(String[] args) {
        Map<String,String> u= new Hashtable();
        u.put("vel", "Selenium");
        u.put("Ganesh", "framework");
        u.put("Dinesh", "oracle");
        u.put("Venkat", "corejava");
        u.put("subash", "jira");
        Set<Entry<String, String>> entrySet = u.entrySet();
        for (Entry<String, String> entry : entrySet) {
            System.out.println(entry.getValue());
        }

    }

}
```

Console

<terminated> Test4 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.open

```
framework
corejava
Selenium
oracle
jira
```

QUESTION 4:

QUESTION 4.1:

Description : Create a HashMap with the below key and values and iterate it using enhanced for loop.

key : 10,20,30,40,50,60,10,50,40
values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

QUESTION 4.2:

Description : Create a LinkedHashMap with the below key and values and iterate it using enhanced for loop.

key : 10,20,30,40,50,60,10,50,40
Values : 10,20,30,40,50,60,10,50,40

QUESTION 4.3:

Description : Create a TreeHashMap with the below key and values and iterate it using enhanced for loop.

key : !,@,#,$,%,^,&,*,(,

Values : 10,20,30,40,50,60,10,50,409

QUESTION 4.4:

Description : Create a HashTable with the below key and values and iterate it using enhanced for loop.
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira

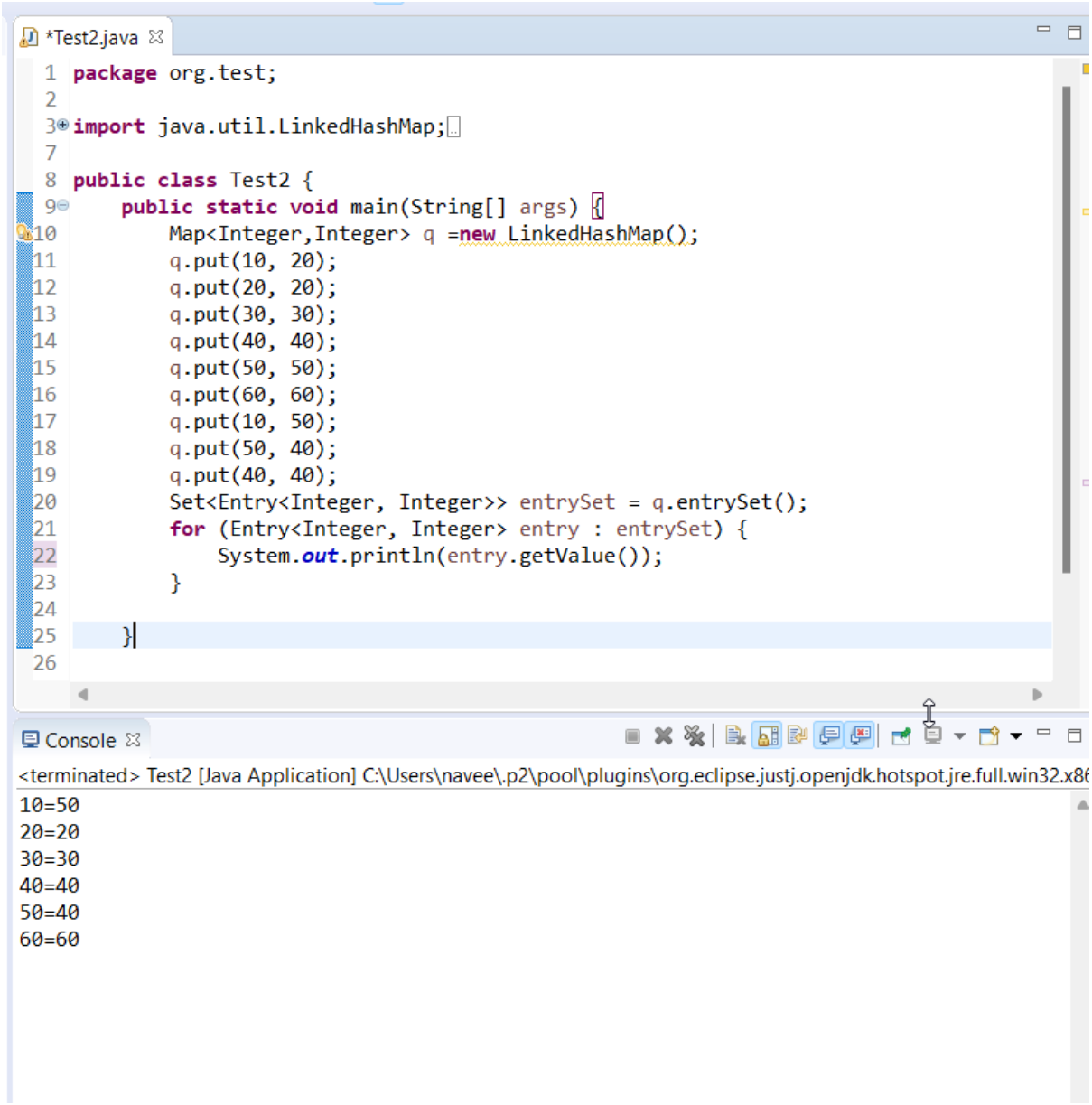
Test3.java

```
1  package org.test;
2  import java.util.Map;
6  public class Test3 {
7      public static void main(String[] args) {
8      Map<String,Integer> a =new TreeMap();
9      a.put("!", 10);
10     a.put("@", 20);
11     a.put("#", 30);
12     a.put("$", 40);
13     a.put("%", 50);
14     a.put("^", 60);
15     a.put("&", 10);
16     a.put("*", 50);
17     a.put("(", 40);
18     Set<Entry<String, Integer>> entrySet = a.entrySet();
19     for (Entry<String, Integer> entry : entrySet) {
20         System.out.println(entry);
21     }
22 }
23 }
```

Console

<terminated> Test3 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse

```
!=10
#=30
$=40
%=50
&=10
(=40
*=50
@=20
^=60
```


*Test2.java
1 package org.test;
2
3 import java.util.LinkedHashMap;
7
8 public class Test2 {
9 public static void main(String[] args) {
10 Map<Integer,Integer> q =new LinkedHashMap();
11 q.put(10, 20);
12 q.put(20, 20);
13 q.put(30, 30);
14 q.put(40, 40);
15 q.put(50, 50);
16 q.put(60, 60);
17 q.put(10, 50);
18 q.put(50, 40);
19 q.put(40, 40);
20 Set<Entry<Integer, Integer>> entrySet = q.entrySet();
21 for (Entry<Integer, Integer> entry : entrySet) {
22 System.out.println(entry.getValue());
23 }
24
25 }
26
Console
<terminated> Test2 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86
10=50
20=20
30=30
40=40
50=40
60=60

Test2.java

```java
package org.test;

import java.util.LinkedHashMap;

public class Test2 {
    public static void main(String[] args) {
        Map<Integer,Integer> q =new LinkedHashMap();
        q.put(10, 20);
        q.put(20, 20);
        q.put(30, 30);
        q.put(40, 40);
        q.put(50, 50);
        q.put(60, 60);
        q.put(10, 50);
        q.put(50, 40);
        q.put(40, 40);
        Set<Entry<Integer, Integer>> entrySet = q.entrySet();
        for (Entry<Integer, Integer> entry : entrySet) {
            System.out.println(entry);
        }

    }

```

Console

<terminated> Test2 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.wi

```
10=50
20=20
30=30
40=40
50=40
60=60
```

Test.java

```
1  package org.test;
2
3  import java.util.ArrayList;
16
17 public class Test {
18     public static void main(String[] args) {
19
20         Map<Integer,String> q =new HashMap();
21         q.put(10, "java");
22         q.put(20, "sql");
23         q.put(30, "oops");
24         q.put(40, "sql");
25         q.put(50, "oracle");
26         q.put(60, "DB");
27         q.put(10, "selenium");
28         q.put(50, "psql");
29         q.put(40, "Hadoop");
30         Set<Entry<Integer, String>> entrySet = q.entrySet();
31         for (Entry<Integer, String> entry : entrySet) {
32             System.out.println(entry);
33         }
34 }
35 }
36
```

Console

<terminated> Test (6) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot

```
50=psql
20=sql
40=Hadoop
10=selenium
60=DB
30=oops
```

Test4.java

```java
package org.test;

import java.util.Hashtable;

public class Test4 {
    public static void main(String[] args) {
        Map<String,String> u= new Hashtable();
        u.put("vel", "Selenium");
        u.put("Ganesh", "framework");
        u.put("Dinesh", "oracle");
        u.put("Venkat", "corejava");
        u.put("subash", "jira");
        Set<Entry<String, String>> entrySet = u.entrySet();
        for (Entry<String, String> entry : entrySet) {
            System.out.println(entry);
        }

    }

}
```

Console

<terminated> Test4 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.ope

```
Ganesh=framework
Venkat=corejava
vel=Selenium
Dinesh=oracle
subash=jira
```

QUESTION 5:

QUESTION 5.1:

Description : Create a HashMap with the below key and values and iterate it using enhanced for loop and get the key and values combination.
key : 10,20,30,40,50,60,10,50,40
values : java,sql,oops,Sql,oracle,DB,selenium,psql,Hadoop.

Sample Output:

10
java
20
sql
30
oops
40
sql
Description : like this you have to itearte the map

QUESTION 5.2:

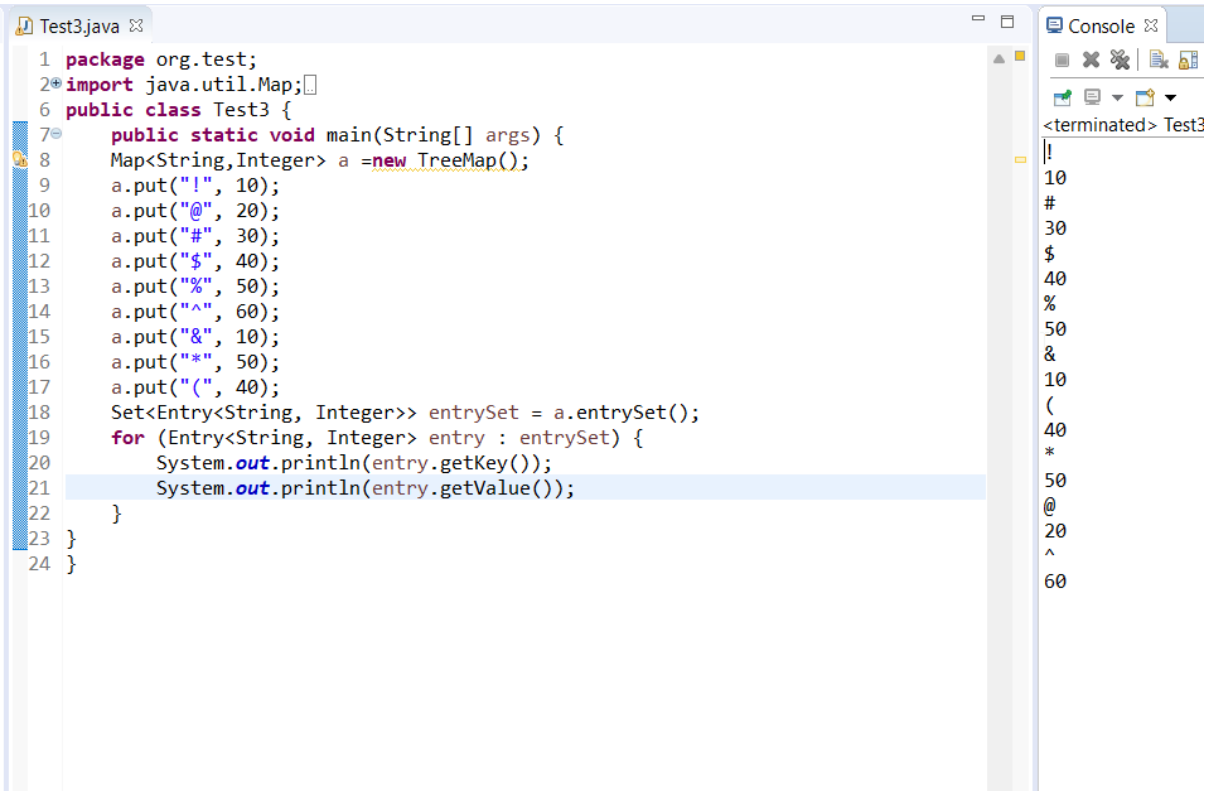
Description : Create a LinkedHashMap with the below key and values and iterate it using enhanced for loop and get the key and values combination..
key : 10,20,30,40,50,60,10,50,40
Values : 10,20,30,40,50,60,10,50,40

QUESTION 5.3:

Description : Create a TreeHashMap with the below key and values and iterate it using enhanced for loop and get the key and values combination.
key : !,@,#,$,%,^,&,*,(,
Values : 10,20,30,40,50,60,10,50,409

QUESTION 5.4:

Description : Create a HashTable with the below key and values and iterate it using enhanced for loop and get the key and values combination.
Key : vel,Ganesh,Dinesh,Vengat,subash
Values : Selenium,framework,oracle,corejava,jira




```java
package org.test;
import java.util.Map;
public class Test3 {
    public static void main(String[] args) {
    Map<String,Integer> a =new TreeMap();
    a.put("!", 10);
    a.put("@", 20);
    a.put("#", 30);
    a.put("$", 40);
    a.put("%", 50);
    a.put("^", 60);
    a.put("&", 10);
    a.put("*", 50);
    a.put("(", 40);
    Set<Entry<String, Integer>> entrySet = a.entrySet();
    for (Entry<String, Integer> entry : entrySet) {
        System.out.println(entry.getKey());
        System.out.println(entry.getValue());
    }
}
}
```

Test2.java

```java
package org.test;

import java.util.LinkedHashMap;

public class Test2 {
    public static void main(String[] args) {
        Map<Integer,Integer> q =new LinkedHashMap();
        q.put(10, 20);
        q.put(20, 20);
        q.put(30, 30);
        q.put(40, 40);
        q.put(50, 50);
        q.put(60, 60);
        q.put(10, 50);
        q.put(50, 40);
        q.put(40, 40);
        Set<Entry<Integer, Integer>> entrySet = q.entrySet();
        for (Entry<Integer, Integer> entry : entrySet) {
            System.out.println(entry.getKey());
            System.out.println(entry.getValue());
        }

    }

}
```

Console

<terminated> Test2 [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win3

```
10
50
20
20
30
30
40
40
50
40
60
60
```

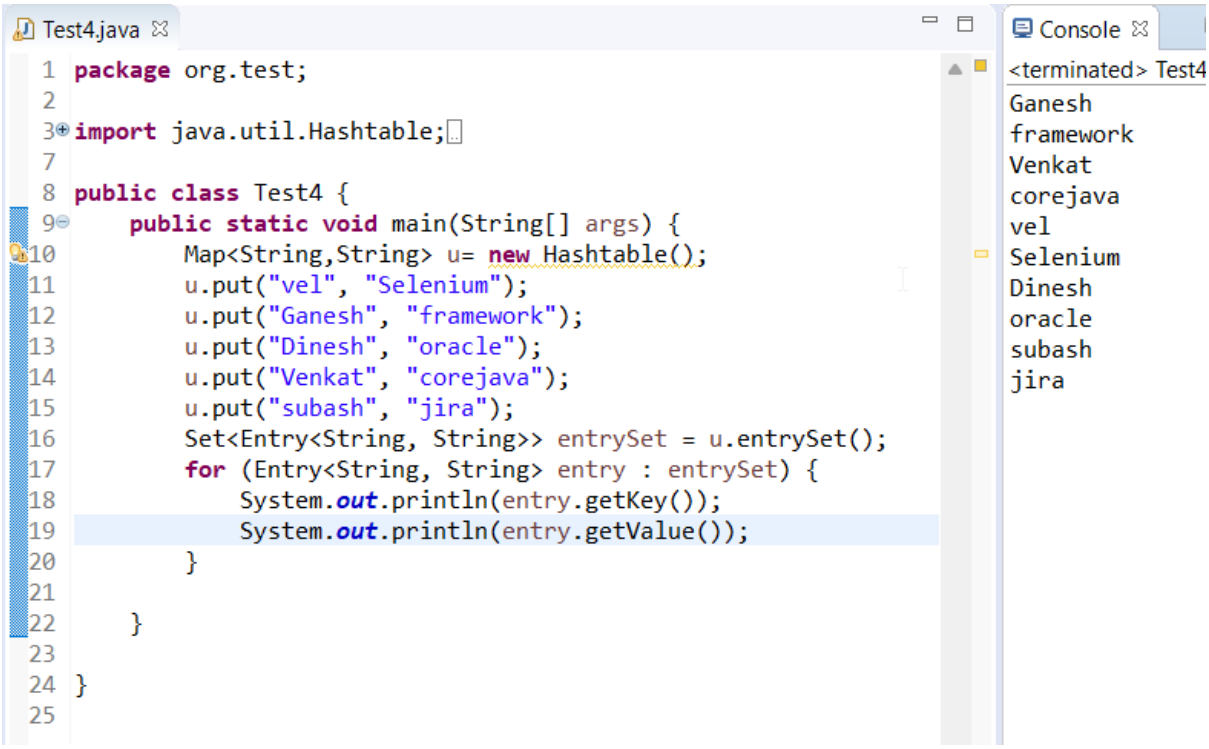
Test.java

```java
package org.test;

import java.util.ArrayList;

public class Test {
    public static void main(String[] args) {

        Map<Integer,String> q =new HashMap();
        q.put(10, "java");
        q.put(20, "sql");
        q.put(30, "oops");
        q.put(40, "sql");
        q.put(50, "oracle");
        q.put(60, "DB");
        q.put(10, "selenium");
        q.put(50, "psql");
        q.put(40, "Hadoop");
        Set<Entry<Integer, String>> entrySet = q.entrySet();
        for (Entry<Integer, String> entry : entrySet) {
            System.out.println(entry.getKey());
            System.out.println(entry.getValue());
        }
}
}
```

Console

<terminated> Test (6) [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.ope

```
50
psql
20
sql
40
Hadoop
10
selenium
60
DB
30
oops
```




```
QUESTION 6:
------------
Description: Create a userdefine Map and insert the 10 employee details
             Key    :  empId,name,phone,address,dob,doj,eMail,gender,Sal
             Give the related values for key for each employee
QUESTION 7:
------------
Description: Create a userdefine Map and insert the 10 Student details
             Key    :  stdId,stdName,stdPhone,address,dOB,eMail,gender
             Give the related values for key for each Student
```

DAY10:

1.Exception
2.throw and throws
3.User define exception

QUESTIONS(Programs)

QUESTION 1:

Description : Find the output for the program:

```
        public class Test
        {
          public static void main(String[] args)
            {
          try
          {
                System.out.printf("1");
              int sum = 9 / 0;
                System.out.printf("2");
          }
          catch(ArithmeticException e)
          {
                System.out.printf("3");
          }
           catch(Exception e)
          {
                System.out.printf("4");
          }
          finally
          {
             System.out.printf("5");
          }
             }
         }
```

QUESTION 2:

Description : Find the output for the program:

```
          public class Test
            {
           private void m1()
               {
           m2();
            System.out.printf("1");
                 }
             private void m2()
               {
            m3();
            System.out.printf("2");
                 }
           private void m3()
               {
           System.out.printf("3");
          try
          {
                int sum = 4/0;
```

```
                System.out.printf("4");
                   }
           catch(ArithmeticException e)
           {
                System.out.printf("5");
           }

                System.out.printf("7");
                 }
             public static void main(String[] args)
              {
                 Test obj = new Test();
                obj.m1();
                }
                 }
```

QUESTION 3:

Description : Find the output for the program:

```
                public class Test
                 {
               public static void main(String[] args)
                 {
                   try
                 {
                  System.out.printf("1");
                   int data = 5 / 0;
                 }
                catch(ArithmeticException e)
                 {
                System.out.printf("2");
                System.exit(0);
                  }
                finally
                   {
                 System.out.printf("3");
                   }
                System.out.printf("4");
                  }
                   }
```

QUESTION 4:

Description : Find the output for the program:

```
                public class Test
                  {
                public static void main(String[] args)
                  {
                  try
                  {
                 System.out.printf("1");
                   int data = 5 / 0;
                   }
                catch(ArithmeticException e)
                  {
                Throwable obj = new Throwable("Sample");
                 try
                  {
                 throw obj;
```

```
                  }
                catch (Throwable e1)
                   {
                System.out.printf("8");
                   }
                   }
              finally
               {
              System.out.printf("3");
               }
           System.out.printf("4");
              }
               }
```

QUESTION 5:

Description : Find the output for the program:

```
              import java.io.EOFException;
              import java.io.IOException;

               public class Test
               {
              public static void main(String[] args)
              {
              try
              {
              System.out.printf("1");
              int value = 10 / 0;
               throw new IOException();
               }
              catch(EOFException e)
                 {
           System.out.printf("2");
              }
          catch(ArithmeticException e)
            {
           System.out.printf("3");
              }
          catch(NullPointerException e)
           {
           System.out.printf("4");
           }
          catch(IOException e)
            {
           System.out.printf("5");
           }
              catch(Exception e)
          {
           System.out.printf("6");
             }
             }
             }
```

Test.java

```
1  package org.test;
2
3  public class Test {
4      public static void main(String[] args)
5      {
6      try
7      {
8     System.out.printf("1");
9       int data = 5 / 0;
10      }
11    catch(ArithmeticException e)
12      {
13    Throwable obj = new Throwable("Sample");
14     try
15      {
16     throw obj;
17       }
18     catch (Throwable e1)
19        {
20     System.out.printf("8");
21        }
22        }
23   finally
24    {
25   System.out.printf("3");
26    }
27  System.out.printf("4");
28    }
29     }
```

Console

<terminated>

1834

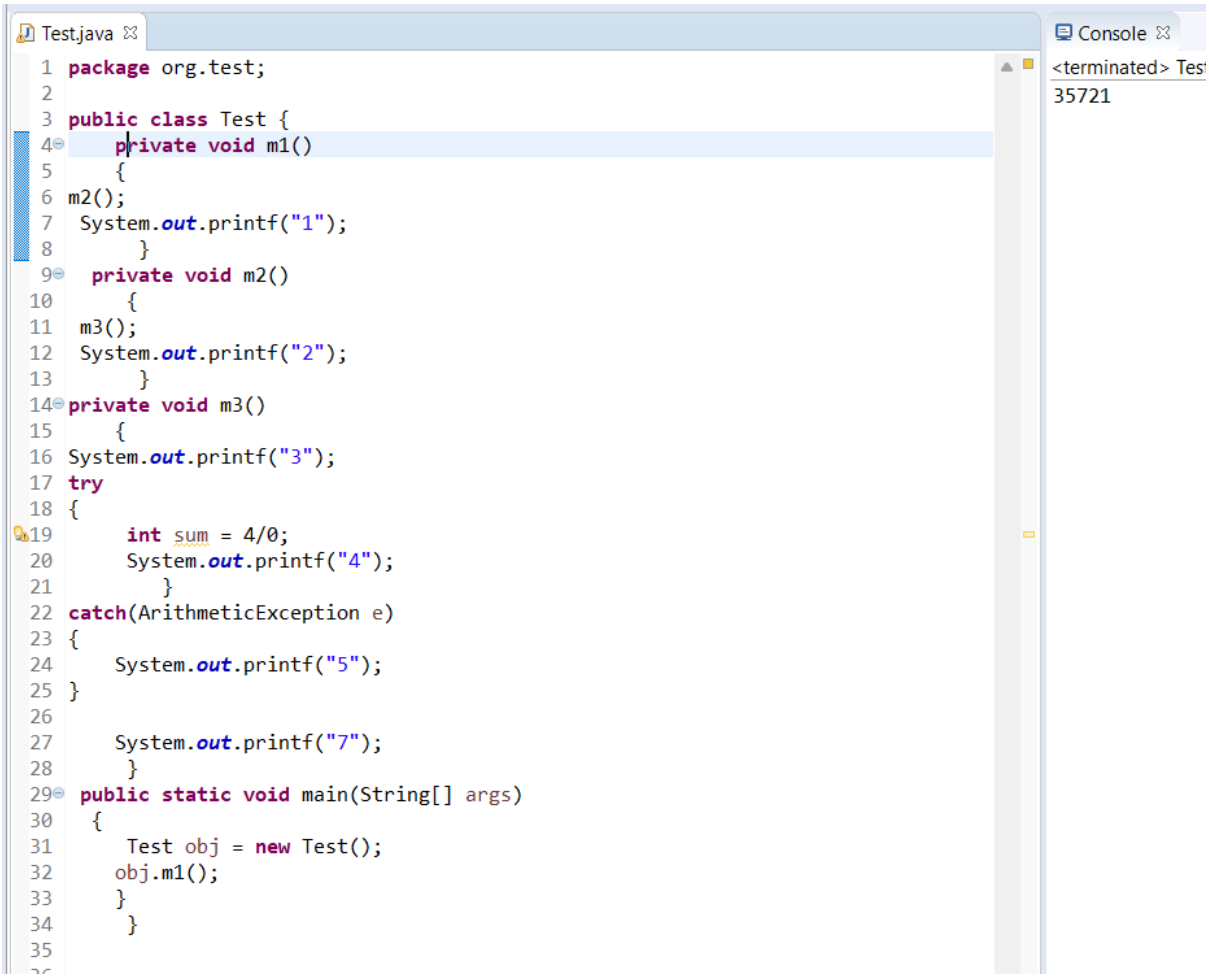
Test.java

```
1  package org.test;
2
3  public class Test {
4      public static void main(String[] args)
5      {
6        try
7      {
8       System.out.printf("1");
9        int data = 5 / 0;
10     }
11    catch(ArithmeticException e)
12     {
13    System.out.printf("2");
14    System.exit(0);
15      }
16    finally
17       {
18     System.out.printf("3");
19       }
20    System.out.printf("4");
21      }
22       }
23
```

Console

<terminated> Te

12


Test.java
Console
<terminated> Test
35721
1 package org.test;
2
3 public class Test {
4 private void m1()
5 {
6 m2();
7 System.out.printf("1");
8 }
9 private void m2()
10 {
11 m3();
12 System.out.printf("2");
13 }
14 private void m3()
15 {
16 System.out.printf("3");
17 try
18 {
19 int sum = 4/0;
20 System.out.printf("4");
21 }
22 catch(ArithmeticException e)
23 {
24 System.out.printf("5");
25 }
26
27 System.out.printf("7");
28 }
29 public static void main(String[] args)
30 {
31 Test obj = new Test();
32 obj.m1();
33 }
34 }
35

Test.java

```java
package org.test;

public class Test {
    public static void main(String[] args)
    {
  try
  {
        System.out.printf("1");
      int sum = 9 / 0;
        System.out.printf("2");
  }
  catch(ArithmeticException e)
  {
        System.out.printf("3");
  }
   catch(Exception e)
  {
        System.out.printf("4");
  }
  finally
  {
     System.out.printf("5");
  }
    }
 }
```

Console

<terminated> Test (6) [Java Application] C:\Users\navee\.p2\pool\p

```
135
```

Test.java

```
1  package org.test;
2
3  import java.io.EOFException;
4  import java.io.IOException;
5
6  public class Test {
7      public static void main(String[] args)
8      {
9      try
10     {
11     System.out.printf("1");
12     int value = 10 / 0;
13      throw new IOException();
14      }
15     catch(EOFException e)
16        {
17   System.out.printf("2");
18     }
19   catch(ArithmeticException e)
20    {
21   System.out.printf("3");
22      }
23   catch(NullPointerException e)
24    {
25   System.out.printf("4");
26    }
27   catch(IOException e)
28    {
29   System.out.printf("5");
30    }
31      catch(Exception e)
32   {
33    System.out.printf("6");
34     }
35     }
36     }
37
```

Console

<terminate

13

DAY12-JAVA TASK

File Operation:
QUESTIONS(Practical)

QUESTION 1

NOTE: Create a new file
write the file with some 10 lines of text.

FileExercise.java

```
1  package org.hcl;
2
3  import java.io.File;
4  import java.io.IOException;
5
6  import org.apache.commons.io.FileUtils;
7
8  //import org.apache.commons.io.FileUtils;
9
10 public class FileExercise {
11     public static void main(String[] args) throws IOException {
12
13      String File1 ="C:\\Users\\navee\\OneDrive\\Desktop\\javatask\\sample10line.txt";
14     File f =new File(File1);
15     FileUtils.write(f, "hi i am learning java \n java is welcome \n python program \n java greens \n bigdat data science \n greens omr  \n selenium program \n sql program \
16     }
17
18 }
19
```

This PC › Desktop › javatask

Name	Date modified	Type	Size
sample10line	01-02-2022 14:17	Text Document	0 KB

sample10line - Notepad

File Edit Format View Help

```
hi i am learning java
 java is welcome
 python program
 java greens
 bigdat data science
 greens omr
 selenium program
 sql program
 greens tambaram
 greens adayar chennai
```

QUESTION 2

NOTE: Retrieve the text from the file
Check "java" word contains or not?

QUESTION 3

NOTE: Find the row count from the file.

FileExercise.java

```java
package org.hcl;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileReader;
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.util.Scanner;

import org.apache.commons.io.FileUtils;

//import org.apache.commons.io.FileUtils;

public class FileExercise {
    public static void main(String[] args) throws IOException {
        File file = new File("D://SampleFile.txt");
          FileInputStream fis = new FileInputStream(file);
          byte[] byteArray = new byte[(int)file.length()];
          fis.read(byteArray);
          String data = new String(byteArray);
          String[] stringArray = data.split("\r\n");
          System.out.println("Number of lines in the file are ::"+stringArray.length);
       }
    }
```

Console

```
<terminated> FileExercise [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x8
Number of lines in the file are ::7
```

QUESTION 4

NOTE: Print the last 5 lines from the file.

ExeDemo.java

```java
package org.hcl;

import java.io.File;
import java.io.IOException;
import java.io.LineNumberReader;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;

import org.apache.commons.io.input.ReversedLinesFileReader;

public class ExeDemo {

	public static void main(String[] args) {
		File file = new File("D:\\SampleFile.txt");
		int n_lines = 5;
		int counter = 0;
		ReversedLinesFileReader object = null;
		try {
			object = new ReversedLinesFileReader(file);
		} catch (IOException e) {
			// TODO Auto-generated catch block
			e.printStackTrace();
		}
		while(counter < n_lines) {
			try {
				System.out.println(object.readLine());
			} catch (IOException e) {
				// TODO Auto-generated catch block
				e.printStackTrace();
			}
			counter++;
		}
	}
}
```

Console

```
<terminated> ExeDemo [Java Appli
hi i am learning python10
 learning java9
hi i am java8
hi i am learning java7
hi i am learning java6
```

QUESTION 5

NOTE: Print the odd lines from the file.

```java
package org.hcl;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.IOException;

public class FileOperationPractice {
    //public static void main(String[] args) {
        public static void main(String[] args) throws IOException {
            File fileIn1 = new File("D:\\SampleFile.txt");

            try (BufferedReader bf = new BufferedReader(new FileReader(fileIn1))) {//SURROUND WITH try with resources FOR THE EXCEPTIONS

                String readLine;
                int line = 0;

                while ((readLine = bf.readLine()) != null) {
                    if (line % 2 != 0) {//CHECKING WHETHER THE ROW IS ODD
                        System.out.println(readLine);
                    }
                    line++;
                }

            }
        }
    }
```

QUESTION 6

NOTE: Count the number of dublicate words available in the file.

*FileOperationPractice.java

```java
package org.hcl;

import java.io.File;

public class FileOperationPractice {
public static void main(String[] args) {
    File file = new File("D:\\Java");

          boolean mkdir = file.mkdir();
          System.out.println();
}
}
```

QUESTION 7

NOTE: Check directory D:\Java is available or not.

FileOperationPractice.java

```java
package org.hcl;

import java.io.File;

public class FileOperationPractice {
    public static void main(String[] args) {


        File f =new File ("D:\\Java");
                boolean exists = f.exists();
                System.out.println(exists);
}
}
```

Console

```
<terminated> FileOperationPractice [Java Application] C:\Users\navee\.p
true
```

FileOperationPractice.java

```java
package org.hcl;

import java.io.File;

public class FileOperationPractice {
public static void main(String[] args) {


          try {
             File file = new File("D:\\Java");
             file.createNewFile();
             System.out.println(file.exists());
          } catch(Exception e) {
             e.printStackTrace();
          }
       }

}
```

Console

```
<terminated> FileOperationPractice [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclip
true
```

QUESTION 8

NOTE: Check directory D:\Java is available or not.

If not create new directory.

FileOperationPractice.java

```java
package org.hcl;

import java.io.File;

public class FileOperationPractice {
public static void main(String[] args) {


    File f =new File("D:\\Java\\Selenium\\Material");
    boolean mkdirs = f.mkdirs();
    System.out.println(mkdirs);
}
}
```

Console

```
<terminated> FileOperationPractice [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_6
true
```

QUESTION 9

NOTE: Create sub directory D:\Java\Selenium\Material.

QUESTION 10

NOTE: Print all the available files from an existing folder.

ExeDemo.java

```java
package org.hcl;

import java.io.BufferedReader;
import java.io.File;
import java.io.IOException;
import java.io.LineNumberReader;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.util.ArrayList;
import java.util.Iterator;

import org.apache.commons.io.input.ReversedLinesFileReader;

public class ExeDemo {


        public static void main(String[] args) {

            // creates a file object
            File file = new File("C:\\Users\\navee\\OneDrive\\Desktop");

            // returns an array of all files
            String[] fileList = file.list();

            for(String str : fileList) {
              System.out.println(str);
            }
          }
        }
```

Console

```
<terminated> ExeDemo [Java Application] C:\Users\navee\.p2\pool\plugins\org.eclipse.j
desktop.ini
eclipse - Shortcut.lnk
File operation
Folderfile
home.html
javatask
Microsoft Edge.lnk
navee.html
NAVEEN123
New Text Document.txt
selenium driver path.txt
SELENIUM2022
sqldeveloper - Shortcut.lnk
Subash Balachandran is inviting you to a scheduled Zoom meeting
syntax for java.txt
Zoom.lnk
```

4

```java
package org.hcl;

import java.io.File;
import java.io.IOException;
import java.io.LineNumberReader;
import java.nio.file.Files;
import java.nio.file.Path;
```

```
import java.nio.file.Paths;

import org.apache.commons.io.input.ReversedLinesFileReader;

public class ExeDemo {

		public static void main(String[] args) {
			File file = new File("D:\\SampleFile.txt");
			int n_lines = 5;
			int counter = 0;
			ReversedLinesFileReader object = null;
			try {
				object = new ReversedLinesFileReader(file);
			} catch (IOException e) {
				// TODO Auto-generated catch block
				e.printStackTrace();
			}
			while(counter < n_lines) {
			    try {
					System.out.println(object.readLine());
				} catch (IOException e) {
					// TODO Auto-generated catch block
					e.printStackTrace();
				}
			    counter++;
			}
		}
}
```

10

```
package org.hcl;

import java.io.BufferedReader;
import java.io.File;
import java.io.IOException;
import java.io.LineNumberReader;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.util.ArrayList;
import java.util.Iterator;

import org.apache.commons.io.input.ReversedLinesFileReader;

public class ExeDemo {


			public static void main(String[] args) {

			    // creates a file object
			    File file = new
File("C:\\Users\\navee\\OneDrive\\Desktop");

			    // returns an array of all files
			    String[] fileList = file.list();

			    for(String str : fileList) {
```

```
                        System.out.println(str);
                    }
                }
            }
```

9

```
package org.hcl;

import java.io.File;

public class FileOperationPractice {
public static void main(String[] args) {


       File f =new File("D:\\Java\\Selenium\\Material");
       boolean mkdirs = f.mkdirs();
       System.out.println(mkdirs);
}
}
```

7

```
package org.hcl;

import java.io.File;

public class FileOperationPractice {
public static void main(String[] args) {


             try {
                File file = new File("D:\\Java");
                file.createNewFile();
                System.out.println(file.exists());
             } catch(Exception e) {
                e.printStackTrace();
             }
          }

}
```

8.

```
package org.hcl;

import java.io.File;

public class FileOperationPractice {
public static void main(String[] args) {
       File file = new File("D:\\Java");

             boolean mkdir = file.mkdir();
             System.out.println();
}
}
```

7.

```java
package org.hcl;

import java.io.File;

public class FileOperationPractice {
	public static void main(String[] args) {


		File f =new File ("D:\\Java1");
				boolean exists = f.exists();
				System.out.println(exists);
}
}
```

5.

```java
package org.hcl;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.IOException;

public class FileOperationPractice {
	//public static void main(String[] args) {
		public static void main(String[] args) throws IOException {
		  File fileIn1 = new File("D:\\SampleFile.txt");

		  try (BufferedReader bf = new BufferedReader(new
FileReader(fileIn1))) {//SURROUND WITH try with resources FOR THE EXCEPTIONS

			String readLine;
			int line = 0;

			while ((readLine = bf.readLine()) != null) {
			    if (line % 2 != 0) {//CHECKING WHETHER THE ROW IS ODD
				System.out.println(readLine);
			    }
			    line++;
			}

		  }
	    }
	}
```