**CHALLENGE 1:**

/\*\*

@author: Chanpreet

@application: using setter and getter methods for instance variables

@date: 16 March,2023

@time:4.00 pm

\*/

public class CollegeCourse

{

    int courseReferenceNumber;    // instance variables

    int courseNumber;

// default constructor

public CollegeCourse()

{

}

 // Constructor that takes two arguments

public CollegeCourse(int courseReferenceNumber, int courseNumber)

{

    this.courseReferenceNumber = courseReferenceNumber;

    this.courseNumber = courseNumber;

}

// setter and getter method for each field

public void setcourseReferenceNumber(int courseReferenceNumber)

{

    this.courseReferenceNumber = courseReferenceNumber;

}

public int getcourseReferenceNumber()

{

    return courseReferenceNumber;

}

public void setcourseNumber(int courseNumber)

{

    this.courseNumber = courseNumber;

}

public int getcourseNumber()

{

    return courseNumber;

}

}

**CHALLENGE 2:**

/\*\*

@author: Chanpreet

@application: using main method and printing data and also using try and catch block

@date: 16 March,2023

@time:4.00 pm

\*/

import java.util.Random;

import java.util.Scanner;

public class CollegeCourseTestHarness

{

    public static void main(String[] args)

    {

    // Creating a two dimensional array with 10 columns and 200 rows

      CollegeCourse c1[][] = new CollegeCourse[200][10];

    // Fill the first 100 rows with CollegeCourse Objects in which the course reference number and course number are set to the default value of zero.

     for(int i=0; i<100; i++){

       for(int j=0; j<10; j++)

      {

        c1[i][j]= new CollegeCourse();

        c1[i][j].setcourseReferenceNumber(0);

        c1[i][j].setcourseNumber(0);

      }

     }

     // Fill the second 100 rows with CollegeCourse Objects in which the course reference number and course number are set to the default value of 9999.

     for(int i=100; i<200; i++)

     {

       for(int j=0; j<10; j++)

      {

        c1[i][j]= new CollegeCourse();

        c1[i][j].setcourseReferenceNumber(9999);

        c1[i][j].setcourseNumber(9999);

      }

     }

   // printing the data

      for(int i=0; i<200; i++)

      {

        for(int j=0; j<10; j++)

      {

     System.out.println("Course at row " +i + " column  " +j + ":");

     System.out.println("Course Reference Number is: "  +c1[i][j].getcourseReferenceNumber());

     System.out.println(" Course Number is: " +c1[i][j].getcourseNumber());

     System.out.println("-------------------------------------------");

    }

      }

    // using random function

    Random random= new Random();

     for( int i=0; i<200; i++) {

        for(int j=0; j<10; j++)

      {

        int courseNumber = random.nextInt(10000);

        String CourseNumber = String.format("%05d", courseNumber);

      c1[i][j].setcourseReferenceNumber(9734263);

     System.out.println( "Course Reference Number is: " +c1[i][j].getcourseReferenceNumber());

     System.out.println(CourseNumber);

     System.out.println("-------------------------------------------");

    }

     }

    for(int i=0; i<200; i++){

        for(int j=0; j<10; j++)

      {

      System.out.println("Course at row " +i + " column " +j + ":");

     System.out.println("Course Reference Number is: " + c1[i][j].getcourseReferenceNumber());

     System.out.println(" Course Number is: " +c1[i][j].getcourseNumber());

     System.out.println("-------------------------------------------");

    }

}

Scanner sc= new Scanner(System.in);

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

int rows= sc.nextInt();

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

int columns= sc.nextInt();

 try{

  System.out.println(c1[rows][columns]);

 }

 catch(Exception e){

  System.out.println("The exception is: " + e.getMessage());

 }

 finally{

  System.out.println("You need to provide a valid set of numbers");

 }

}

}

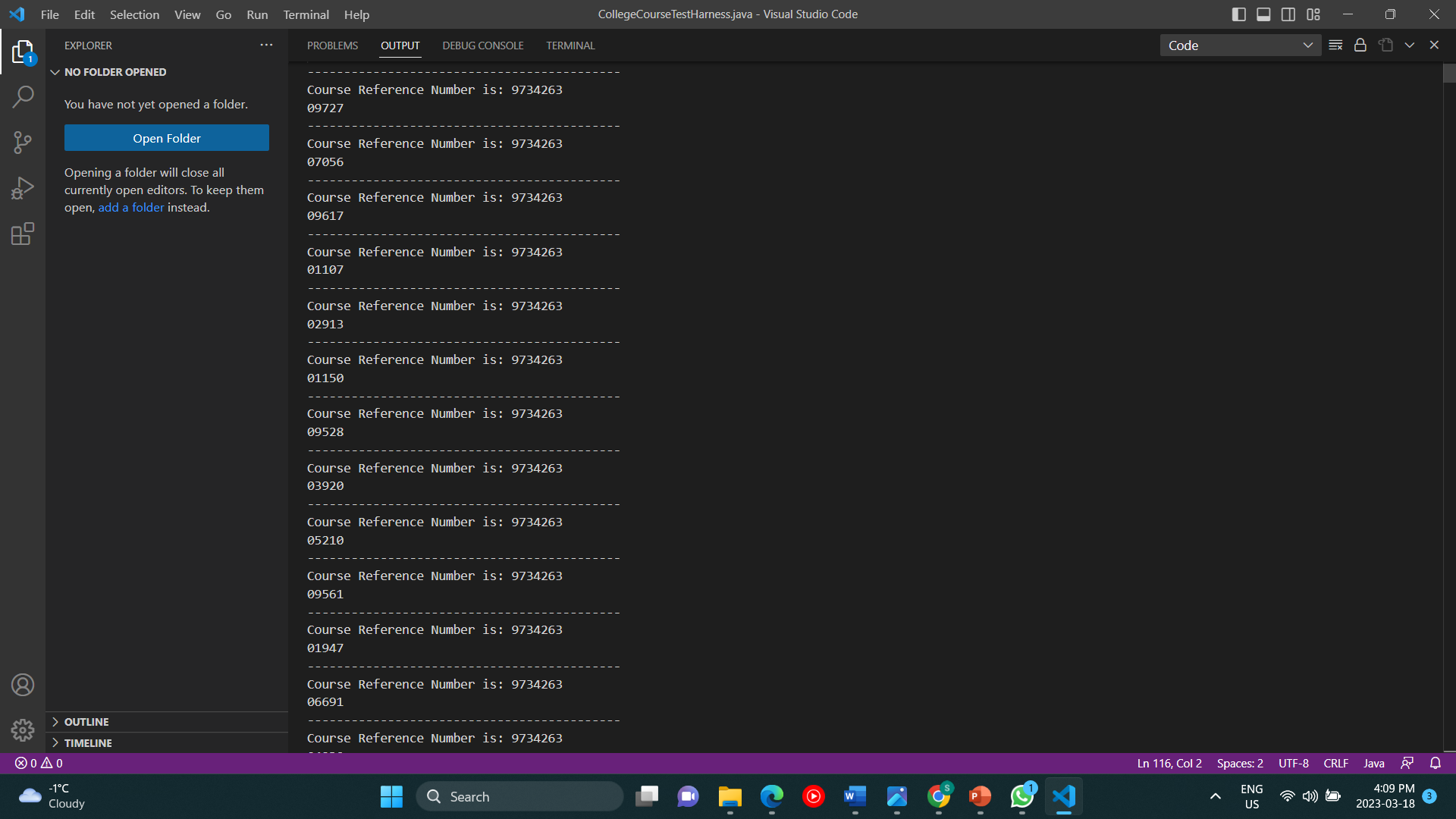
**OUTPUT:**

**A screenshot of a computer

Description automatically generated with medium confidence**

**A screenshot of a computer

Description automatically generated with medium confidence**

****

**A screenshot of a computer

Description automatically generated with medium confidence**