**Assignment2(Dt: 5/09/24)(Sandeep Sir)**

**1. Working with java.lang.Boolean**

**b.** Declare a method-local variable status of type boolean with the value true and convert it to a String using the toString method. (Hint: Use Boolean.toString(Boolean) ).

public class short\_1 {

public static void main(String[] args) {

**boolean** status=**true**;

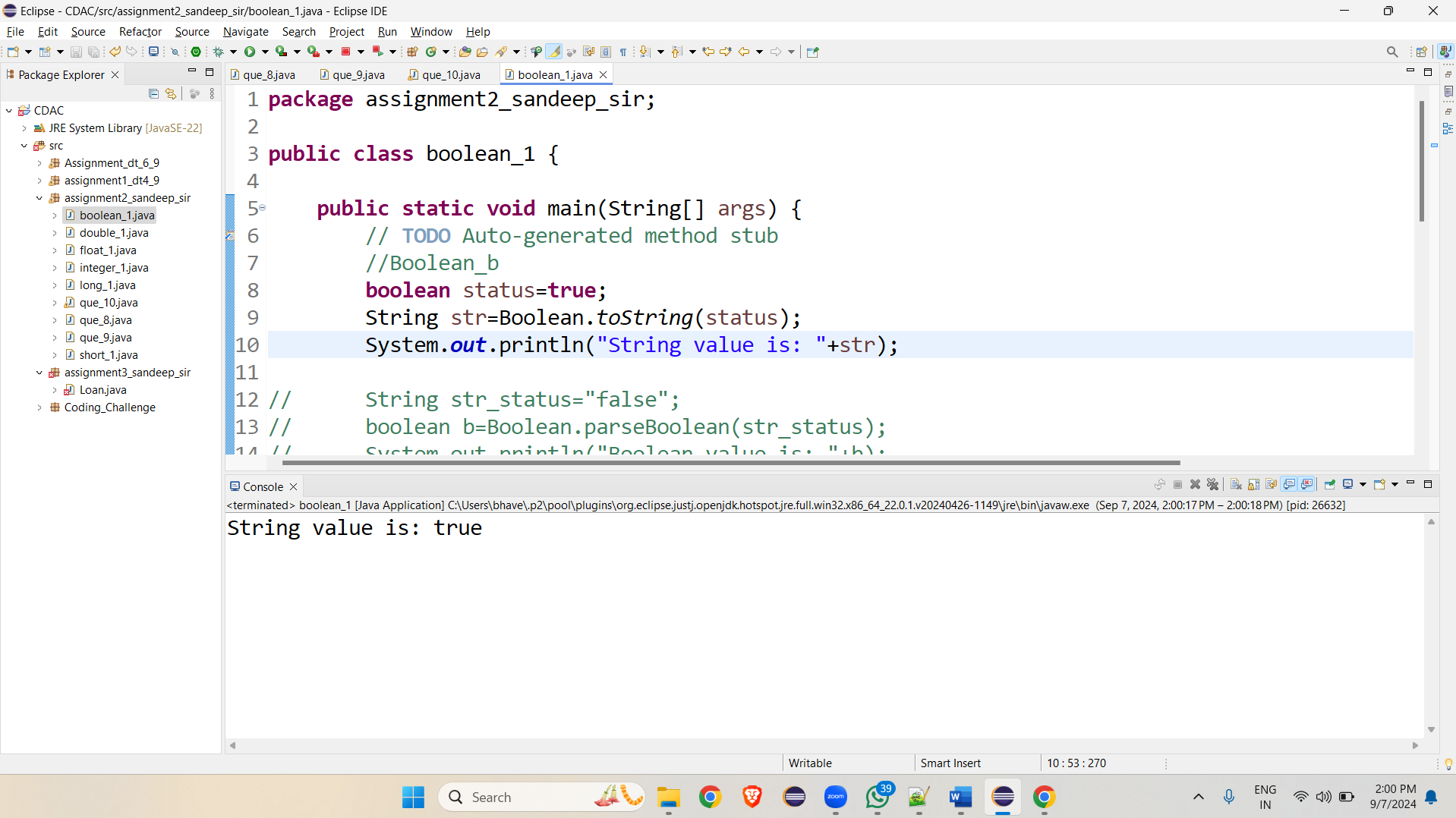
String str=Boolean.*toString*(status);

System.***out***.println("String value is: "+str);

}

}

O/P:



**c.** Declare a method-local variable strStatus of type String with the value "true" and convert it to a boolean using the parseBoolean method. (Hint: Use Boolean.parseBoolean(String)).

public class short\_1 {

public static void main(String[] args) {

String str\_status="false";

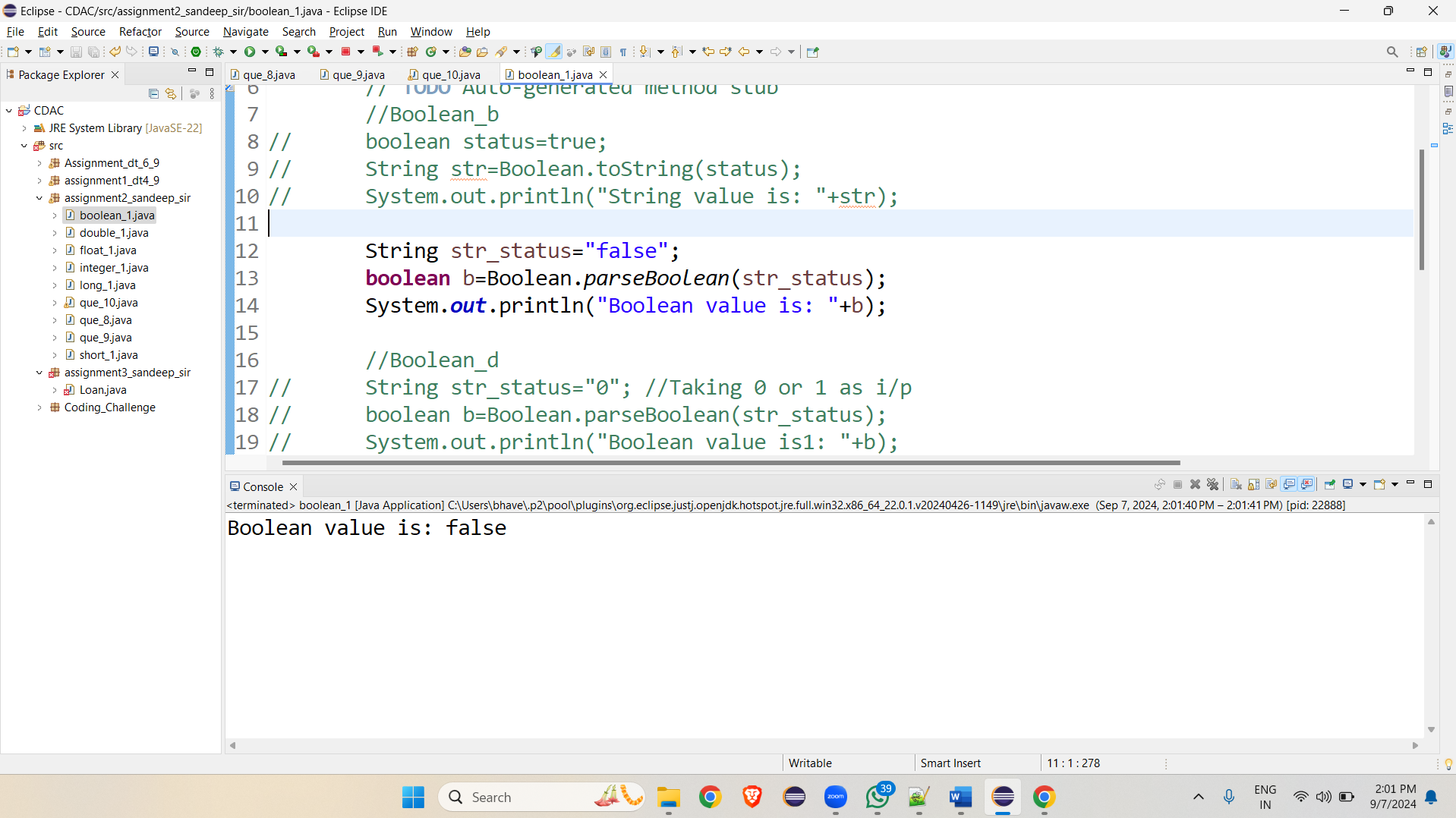
**boolean** b=Boolean.*parseBoolean*(str\_status);

System.***out***.println("Boolean value is: "+b);

}

}

O/P



**d.** Declare a method-local variable strStatus of type String with the value "1" or "0" and attempt to convert it to a boolean. (Hint: parseBoolean method will not work as expected with "1" or "0").

**public class short\_1 {**

**public static void main(String[] args) {**

**String str\_status="0"; //Taking 0 or 1 as i/p**

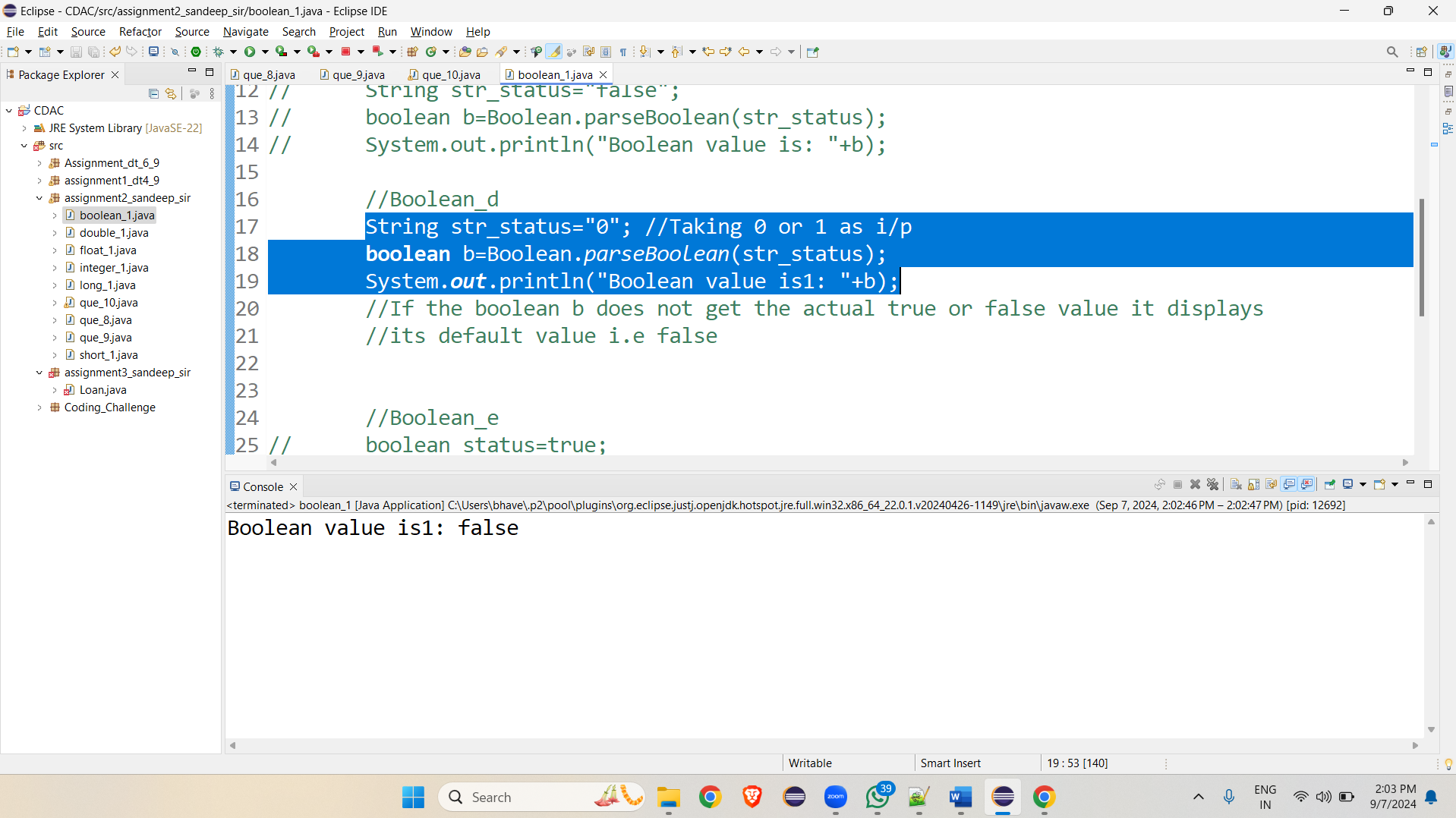
**boolean b=Boolean.*parseBoolean*(str\_status);**

**System.*out*.println("Boolean value is1: "+b);**

**}**

**}**

**O/P:**

****

**e.** Declare a method-local variable status of type boolean with the value true and convert it to the corresponding wrapper class using Boolean.valueOf(). (Hint: Use Boolean.valueOf(boolean)).

public class short\_1 {

public static void main(String[] args) {

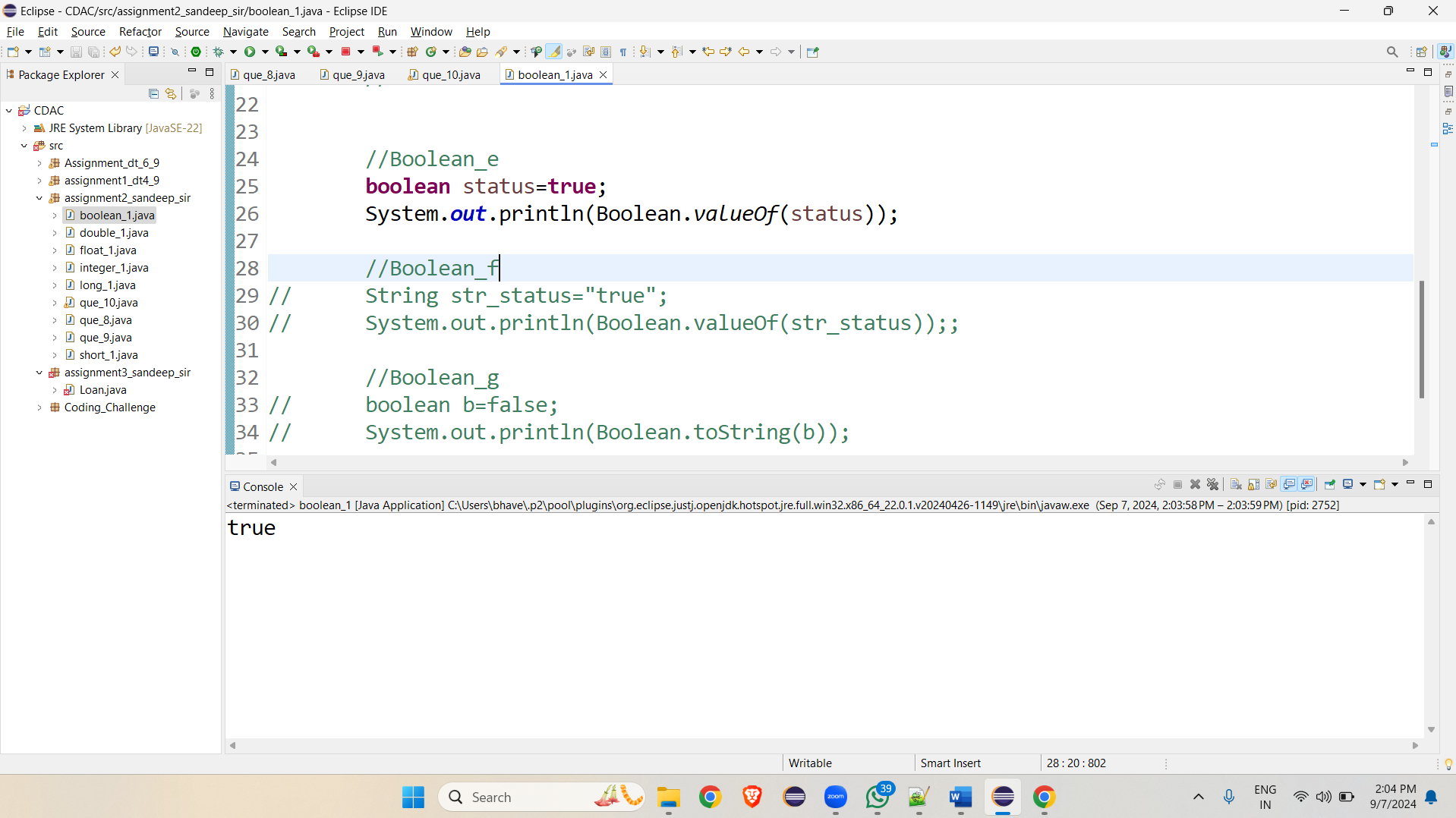
**boolean** status=**true**;

System.***out***.println(Boolean.*valueOf*(status));

}

}

O/p



**f.** Declare a method-local variable strStatus of type String with the value "true" and convert it to the corresponding wrapper class using Boolean.valueOf(). (Hint: Use Boolean.valueOf(String)).

public class short\_1 {

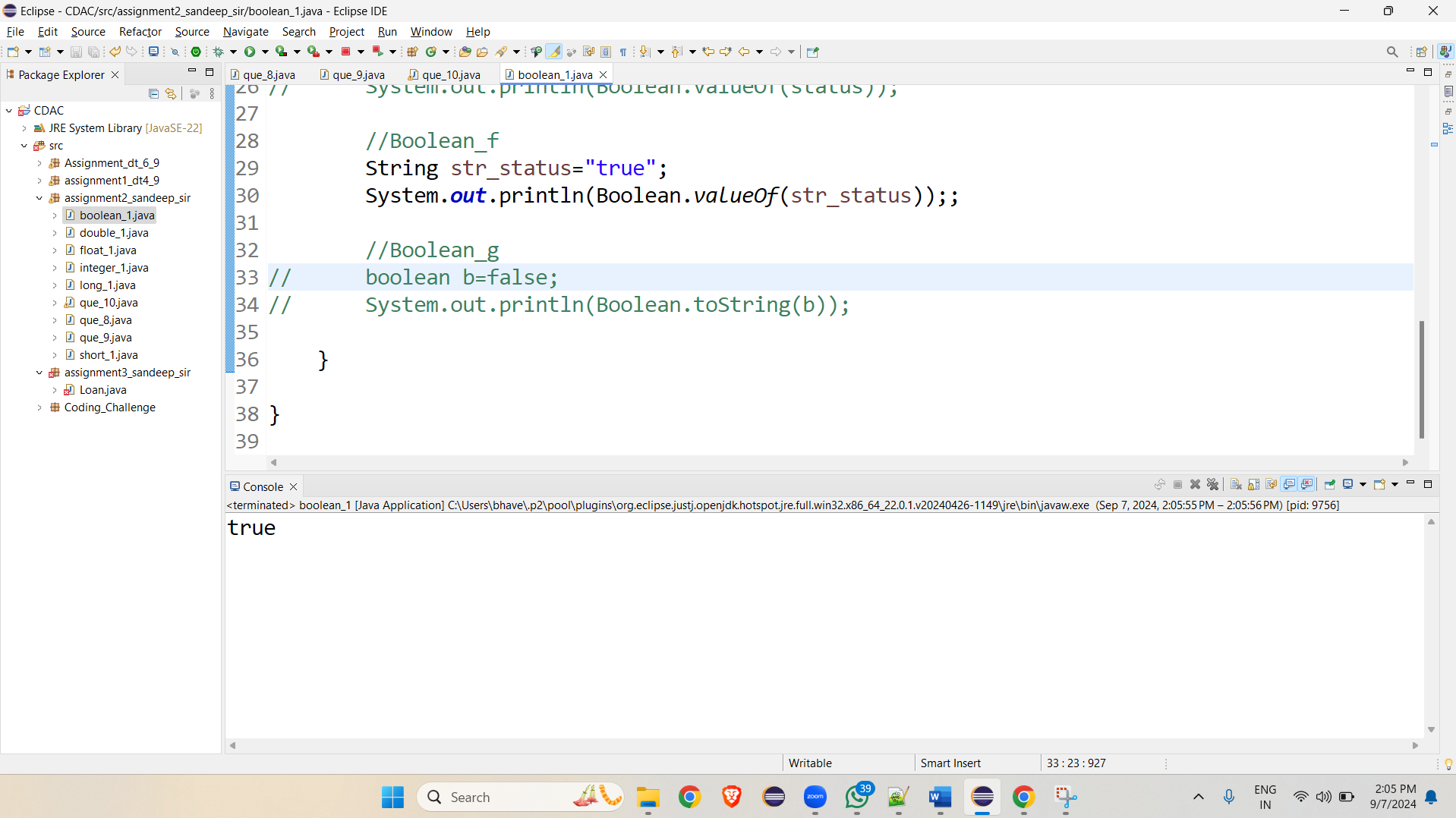
public static void main(String[] args) {

String str\_status="true";

System.***out***.println(Boolean.*valueOf*(str\_status));;

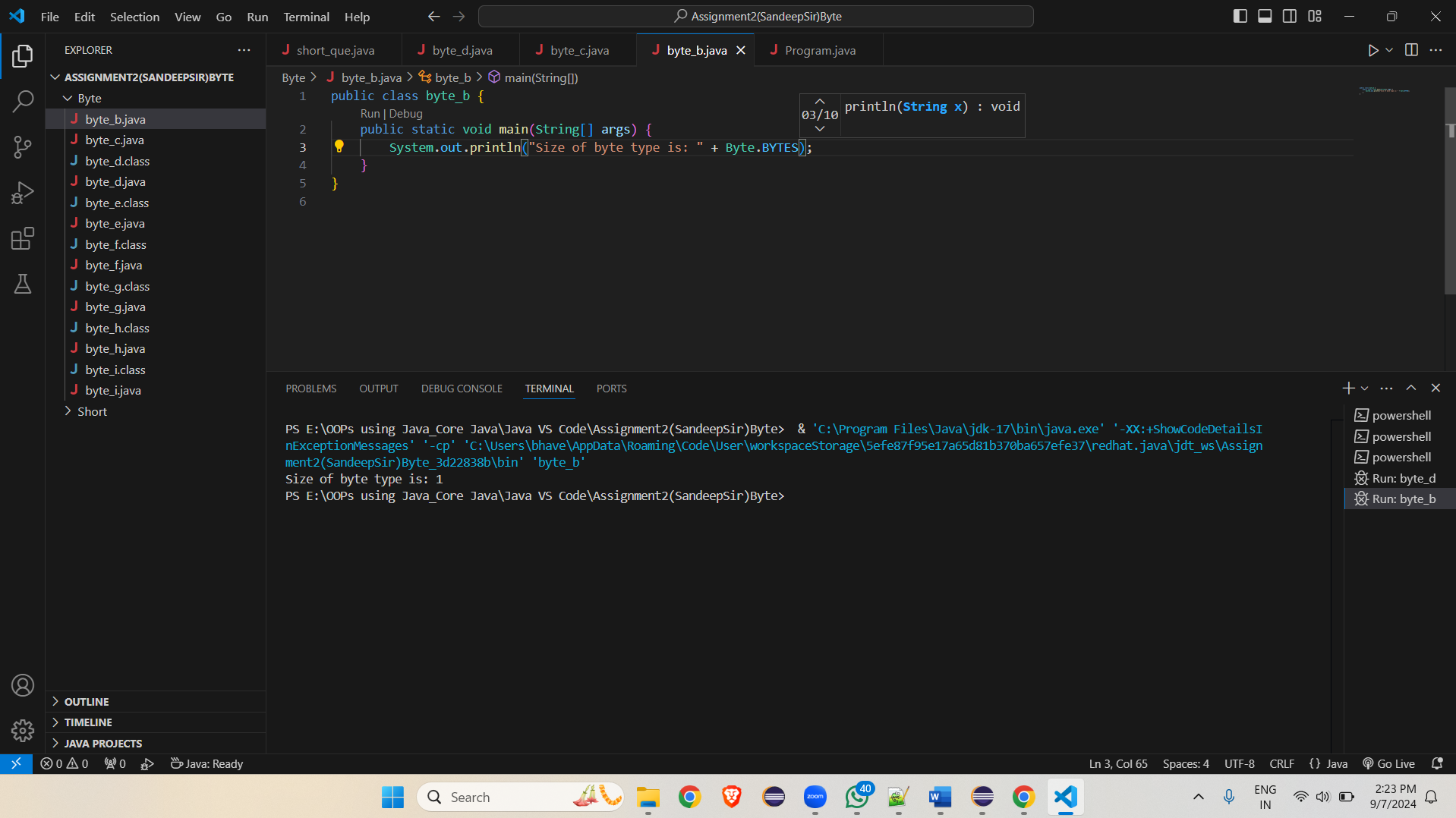
}

}

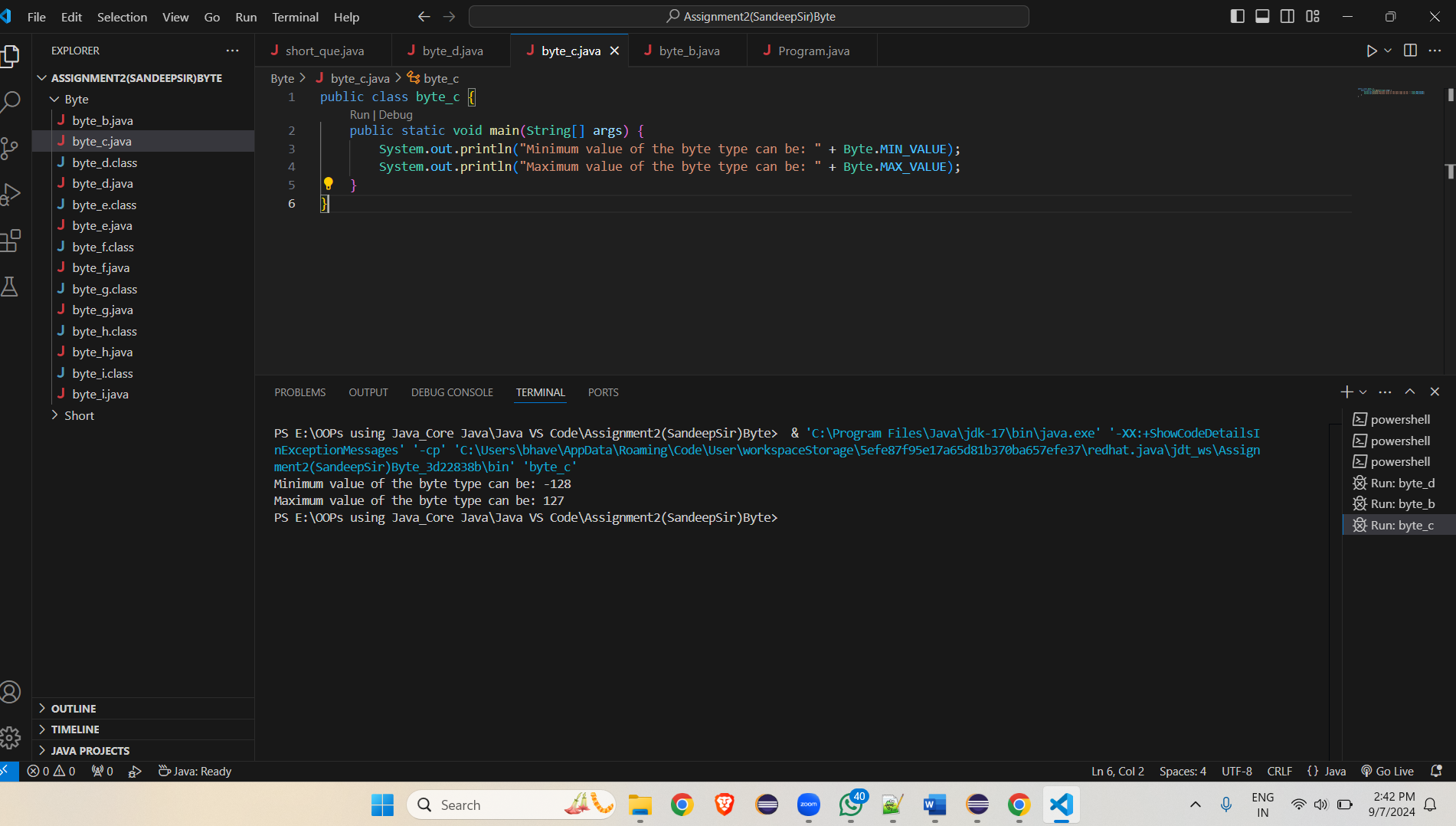


**Working with java.lang.Byte:**

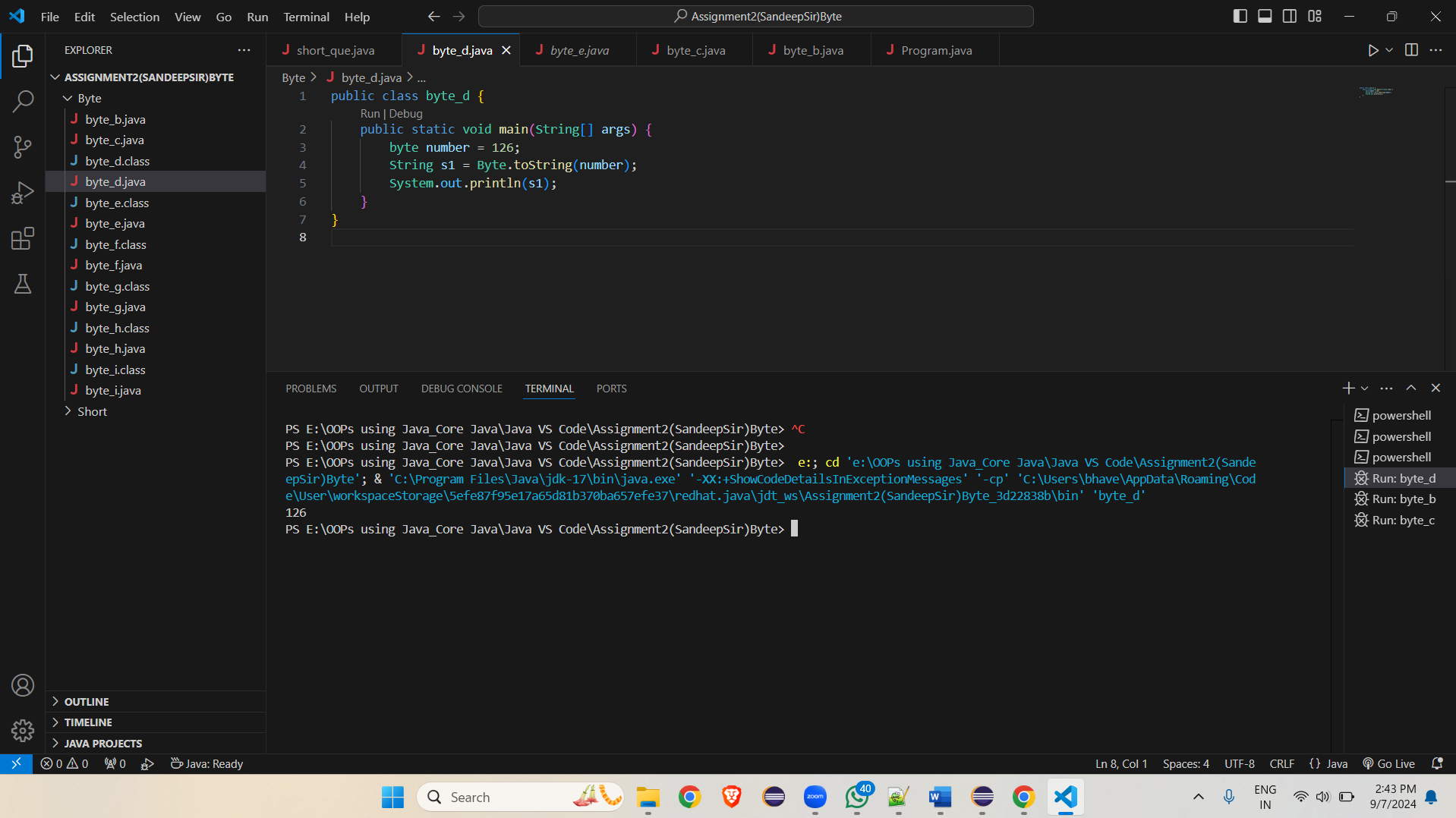
**b.** Write a program to test how many bytes are used to represent a byte value using the BYTES field. (Hint: Use Byte.BYTES).



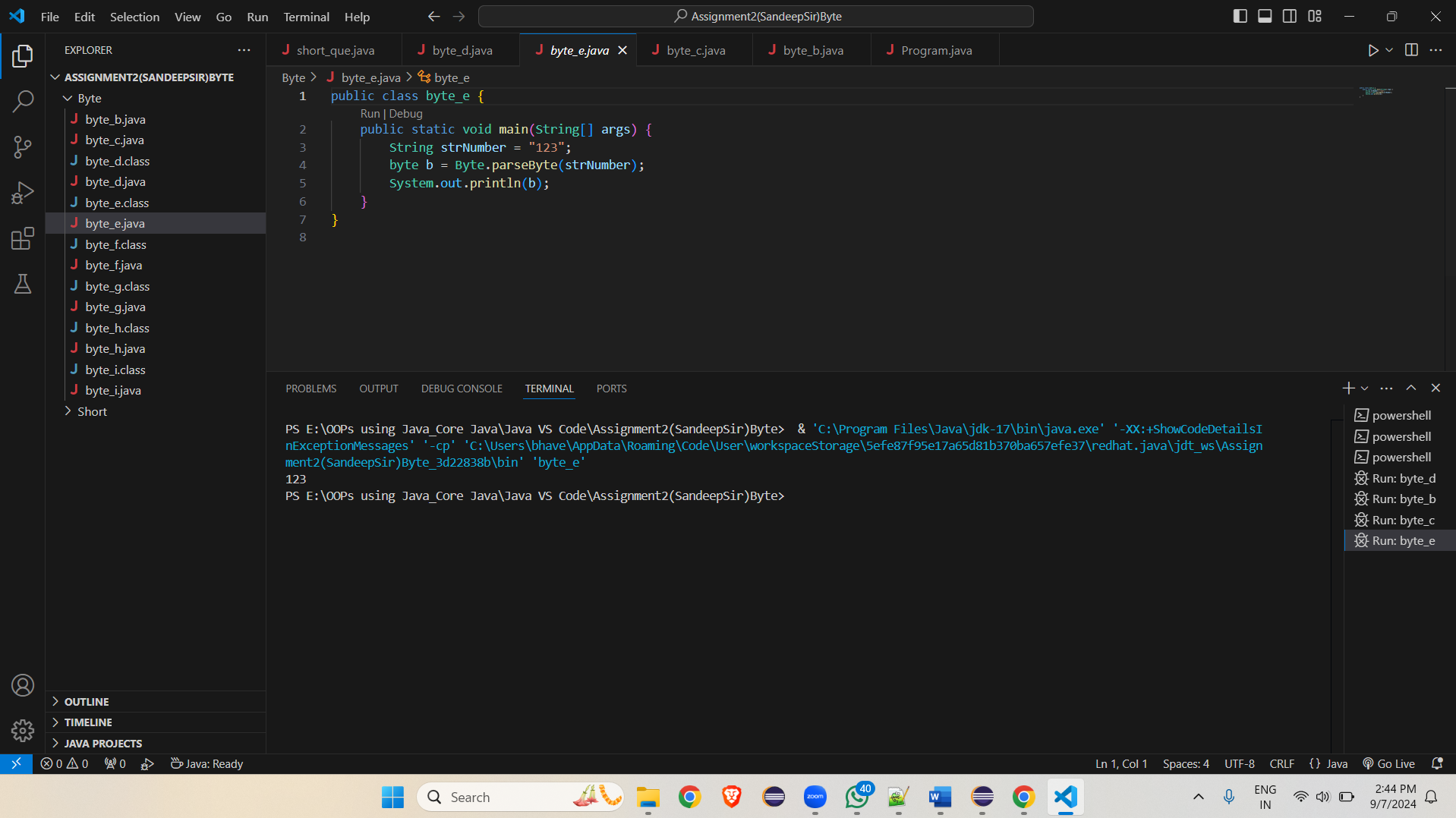
**c.** Write a program to find the minimum and maximum values of byte using the MIN\_VALUE and MAX\_VALUE fields. (Hint: Use Byte.MIN\_VALUE and Byte.MAX\_VALUE).



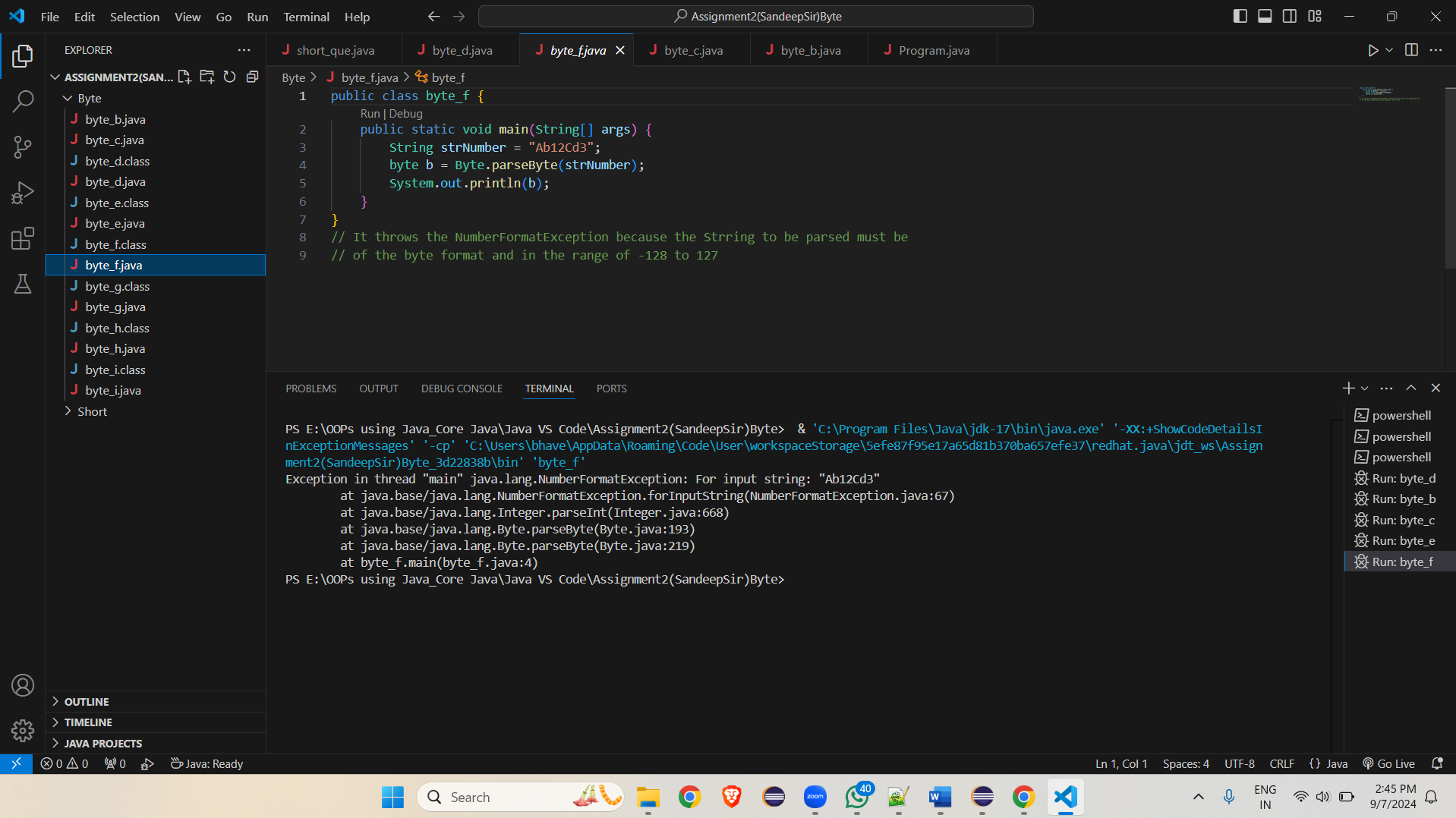
**d.** Declare a method-local variable number of type byte with some value and convert it to a String using the toString method. (Hint: Use Byte.toString(byte)).



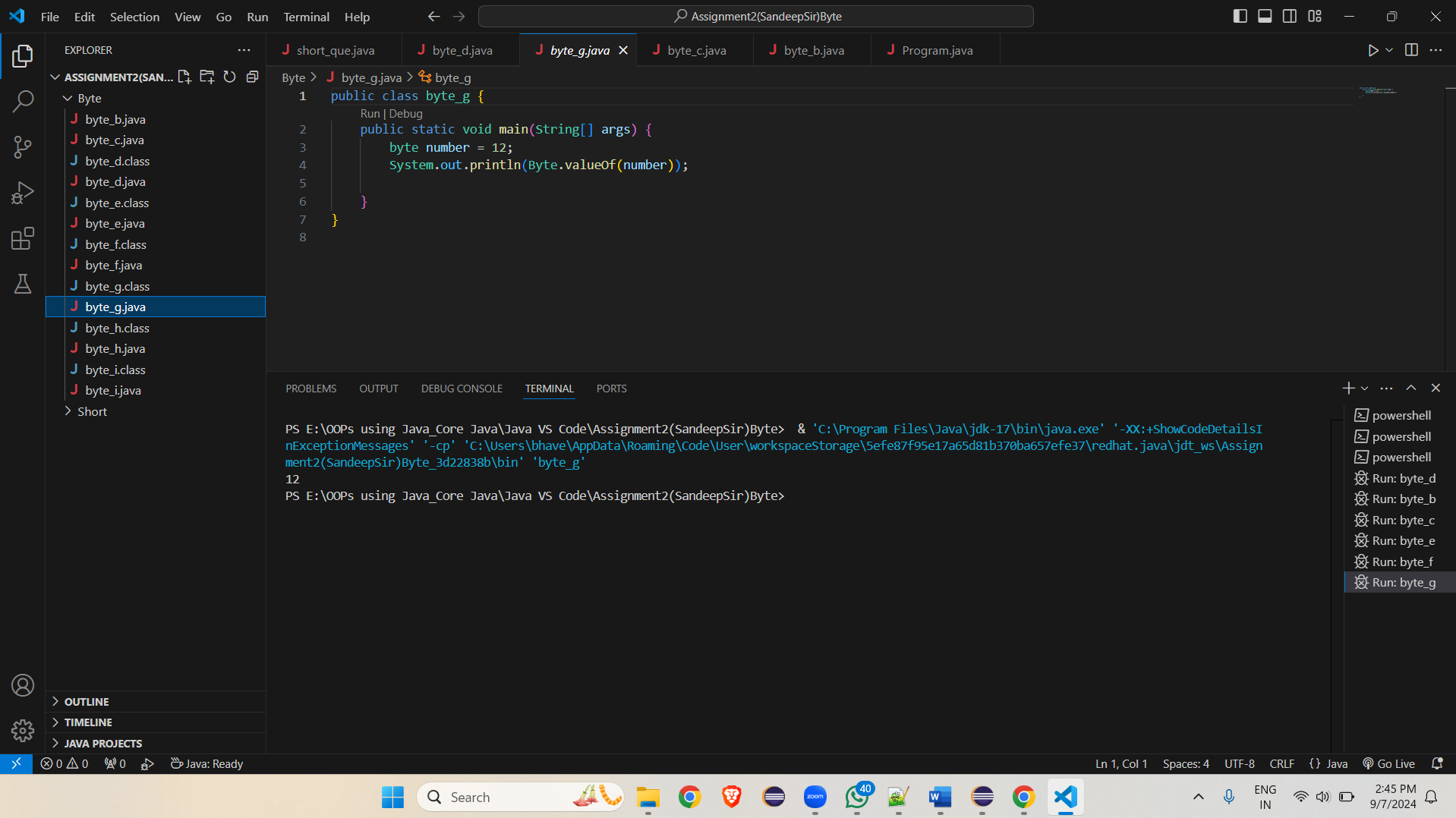
**e.** Declare a method-local variable strNumber of type String with some value and convert it to a byte value using the parseByte method. (Hint: Use Byte.parseByte(String)).



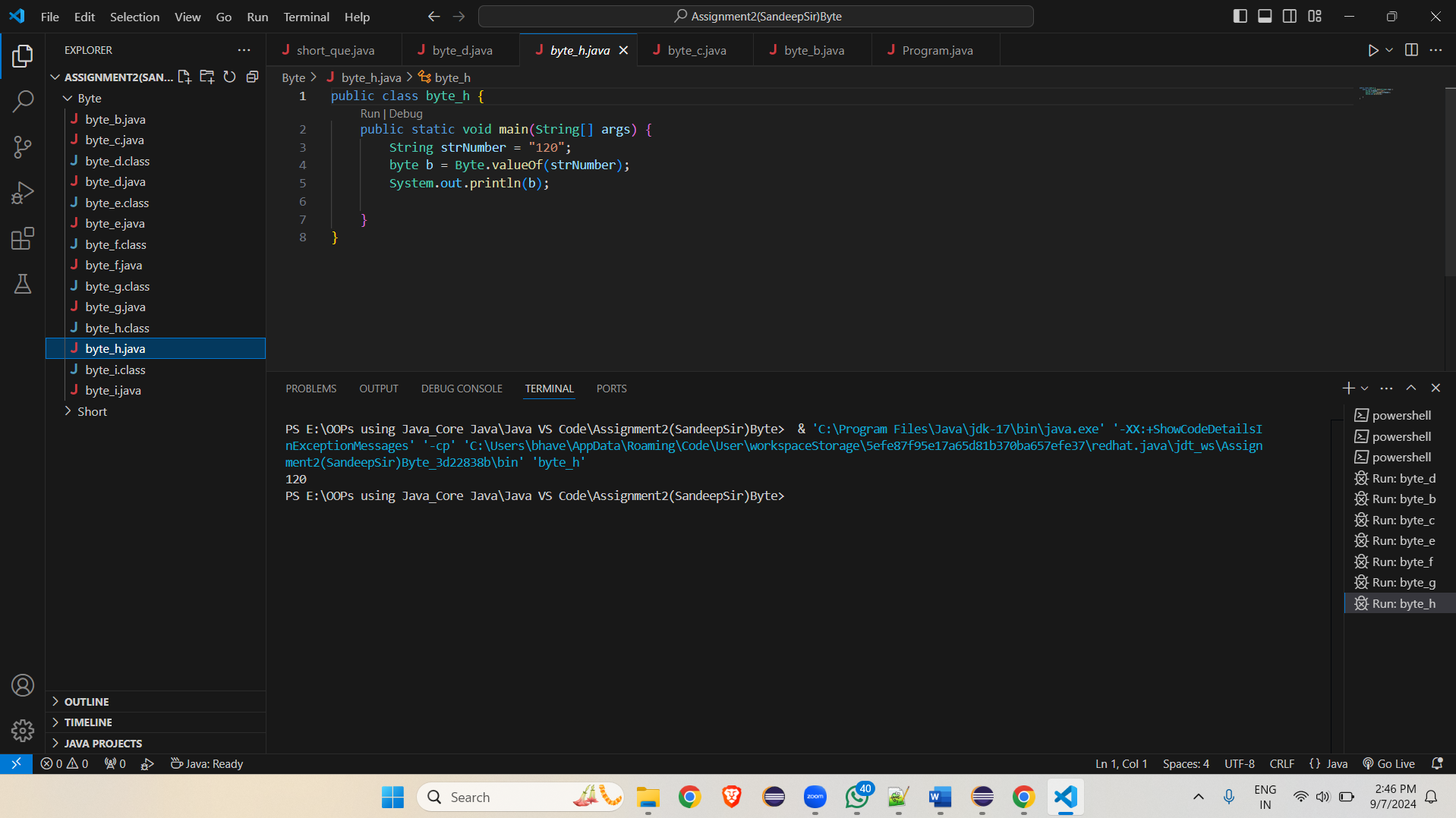
**f.** Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a byte value. (Hint: parseByte method will throw a NumberFormatException).



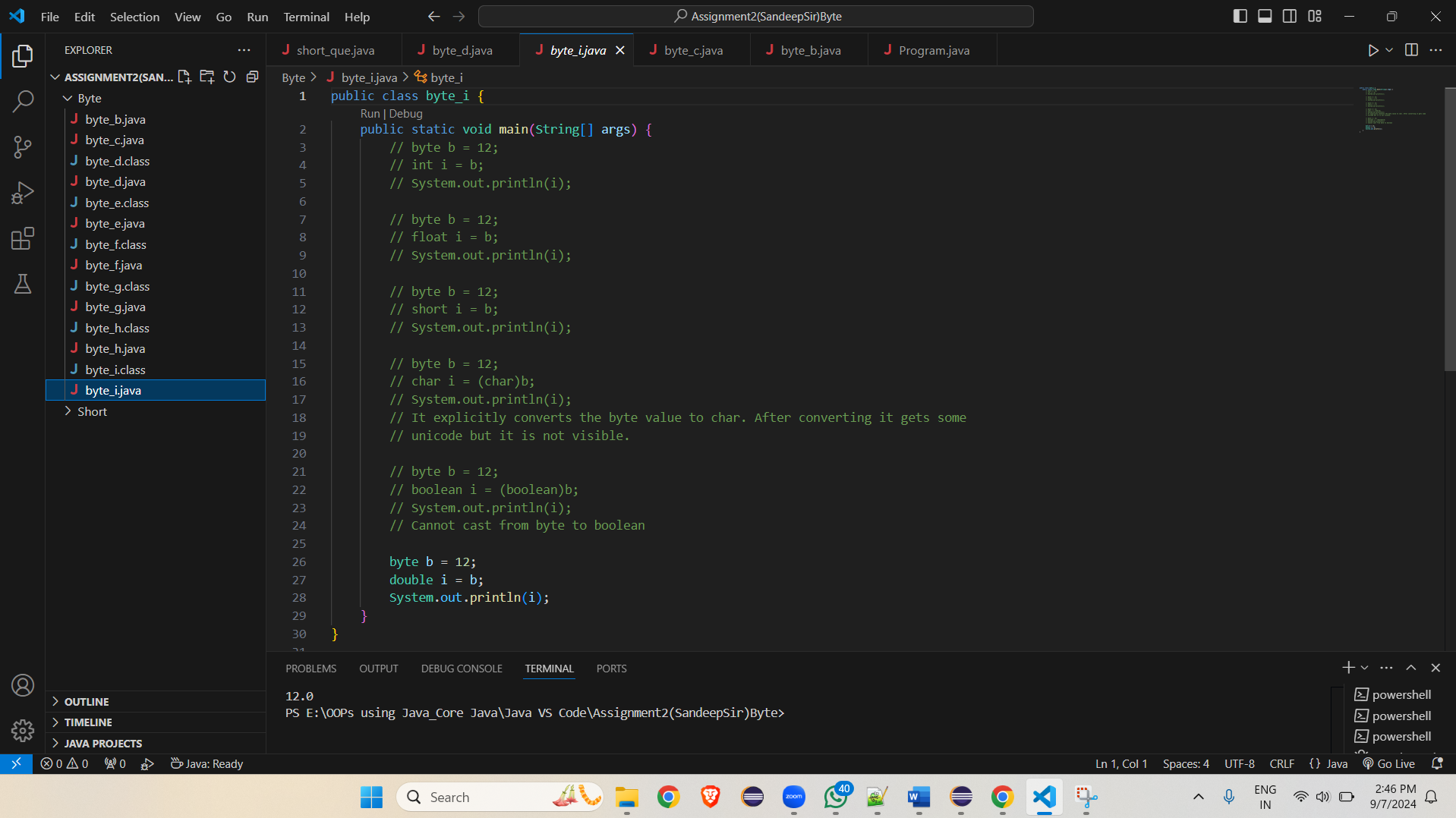
**g.** Declare a method-local variable number of type byte with some value and convert it to the corresponding wrapper class using Byte.valueOf(). (Hint: Use Byte.valueOf(byte)).



**h.** Declare a method-local variable strNumber of type String with some byte value and convert it to the corresponding wrapper class using Byte.valueOf(). (Hint: Use Byte.valueOf(String)).



**i.** Experiment with converting a byte value into other primitive types or vice versa and observe the results.



**Working with** java.lang.Short

**b.** Write a program to test how many bytes are used to represent a short value using the BYTES field. (Hint: Use Short.BYTES).

public class short\_1 {

public static void main(String[] args) {

System.out.println("no of bytes used to represent a short value is: "+Short.BYTES);

}

}

A screenshot of a computer

Description automatically generated

**c.** Write a program to find the minimum and maximum values of short using the MIN\_VALUE and MAX\_VALUE fields. (Hint: Use Short.MIN\_VALUE and Short.MAX\_VALUE).

public class short\_1 {

public static void main(String[] args) {

System.out.println("Minimum value of short can be: "+Short.MIN\_VALUE);

System.out.println("Maximum value of short can be: "+Short.MAX\_VALUE);

}

}

A screenshot of a computer

Description automatically generated

**d.** Declare a method-local variable number of type short with some value and convert it to a String using the toString method. (Hint: Use Short.toString(short)).

**public class short\_1 {**

**public static void main(String[] args) {**

**short number=123;**

**System.out.println(Short.toString(number));**

**}**

**}**

**A screenshot of a computer

Description automatically generated**

**e.** Declare a method-local variable strNumber of type String with some value and convert it to a short value using the parseShort method. (Hint: Use Short.parseShort(String)).

**public class short\_1 {**

**public static void main(String[] args) {**

**String strNumber = "12334";**

**System.out.println(Short.parseShort(strNumber));**

**}**

**}**

**A screenshot of a computer

Description automatically generated**

**f.** Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a short value. (Hint: parseShort method will throw a NumberFormatException).

**public class short\_1 {**

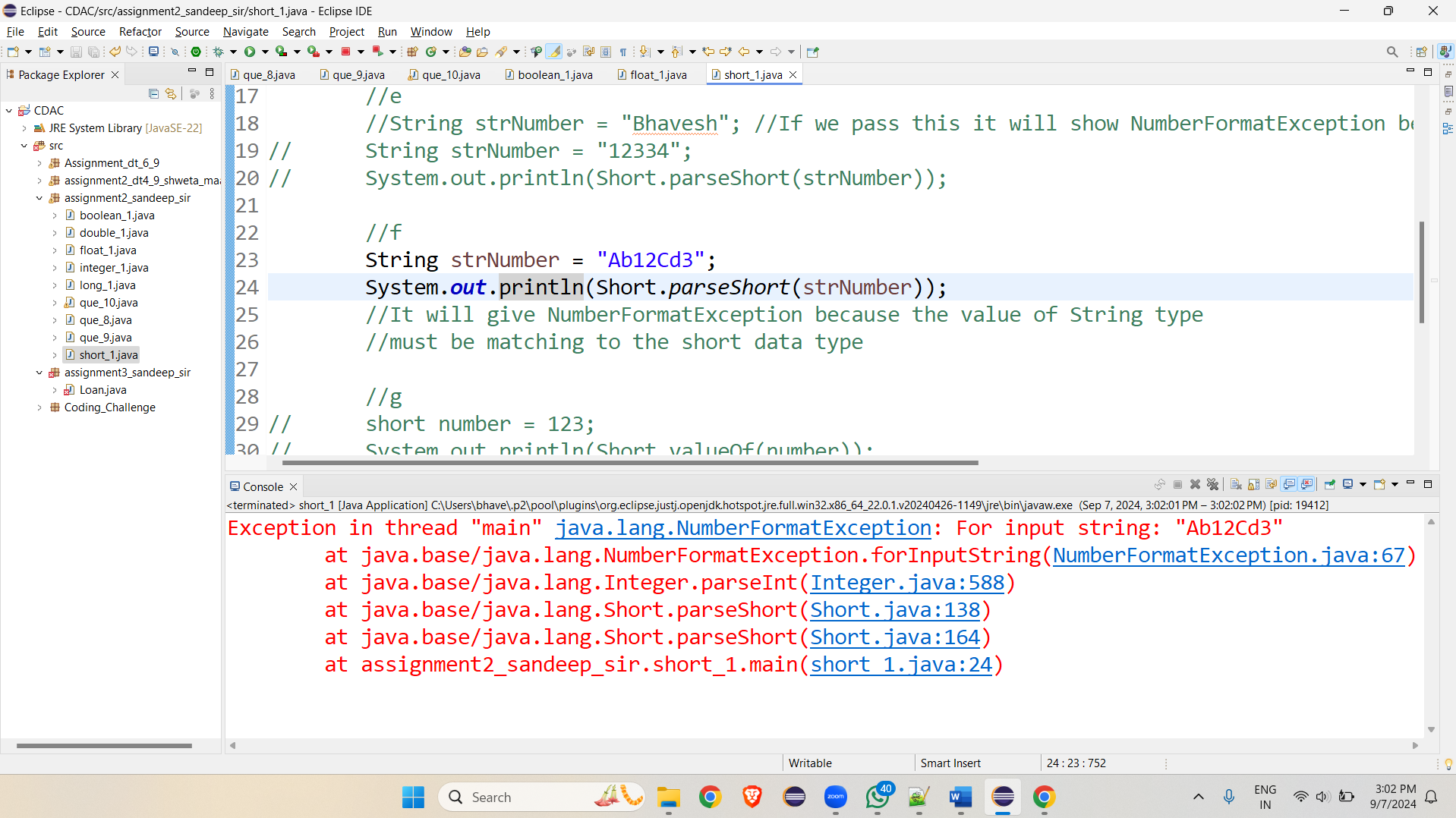
**public static void main(String[] args) {**

**String strNumber = "Ab12Cd3";**

**System.out.println(Short.parseShort(strNumber));**

**}**

**}**

****

**g.** Declare a method-local variable number of type short with some value and convert it to the corresponding wrapper class using Short.valueOf(). (Hint: Use Short.valueOf(short)).

**public class short\_1 {**

**public static void main(String[] args) {**

**short number = 123;**

**System.out.println(Short.valueOf(number));**

**}**

**}**

**A screenshot of a computer

Description automatically generated**

**h.** Declare a method-local variable strNumber of type String with some short value and convert it to the corresponding wrapper class using Short.valueOf(). (Hint: Use Short.valueOf(String)).

**public class short\_1 {**

**public static void main(String[] args) {**

**String strNumber = "25367";**

**System.out.println(Short.valueOf(strNumber));**

**}**

**}**

**A screenshot of a computer

Description automatically generated**

**(Other I have uploaded .java files directly on github).**