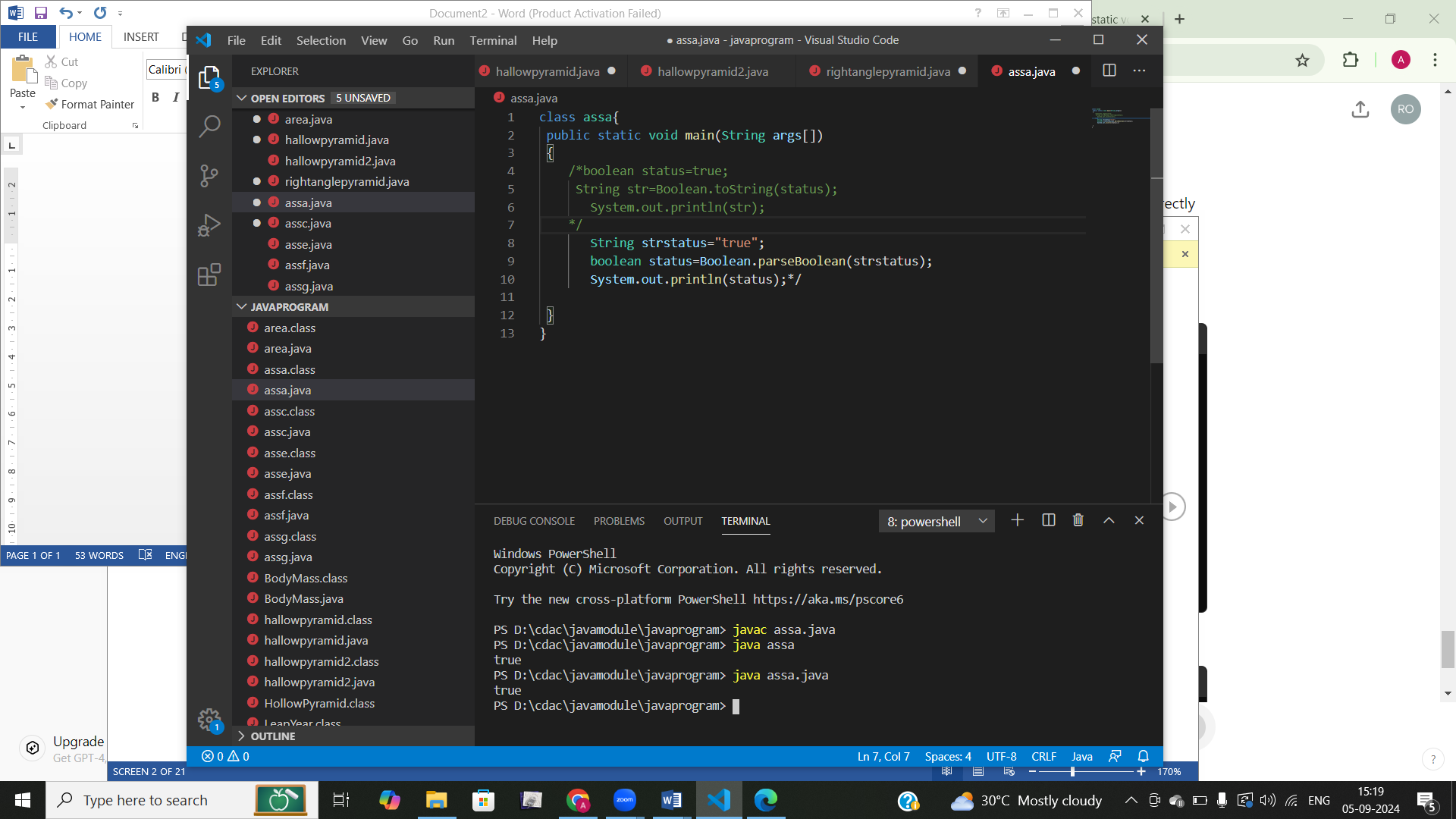
#### ****1. Working with**** java.lang.Boolean

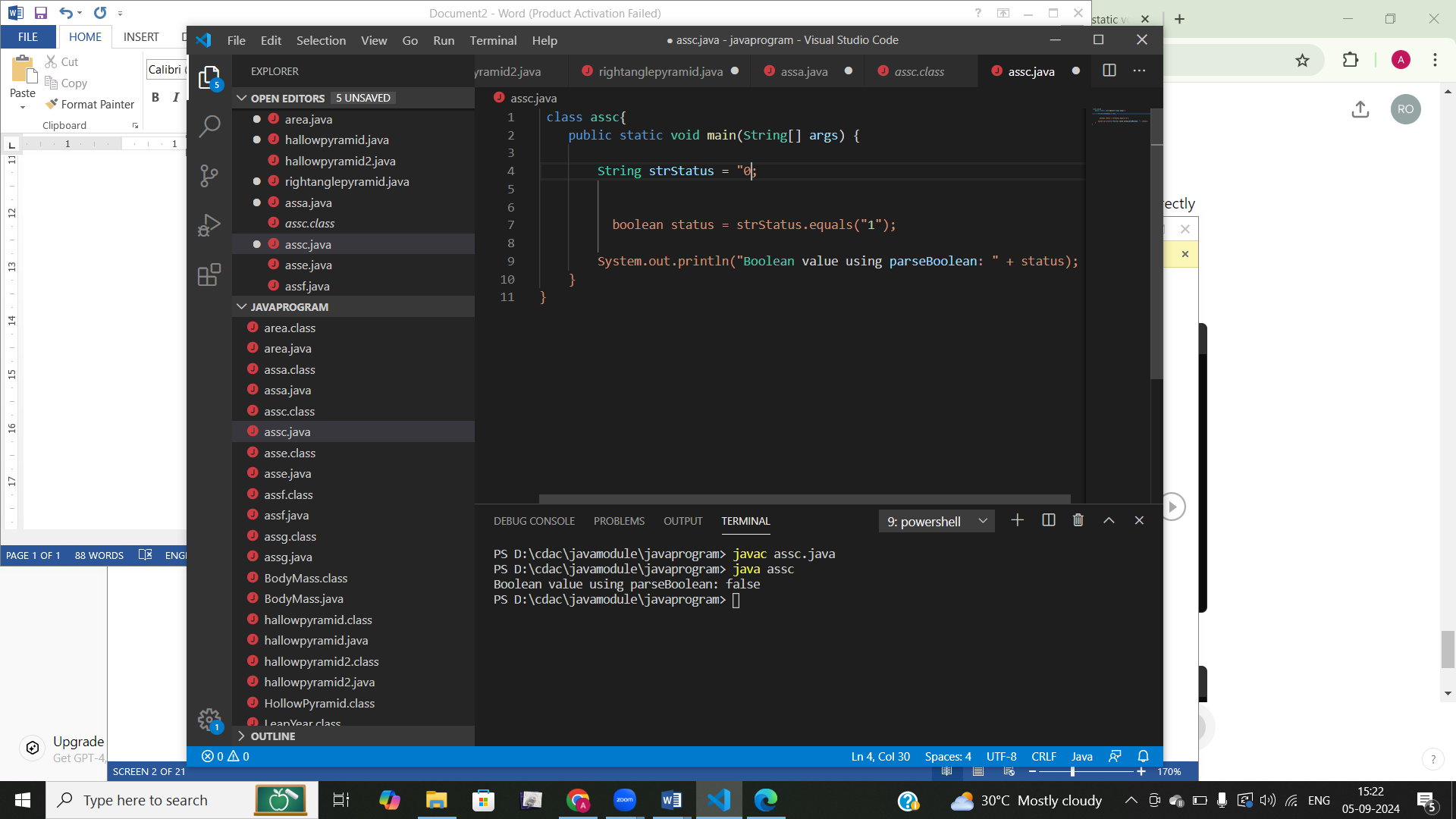
**a.** Explore the [Java API documentation for java.lang.Boolean](https://docs.oracle.com/javase/8/docs/api/java/lang/Boolean.html) and observe its modifiers and super types.

**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) ).

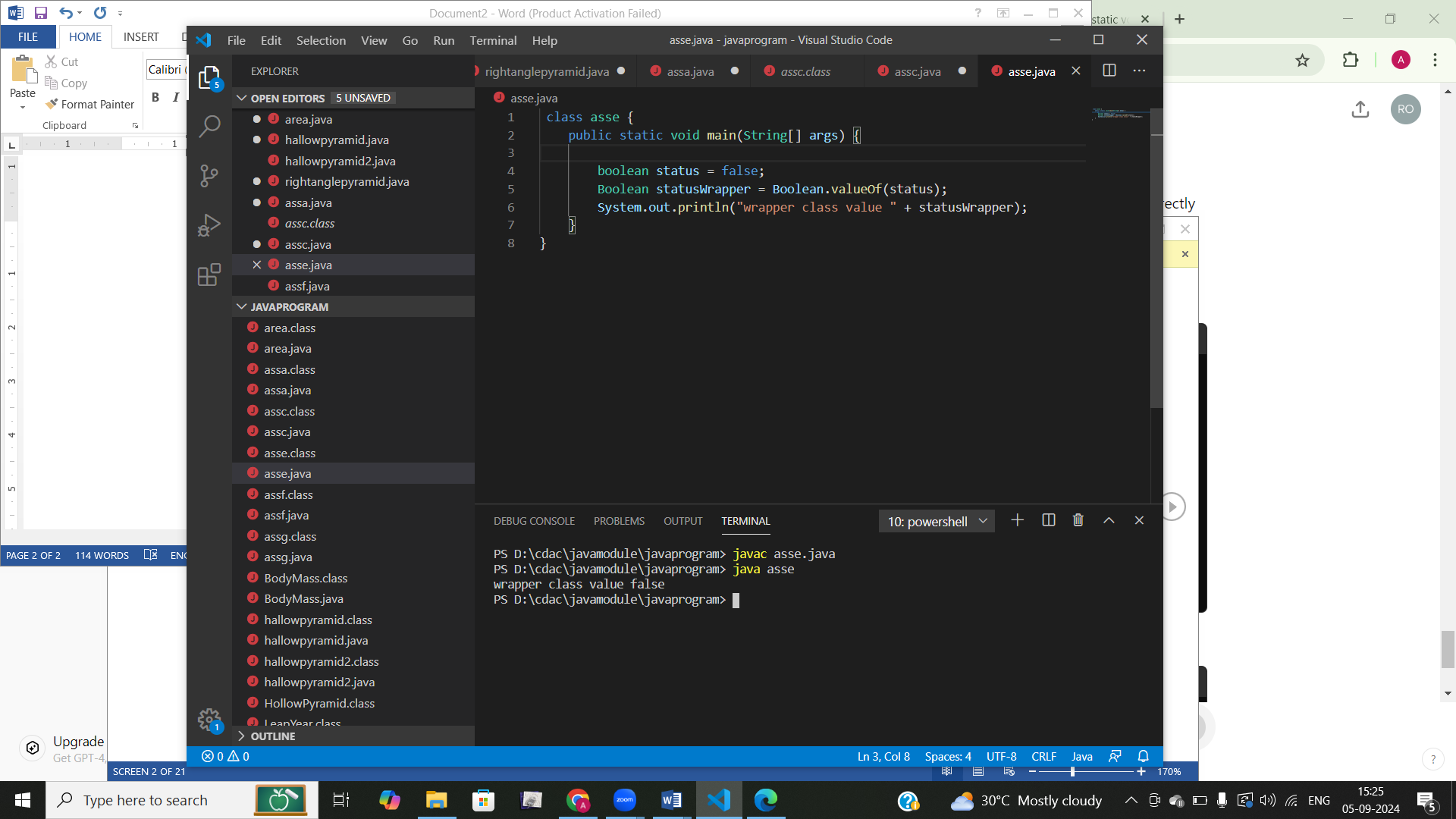
**c.** Declare a method-local variable strStatus of type String with the value "true" and convert it to a boolean using the parseBoolean method. (Hint: Use Boolean.parseBoolean(String)).



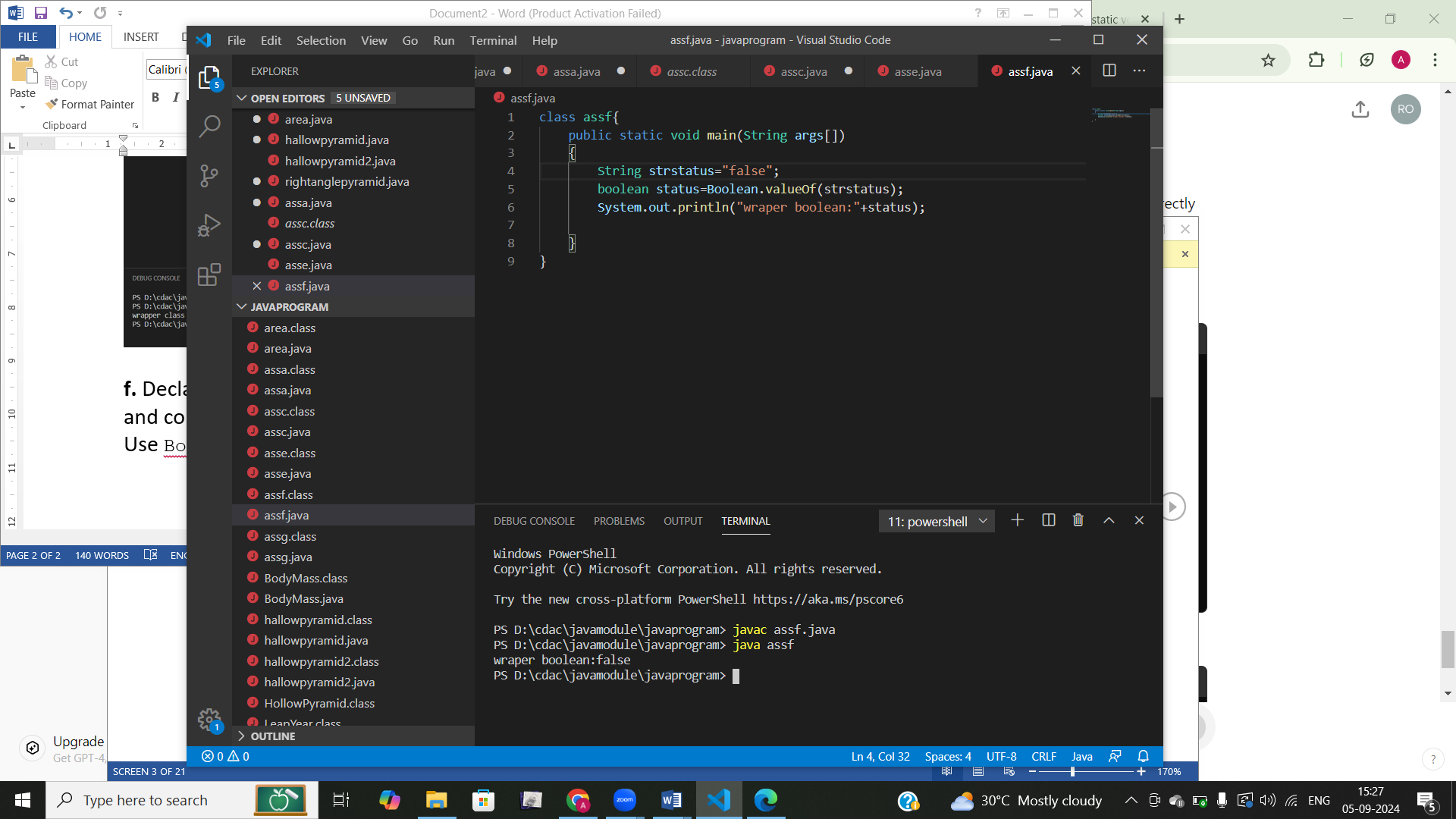
**d.** Declare a method-local variable strStatus of type String with the value "1" or "0" and attempt to convert it to a boolean. (Hint: parseBoolean method will not work as expected with "1" or "0").



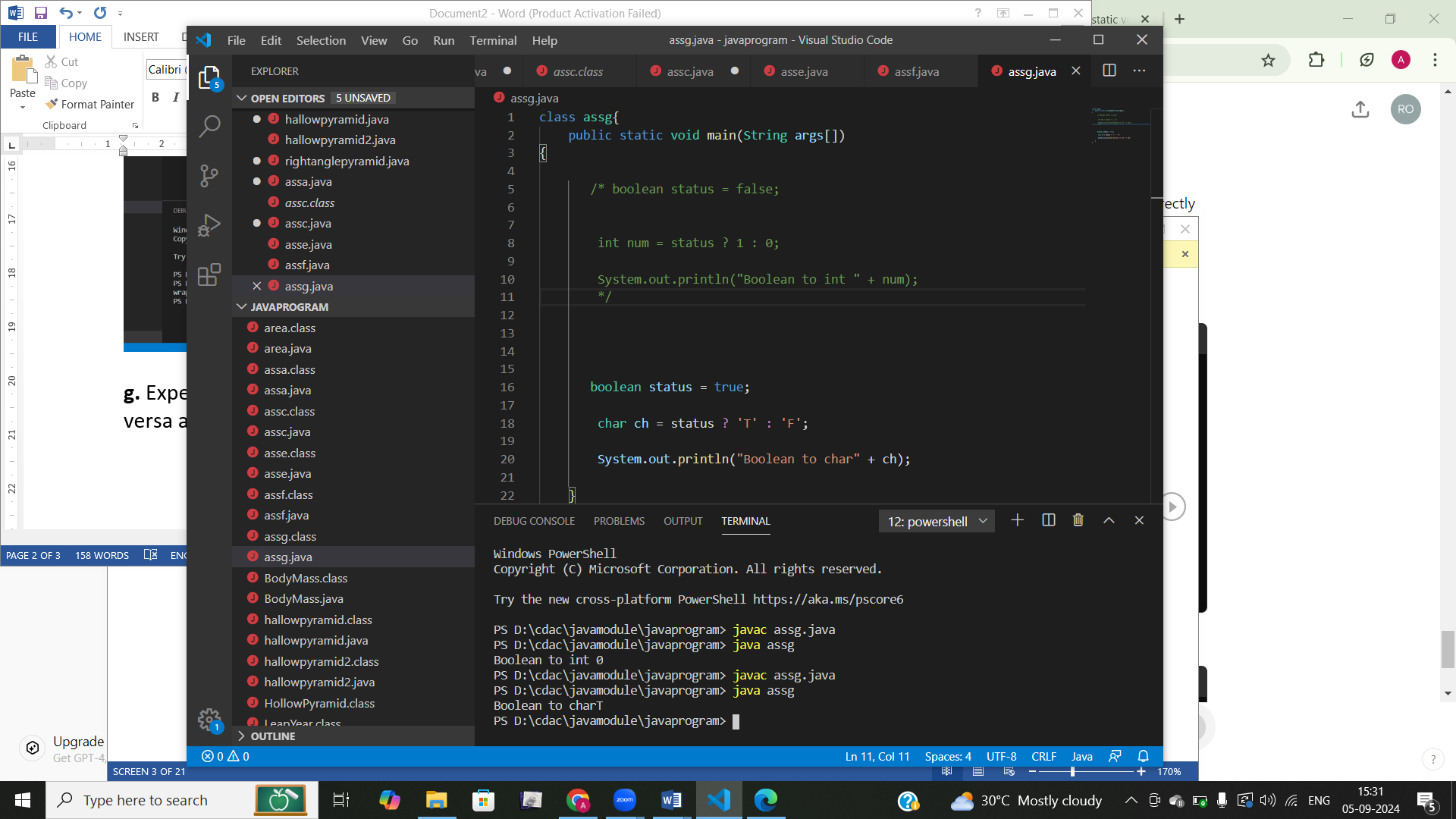
**e.** Declare a method-local variable status of type boolean with the value true and convert it to the corresponding wrapper class using Boolean.valueOf(). (Hint: Use Boolean.valueOf(boolean)).



**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)).



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



#### ****2. Working with**** java.lang.Byte

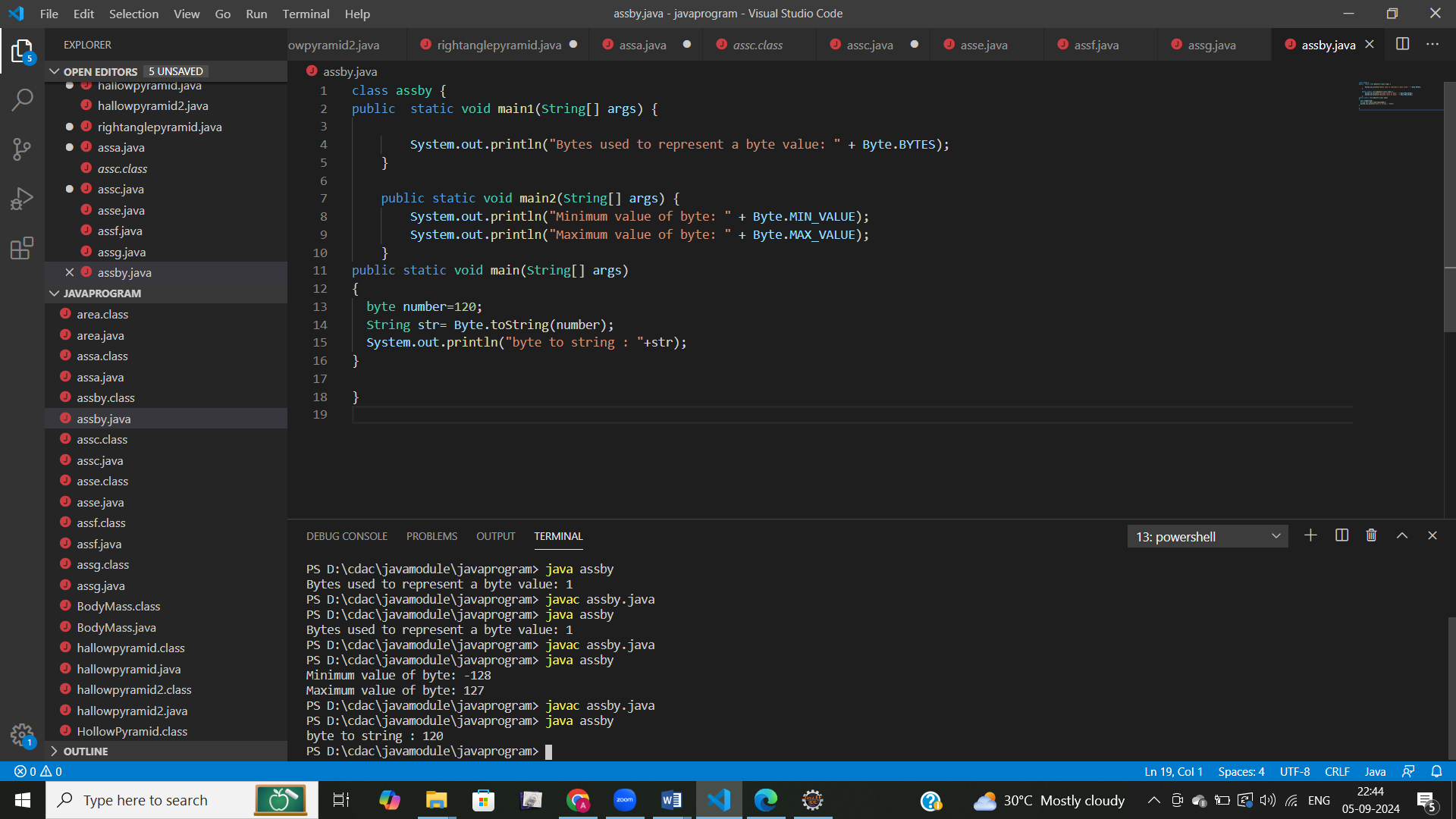
**a.** Explore the [Java API documentation for java.lang.Byte](https://docs.oracle.com/javase/8/docs/api/java/lang/Byte.html) and observe its modifiers and super types.

**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)).

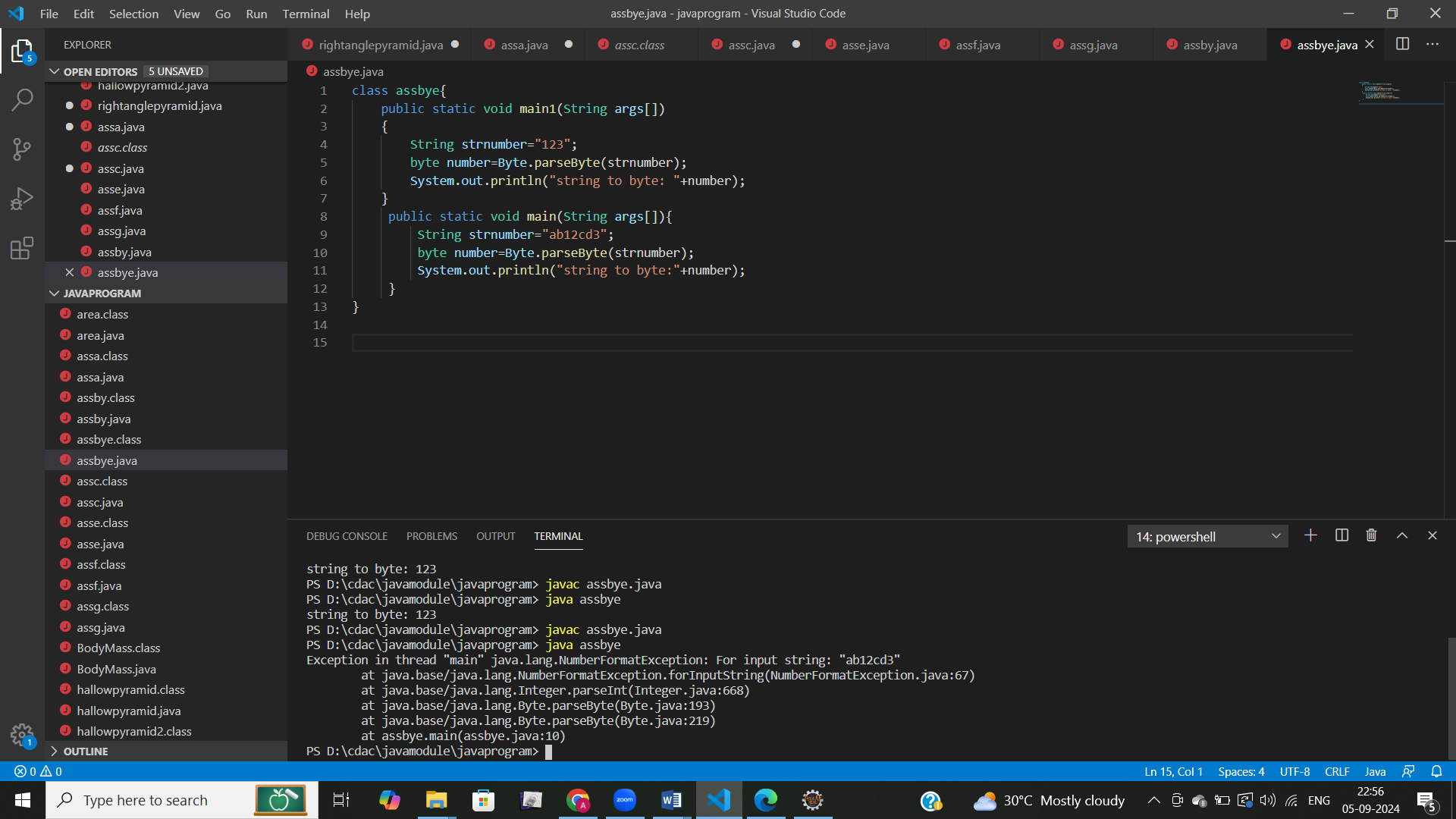
**Answer b,c,d:**



**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).

Answer e,f:



**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)).

Answer g,h:



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

class assbyh{

    public static void main1(String args[])

    {

    byte num=100;

    int a=(int)num;

    char b=(char)num;

    float c=(float)num;

    double d=(double)num;

    short e=(short)num;

    System.out.println("byte to integer: "+a);

    System.out.println("byte to char: "+b);

    System.out.println("byte to float: "+c);

    System.out.println("byte to double: "+d);

    System.out.println("byte to short: "+e);

    }

public static void main(String args[]){

int a=130;

short b=20;

char ch='z';

float d=10.0f;

double e=10.0d;

byte val=(byte)a;

System.out.println("int to byte :"+val);

byte val1=(byte)b;

System.out.println("short to byte :"+val1);

byte val2=(byte)ch;

System.out.println("char to byte :"+val2);

byte val3=(byte)d;

System.out.println("float to byte :"+val3);

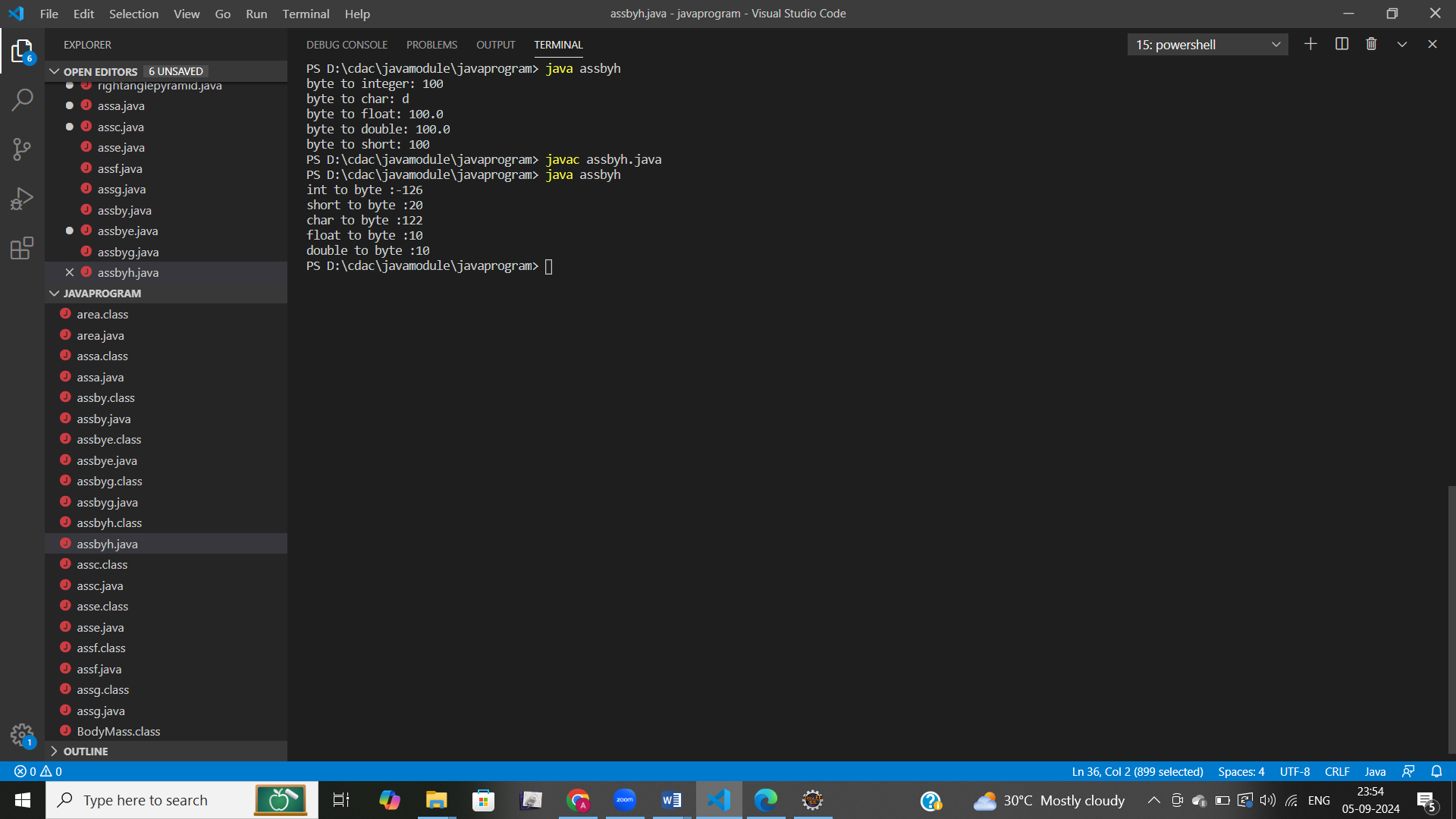
byte val4=(byte)e;

System.out.println("double to byte :"+val4);

}

}

Output:



#### ****3. Working with**** java.lang.Short

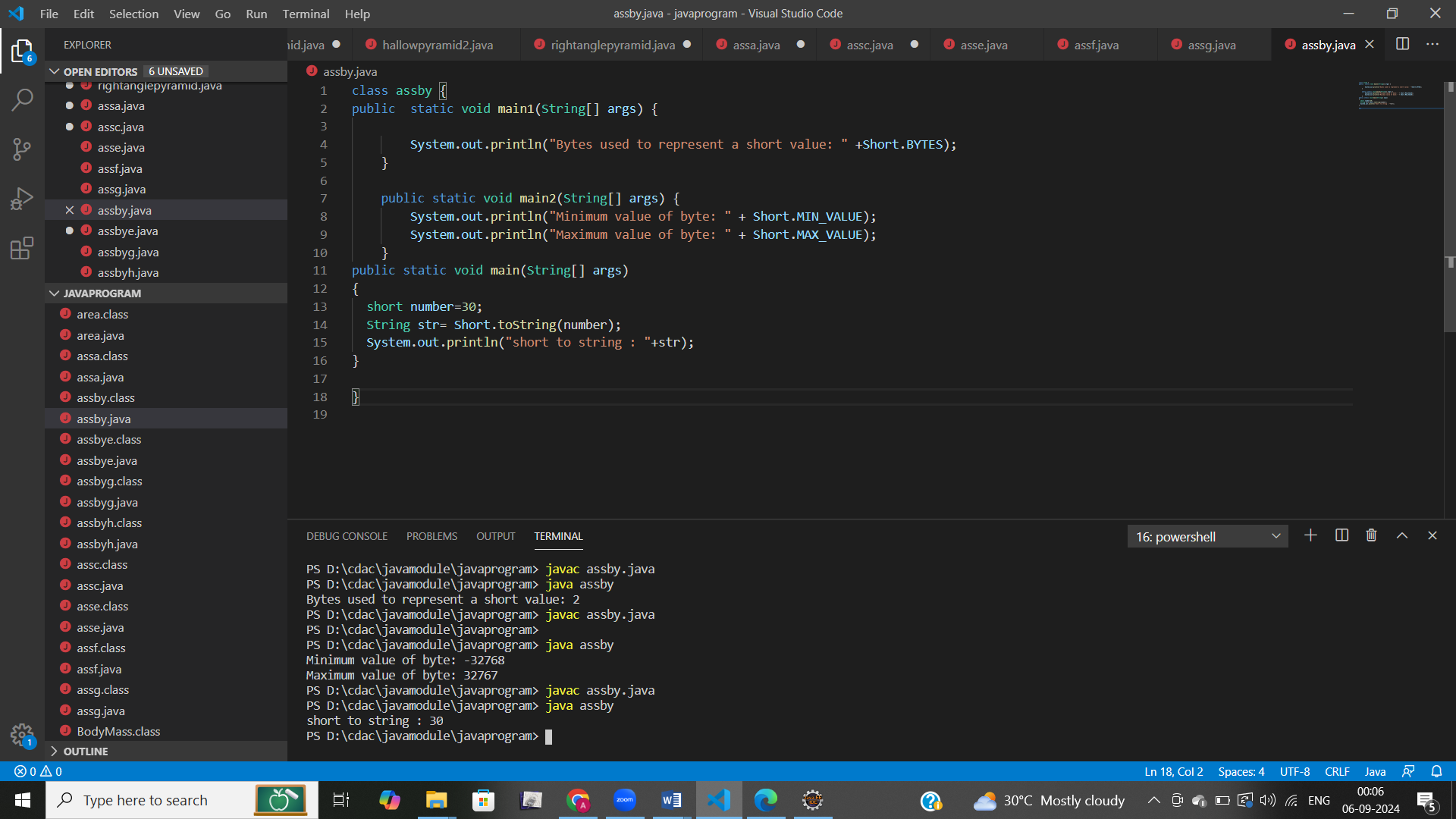
**a.** Explore the [Java API documentation for java.lang.Short](https://docs.oracle.com/javase/8/docs/api/java/lang/Short.html) and observe its modifiers and super types.

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

**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).

**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)).

Answer b,c,d:



**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)).

**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).

**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)).

**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)).

**Answer e,f,g,h:**

class assbye{

    public static void main1(String args[])

    {

        String strnumber="123";

        short number=Short.parseShort(strnumber);

        System.out.println("string to short: "+number);

    }

    public static void main2(String args[])

    {

        String strnumber="ab12c3";

        short number=Short.parseShort(strnumber);

        System.out.println("string to short: "+number);

    }

    public static void main3(String args[])

    {

        short number=123;

        Short wrappernum=Short.valueOf(number);

        System.out.println("short to wrapper class :"+wrappernum);

    }

public static void main(String args[])

    {

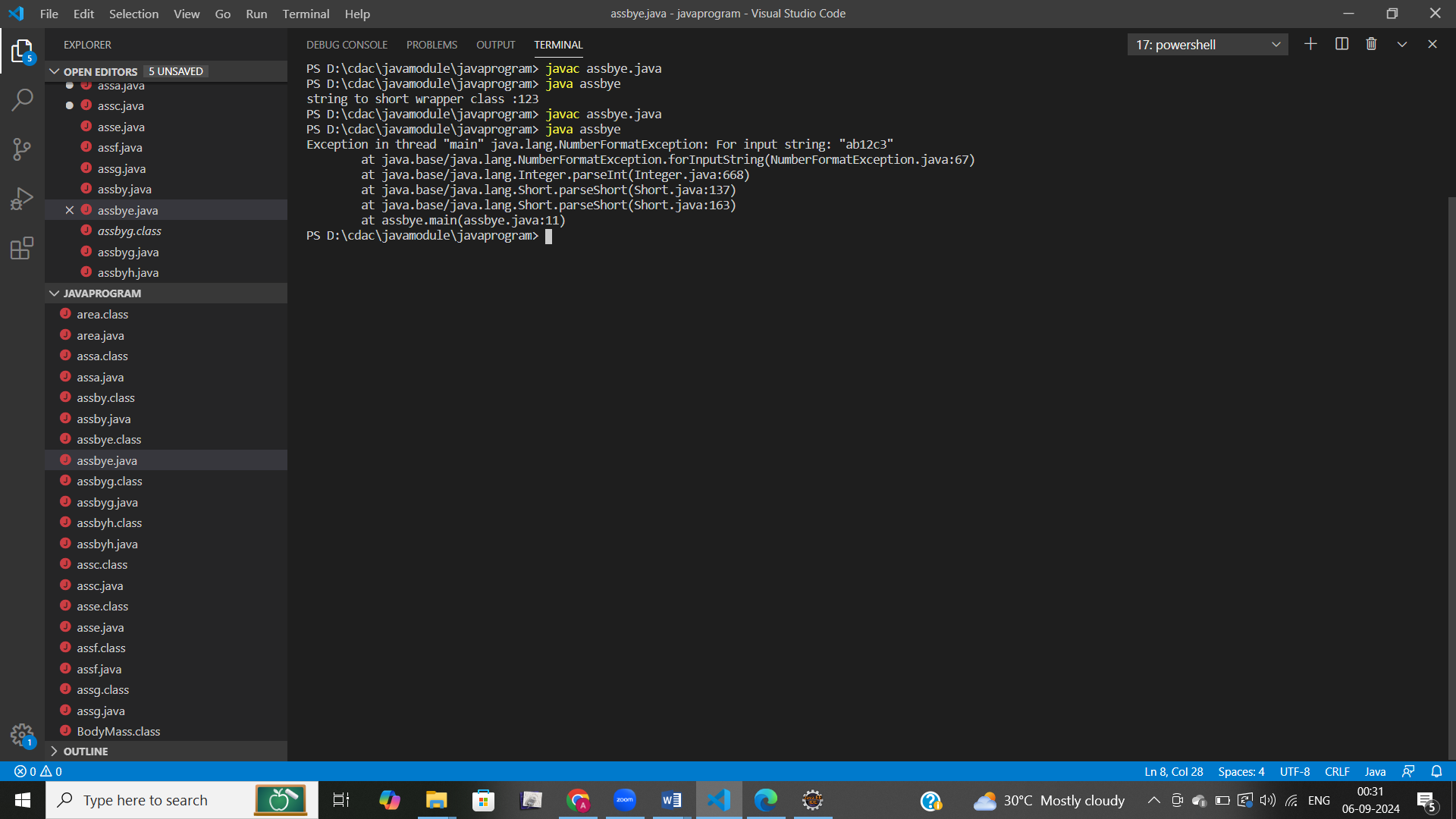
        String strnumber="123";

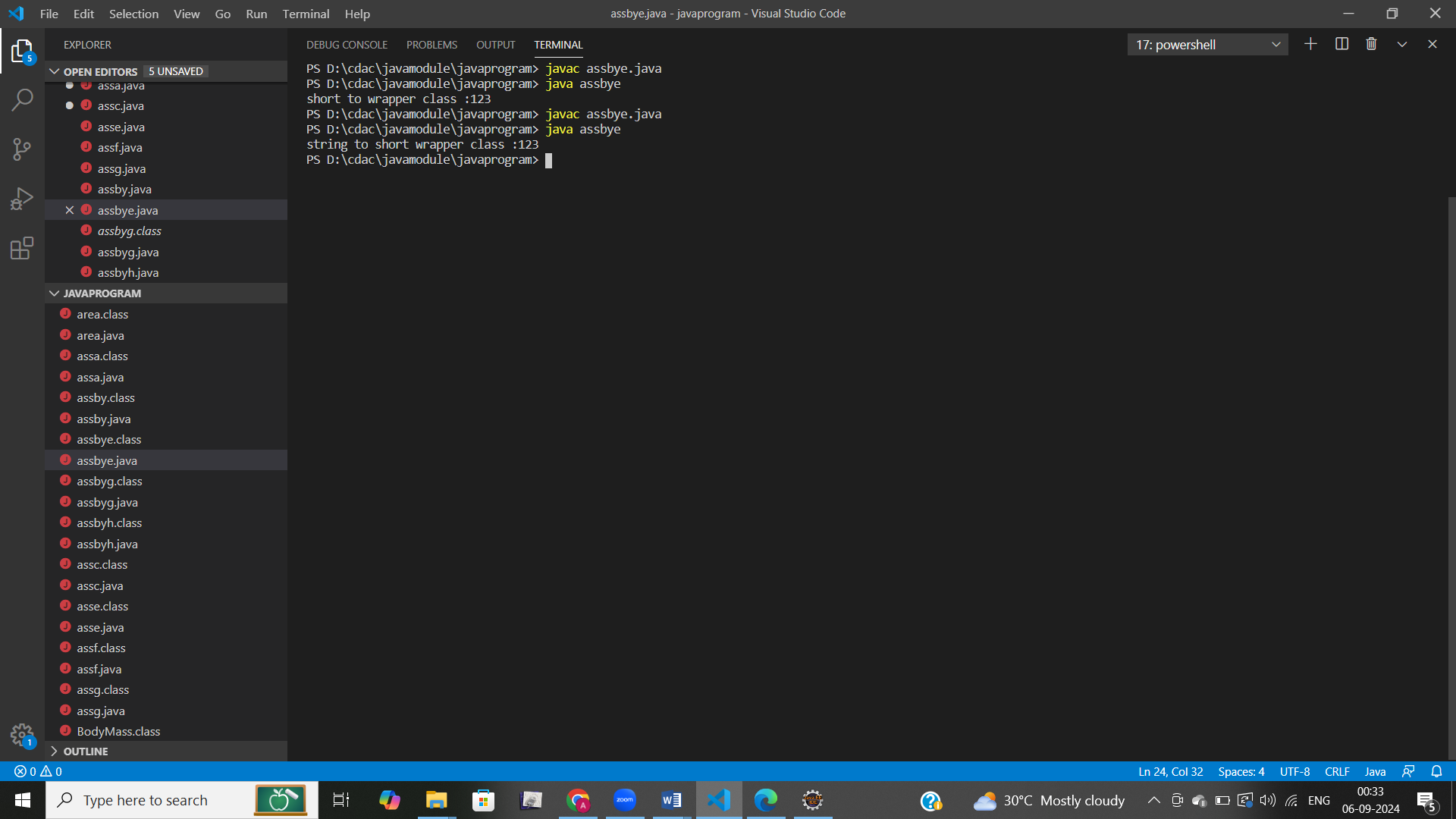
        Short wrappernum=Short.valueOf(strnumber);

        System.out.println("string to short wrapper class :"+wrappernum);

    }

}





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

class asssh{

    public static void main1(String args[])

    {

    short num=30;

    int a=(int)num;

    char b=(char)num;

    float c=(float)num;

    double d=(double)num;

    byte e=(byte)num;

    System.out.println("short to integer: "+a);

    System.out.println("short to char: "+b);

    System.out.println("short to float: "+c);

    System.out.println("short to double: "+d);

    System.out.println("short to byte: "+e);

    }

public static void main(String args[]){

int a=130;

byte b=20;

char ch='z';

float d=10.0f;

double e=10.0d;

short val=(byte)a;

System.out.println("int to short :"+val);

short val1=(byte)b;

System.out.println("byte to short :"+val1);

short val2=(byte)ch;

System.out.println("char to short :"+val2);

short val3=(byte)d;

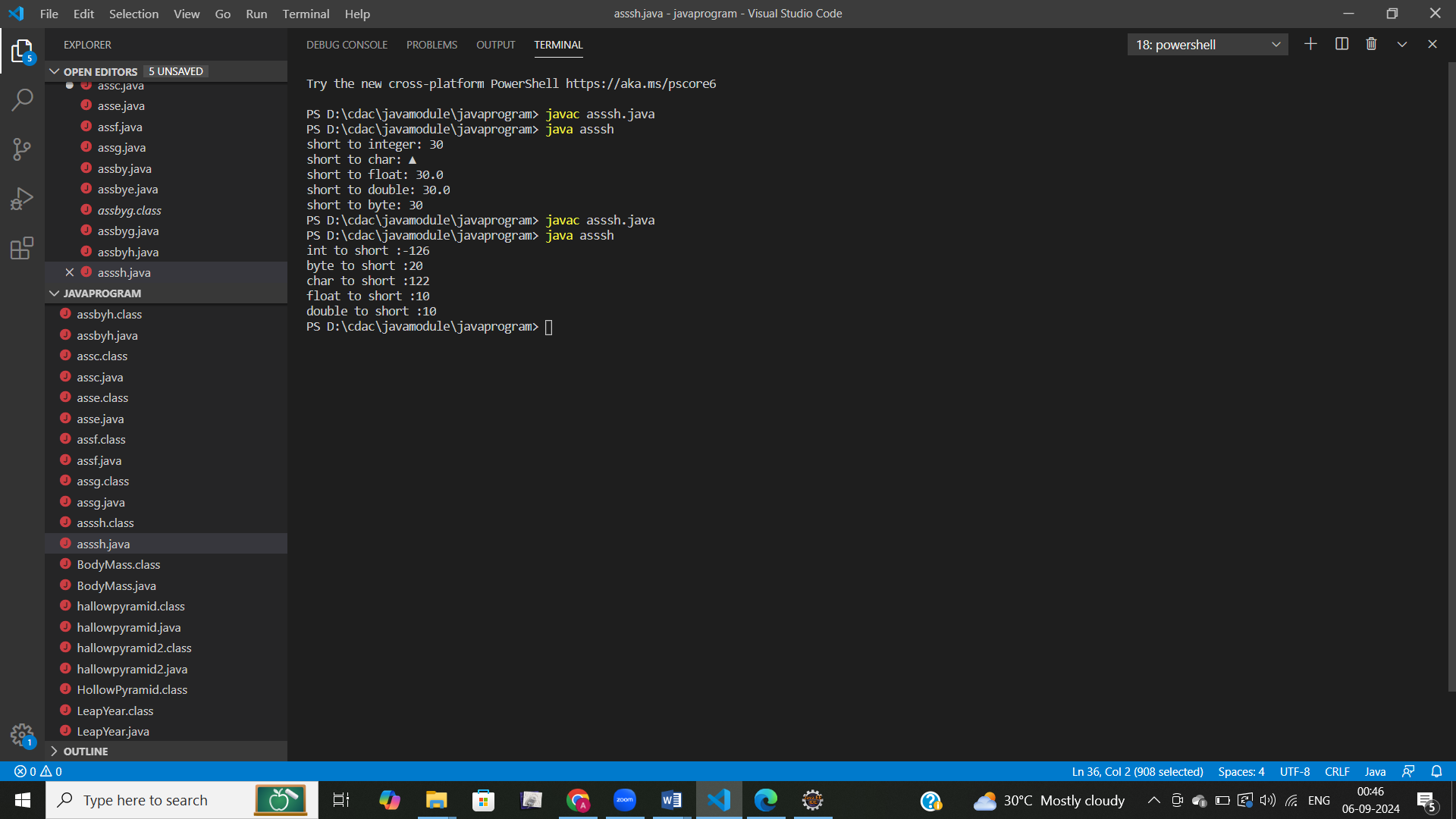
System.out.println("float to short :"+val3);

short val4=(byte)e;

System.out.println("double to short :"+val4);

}

}



#### Q4****. Working with**** java.lang.Integer

**a.** Explore the [Java API documentation for java.lang.Integer](https://docs.oracle.com/javase/8/docs/api/java/lang/Integer.html) and observe its modifiers and super types.

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

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

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

/answer:b,c,d:

class assin{

    public static void main1(String [] args)

{

    System.out.println("size of integer in byte:"+Integer.BYTES);

}

public static void main2(String [] args){

System.out.println("minimum value of integer:"+Integer.MIN\_VALUE);

System.out.println("maximum value of integer:"+Integer.MAX\_VALUE);

}

public static void main(String [] args){

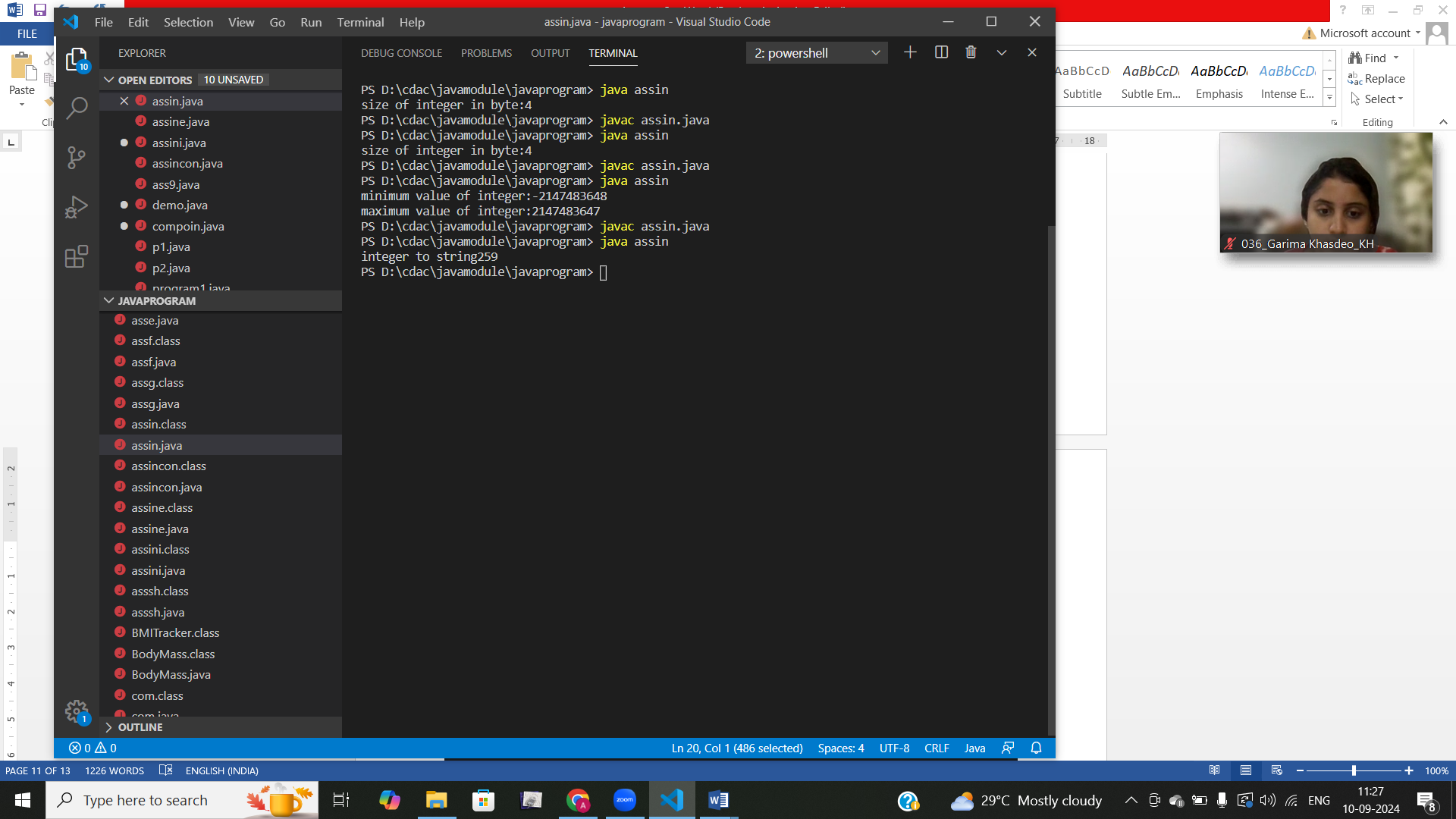
int number=259;

String str=Integer.toString(number);

System.out.println("integer to string"+str);

}

}



**e.** Declare a method-local variable strNumber of type String with some value and convert it to an int value using the parseInt method. (Hint: Use Integer.parseInt(String)).

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

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

class assine{

    public static void main1(String [] args)

{

    String str="123";

    int number=Integer.parseInt(str);

    System.out.println("string to  integer :"+number);

}

public static void main2(String [] args)

{

String str="ab12c3";

    int number=Integer.parseInt(str);

    System.out.println("string to  integer :"+number);

}

public static void main(String [] args)

{

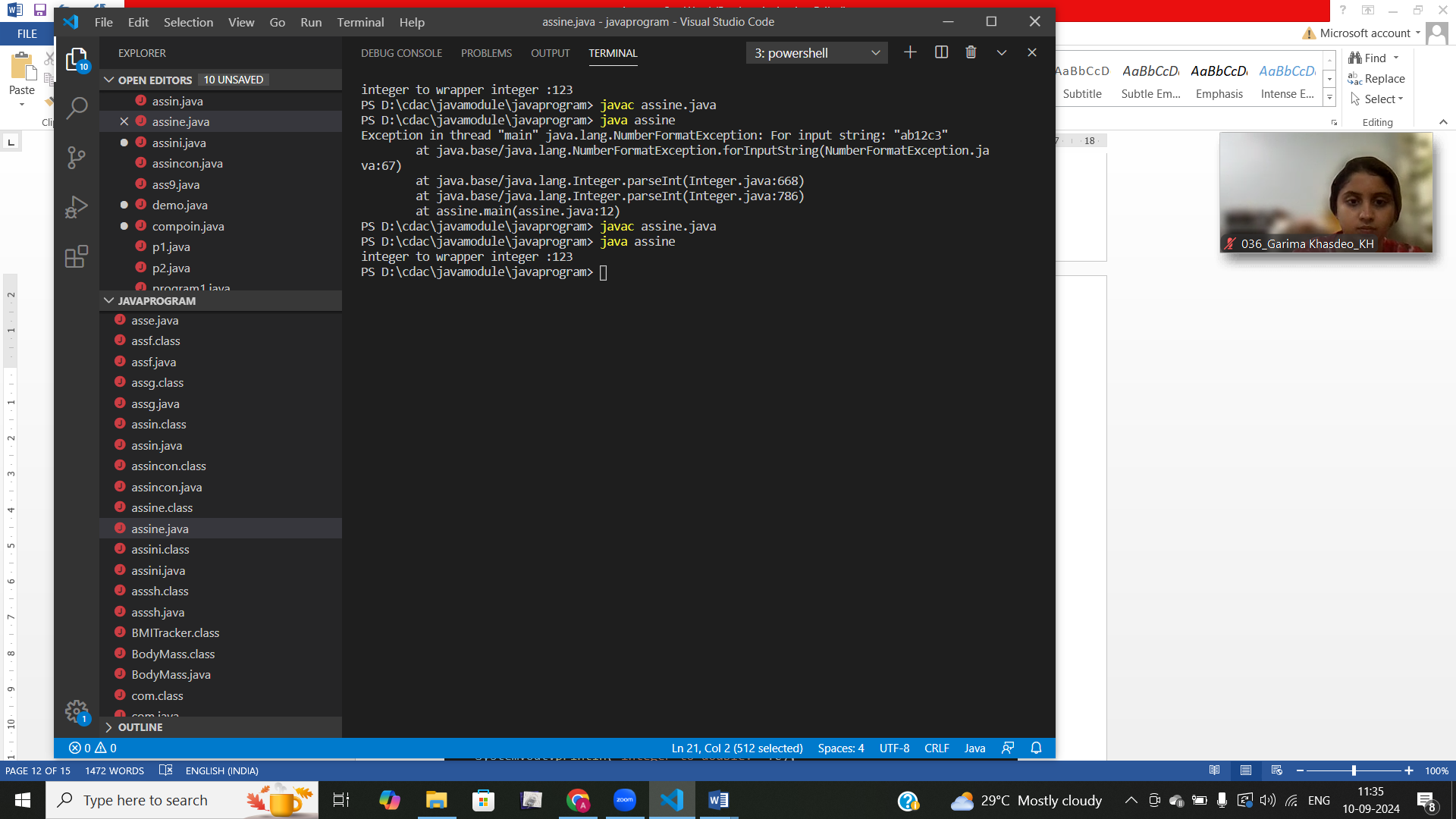
int number=123;

Integer num=Integer.valueOf(number);

System.out.println("integer to wrapper integer :"+num);

}

}



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

class assincon {

    public static void main1(String args [])

    {

        int a=97;

        short b=(short)a;

        double c=(double)a;

        float d=(float)a;

        char ch=(char)a;

        byte e=(byte)a;

    System.out.println("integre to short: "+b);

    System.out.println("integer to double: "+c);

    System.out.println("integer to float: "+d);

    System.out.println("integer  to char: "+ch);

    System.out.println("integer to byte: "+e);

    }

public static void main(String args[]){

short a=123;

byte b=20;

char ch='z';

float d=10.0f;

double e=10.0d;

int val=(int)a;

System.out.println("short to int :"+val);

int val1=(int)b;

System.out.println("byte to int :"+val1);

int val2=(int)ch;

System.out.println("char to int:"+val2);

int val3=(int)d;

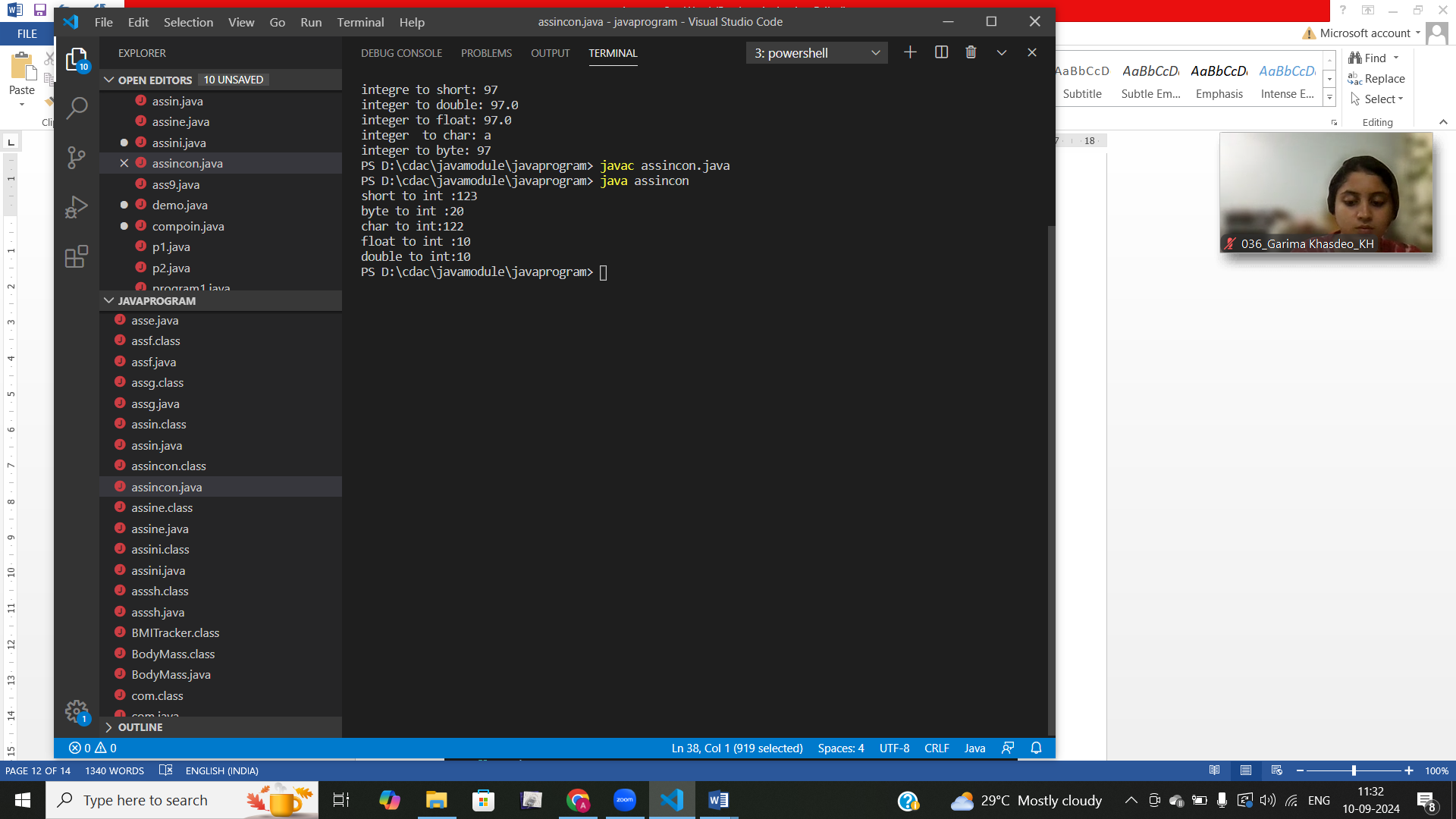
System.out.println("float to int :"+val3);

int val4=(int)e;

System.out.println("double to int:"+val4);

}

}



**i.** Declare two integer variables with values 10 and 20, and add them using a method from the Integer class. (Hint: Use Integer.sum(int, int)).

**j.** Declare two integer variables with values 10 and 20, and find the minimum and maximum values using the Integer class. (Hint: Use Integer.min(int, int) and Integer.max(int, int)).

**k.** Declare an integer variable with the value 7. Convert it to binary, octal, and hexadecimal strings using methods from the Integer class. (Hint: Use Integer.toBinaryString(int), Integer.toOctalString(int), and Integer.toHexString(int)).

class assini{

    public static void main1(String args[]){

        int a=10;

        int b=20;

        System.out.println("addition is :"+Integer.sum(a,b));

        System.out.println("minimum number is :"+Integer.min(a,b));

        System.out.println("maximum number is :"+Integer.max(a,b));

    }

    public static void main(String args[]){

        int a=7;

        String b=Integer.toBinaryString(a);

        System.out.println("integer to binary :"+b);

        String c=Integer.toOctalString(a);

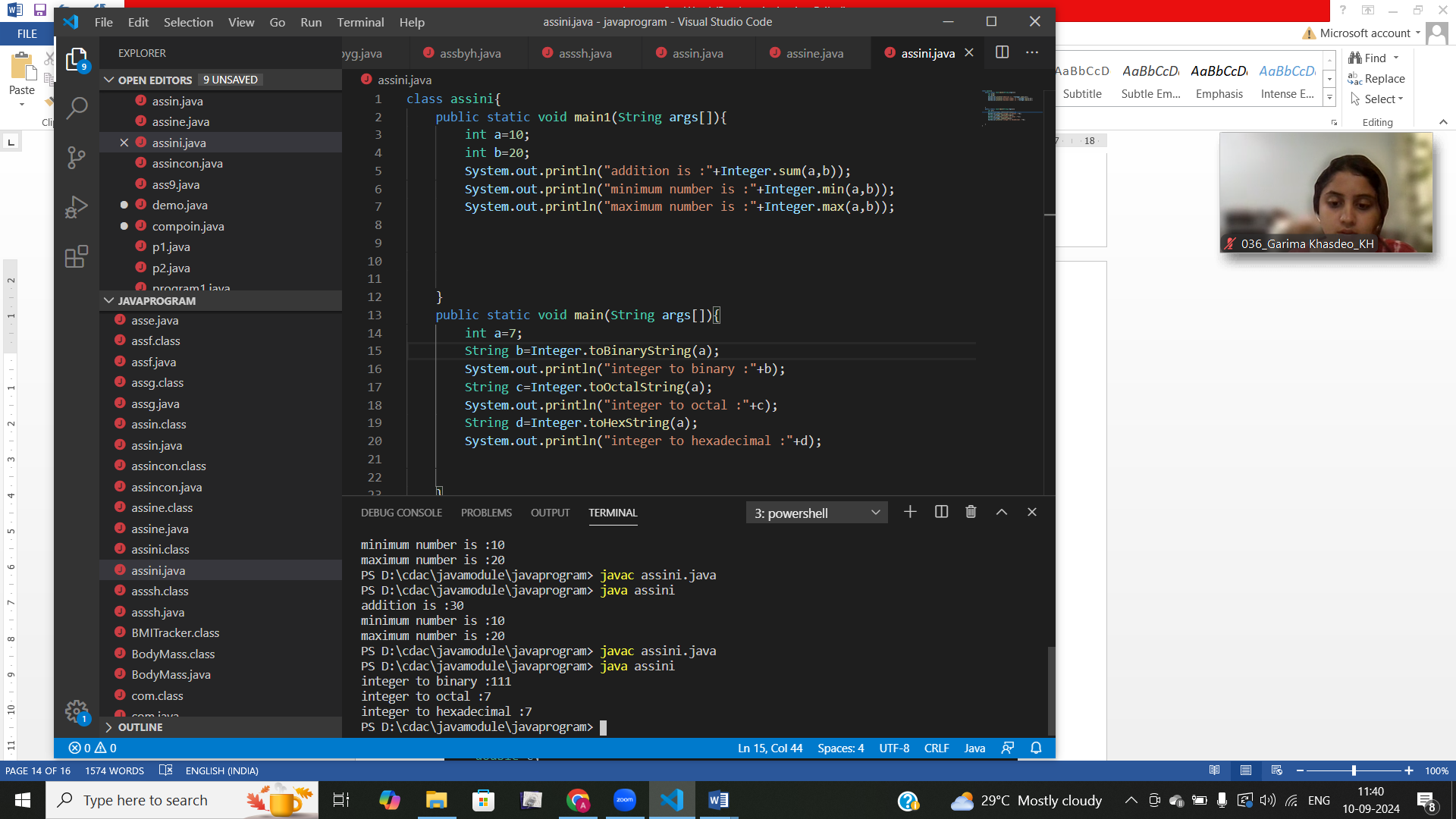
        System.out.println("integer to octal :"+c);

        String d=Integer.toHexString(a);

        System.out.println("integer to hexadecimal :"+d);

    }

}



Q8)

public class assq8{

    public static void main(String[] args) {

        int intValue = 100;

        double doubleValue = 45.67;

        float floatValue = 23.45f;

        long longValue = 123456789L;

        short shortValue = 123;

        byte byteValue = 10;

        char charValue = 'A';

        boolean booleanValue = true;

        System.out.println("Conversion using Wrapper Class toString() methods:");

        String intStr1 = Integer.toString(intValue);

        String doubleStr1 = Double.toString(doubleValue);

        String floatStr1 = Float.toString(floatValue);

        String longStr1 = Long.toString(longValue);

        String shortStr1 = Short.toString(shortValue);

        String byteStr1 = Byte.toString(byteValue);

        String charStr1 = Character.toString(charValue);

        String booleanStr1 = Boolean.toString(booleanValue);

        System.out.println("Integer: " + intStr1);

        System.out.println("Double: " + doubleStr1);

        System.out.println("Float: " + floatStr1);

        System.out.println("Long: " + longStr1);

        System.out.println("Short: " + shortStr1);

        System.out.println("Byte: " + byteStr1);

        System.out.println("Char: " + charStr1);

        System.out.println("Boolean: " + booleanStr1);

        System.out.println("\nConversion using String.valueOf() method:");

        String intStr2 = String.valueOf(intValue);

        String doubleStr2 = String.valueOf(doubleValue);

        String floatStr2 = String.valueOf(floatValue);

        String longStr2 = String.valueOf(longValue);

        String shortStr2 = String.valueOf(shortValue);

        String byteStr2 = String.valueOf(byteValue);

        String charStr2 = String.valueOf(charValue);

        String booleanStr2 = String.valueOf(booleanValue);

        System.out.println("Integer: " + intStr2);

        System.out.println("Double: " + doubleStr2);

        System.out.println("Float: " + floatStr2);

        System.out.println("Long: " + longStr2);

        System.out.println("Short: " + shortStr2);

        System.out.println("Byte: " + byteStr2);

        System.out.println("Char: " + charStr2);

        System.out.println("Boolean: " + booleanStr2);

    }

}

#### Q9) ****. Default Values of Primitive Types****

Declare variables of each primitive type as fields of a class and check their default values. (Note: Default values depend on whether the variables are instance variables or static variables).

class ass{

    int m;

    char b;

    double c;

    boolean d;

    short e;

    long f;

    byte g;

    float h;

}

class ass9{

 public static void main(String args[]){

     ass a=new ass();

     System.out.println("default size of float:"+a.m);

     System.out.println("default size of char:"+a.b + "'");

     System.out.println("default size of double:"+a.c);

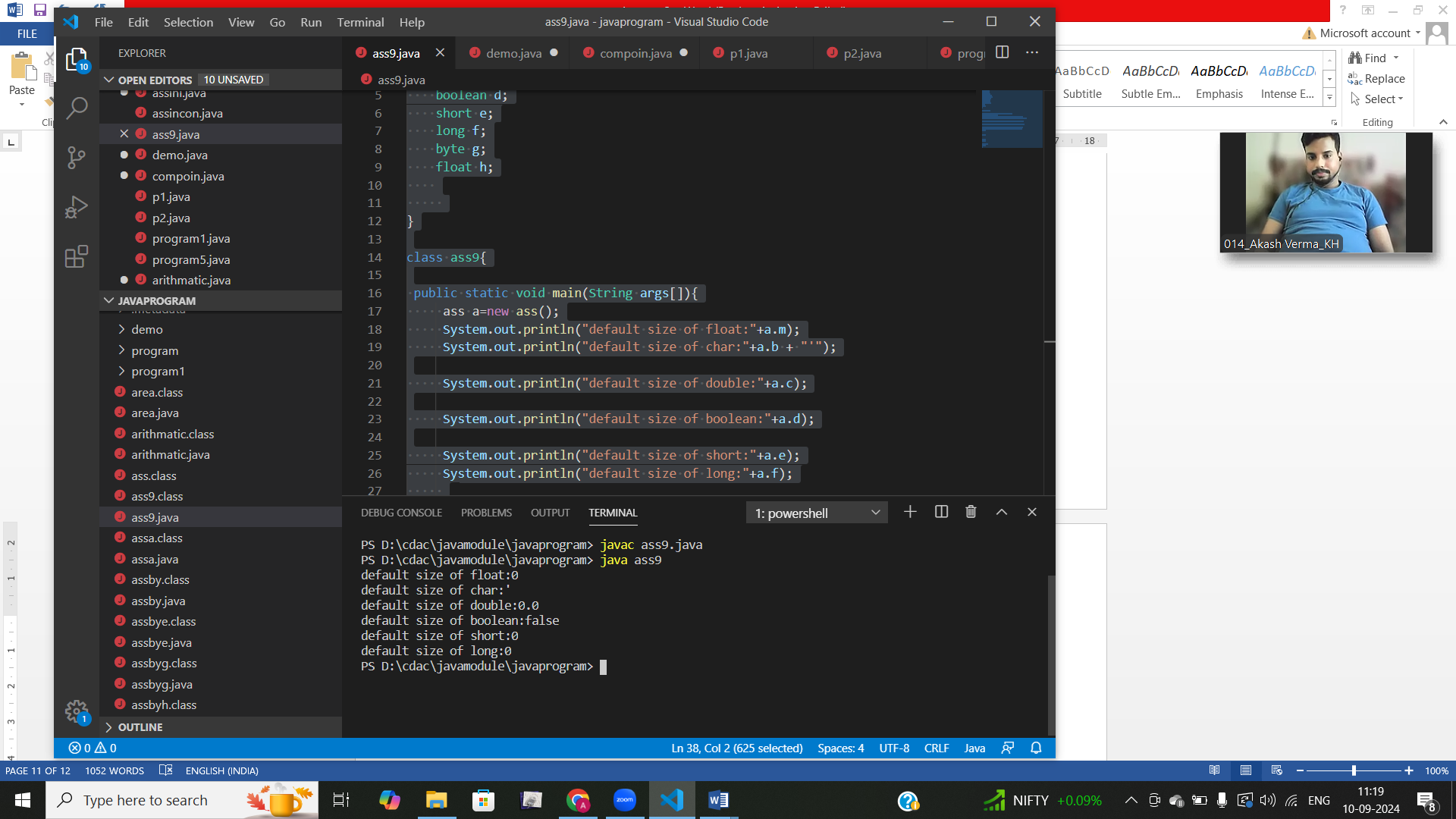
     System.out.println("default size of boolean:"+a.d);

     System.out.println("default size of short:"+a.e);

     System.out.println("default size of long:"+a.f);

    }

}



#### Q10) ****Arithmetic Operations with Command Line Input****

Write a program that accepts two integers and an arithmetic operator (+, -, \*, /) from the command line. Perform the specified arithmetic operation based on the operator provided. (Hint: Use switch-case for operations

public class arithmatic {

    public static void main(String[] args) {

        int num1 = Integer.parseInt(args[0]);

        char operator = args[1].charAt(0);

        int num2 = Integer.parseInt(args[2]);

        int result = 0;

        switch (operator) {

            case '+':

                result = num1 + num2;

                break;

            case '-':

                result = num1 - num2;

                break;

            case '\*':

                result = num1 \* num2;

                break;

            case '/':

                if (num2 == 0) {

                    System.out.println(" Division by zero is not allowed.");

                    return;

                }

                result = num1 / num2;

                break;

            default:

                System.out.println("Error: Invalid operator. Please use +, -, \*, or /");

                return;

        }

        System.out.println("Result: " + result);

    }

}