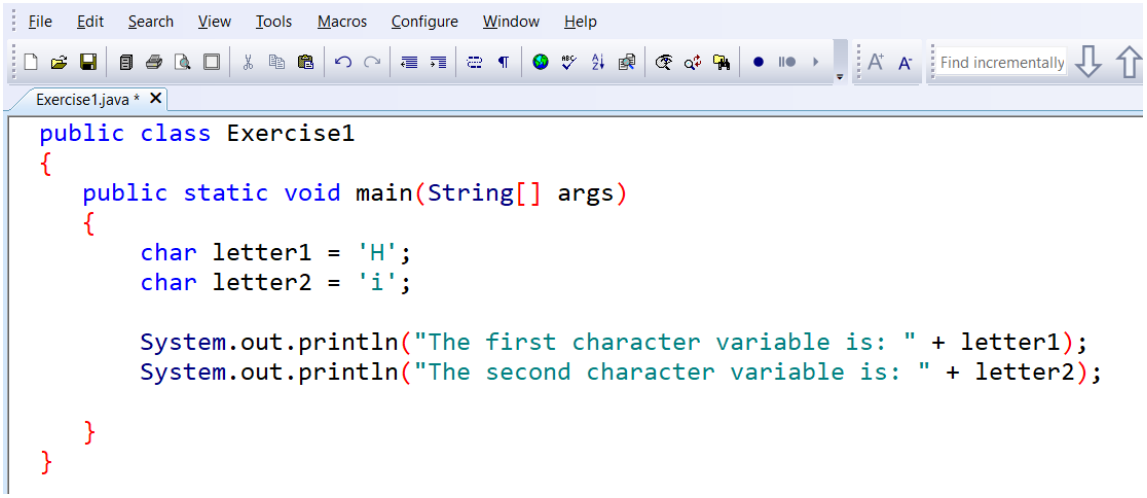
Lab 1 B – Variables

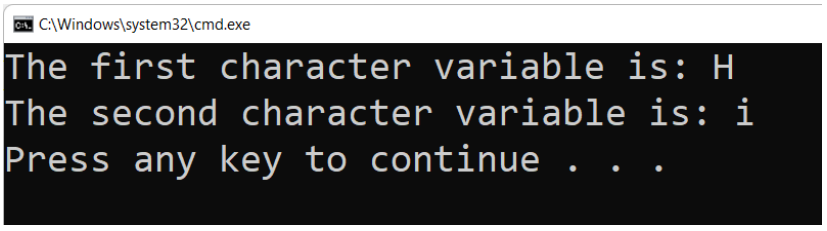
Java Variables Exercise 1

Goal: Create a simple program in Java using TextPad that contains two character variables. Ensure it compiles and runs successfully.

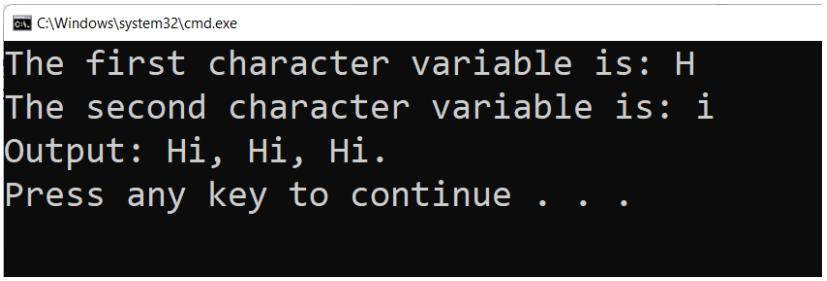
1. Launch TextPad and enter the code shown below, ensuring that the java file is named correctly:



Your output should be similar to as shown below:



Amend your output so that it produces the following output (using the two char variables):



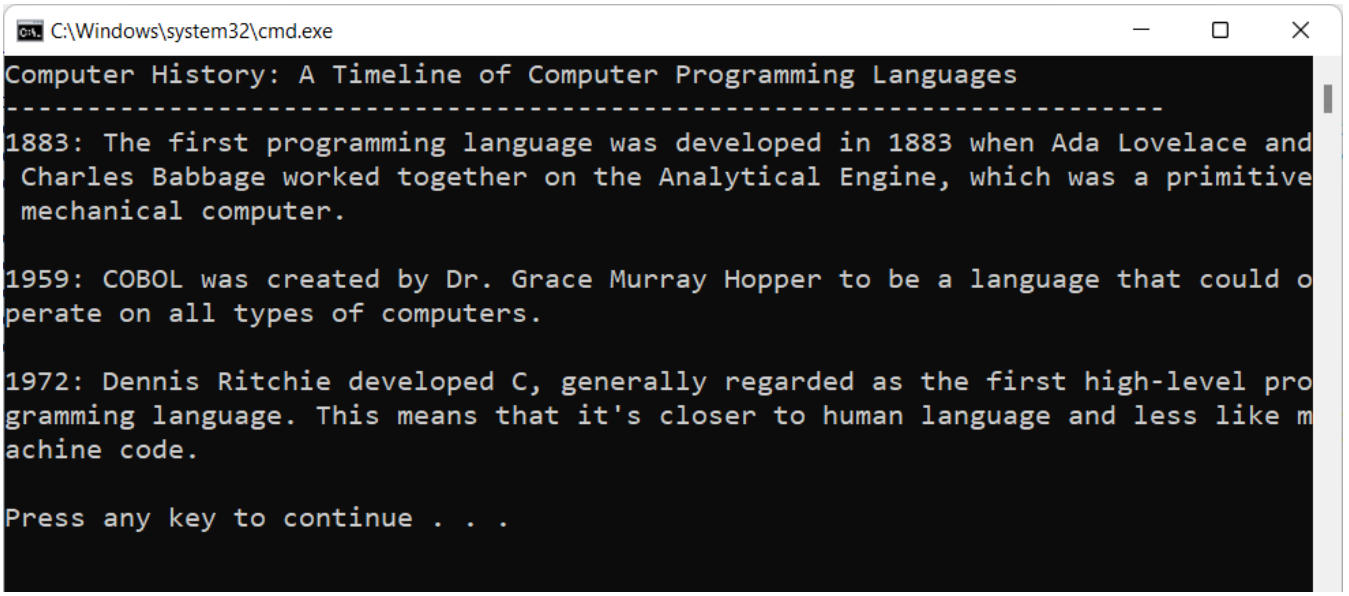
Exercise 2

Goal: Create a simple program in Java that outputs a series of strings. Ensure it compiles and runs successfully.

Create a new file called *StringOutputs* for this exercise. Following the guide below, create a program that declares the following String variables:

```
String title = "Computer History: A Timeline of Computer Programming Languages";
String part1 = "1883: The first programming language was developed in 1883 when Ada Lovelace and Charles Babbage worked together on the Analytical Engine, which was a primitive mechanical computer.";
String part2 = "1959: COBOL was created by Dr. Grace Murray Hopper to be a language that could operate on all types of computers.";
String part3 = "1972: Dennis Ritchie developed C, generally regarded as the first high-level programming language. This means that it's closer to human language and less like machine code.";
```

Write a program that produces output using the above variables similar to as shown:



Amend your code so that the years (1883, 1959 and 1972) are stored as ints (remove them from the String variables). Run your program to ensure that the end resulting output is still the same as above.

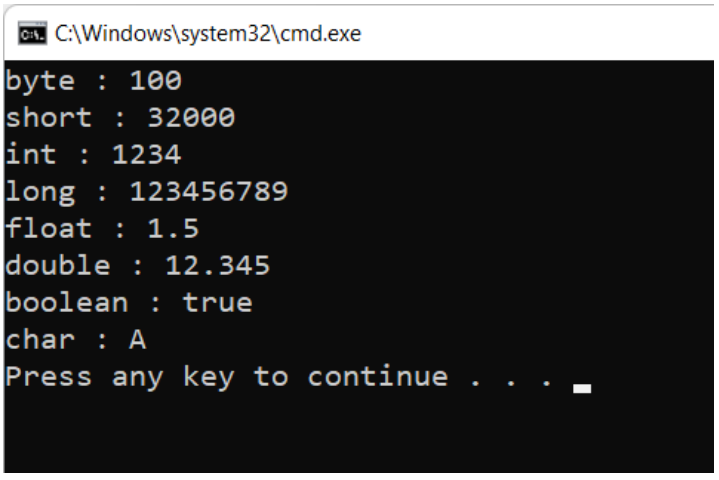
Exercise 3

Goal: Create a program with detailed comments in Java outputs examples of all the primary data type variables. Ensure it compiles and runs successfully.

Your program should include the following declarations and comments:

```java
// byte
// Byte data type is an 8-bit integer
// Minimum value is -128
// Maximum value is 127
// Default value is 0
byte b = 100;

// short
// Short data type is a 16-bit integer
// Minimum value is -32,768
// Maximum value is 32,767
// Default value is 0
short s = 32000;

// int
// Int data type is a 32-bit integer
// Minimum value is -2,147,483,648
// Maximum value is 2,147,483,647
// Default value is 0
int i = 1234;

// long
// Long data type is a 64-bit integer
// Maximum value is 9,223,372,036,854,775,807
// Default value is 0L
long l = 123456789L;

// float
// Float data type is a single-precision 32-bit IEEE 754 floating point
// Float is mainly used to save memory in large arrays of floating point numbers
// Default value is 0.0f
float f = 1.5f;

// double
// double data type is a double-precision 64-bit IEEE 754 floating point
// Double is generally used as the default data type for decimal values, generally the default choice
// Default value is 0.0d
double d = 12.345;

// boolean
// boolean data type represents one bit of information
// There are only two possible values: true and false
// Default value is false
boolean bool = true;

// char
// char data type is a single 16-bit Unicode character
char c = 'A';
```

Ensure your program compiles correctly, and produces output similar to as shown below:



Exercise 4

Goal: Create a simple program in Java that performs a series of calculations. Ensure it compiles and runs successfully.

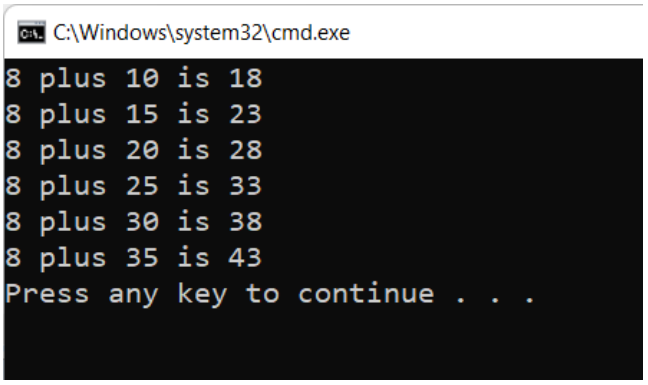
Create a new file called *PlusTen* for this exercise. Following the guide below, create a program with the following specifications:

Declare an int called number;

Output the following by adding 10, 15, 20 etc to the variable in each line.

Your output should be similar to as shown below:

-



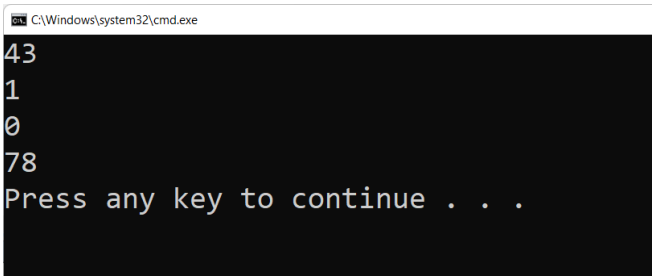
Exercise 5

Goal: Create a simple program in Java that performs a series of calculations. Ensure it compiles and runs successfully.

Create a new file called *Calculations* for this exercise. Following the guide below, create a program that performs the following calculations:

-5 + (8 * 6)
(55+9) % 9
(20 – 3) * (5 / 8)
(5 + 15) / 3 * ((2 + 8) + 3)

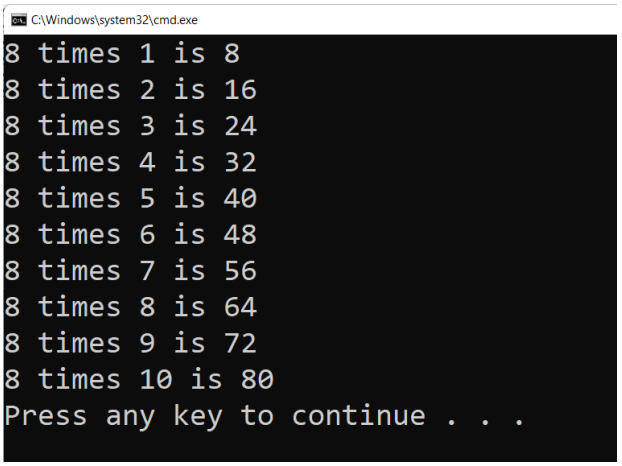
You output should be similar to as shown below:



Exercise 6

Goal: Create a program in Java that generates a multiplication table for a variable. Ensure it compiles and runs successfully.

Create a new file called *Calculations* for this exercise. Declare an integer called number. Following the guide below, create a program that outputs the following calculation of the number variable:



Exercise 7

Goal: Create a simple program in Java that calculates tax to be paid on a set wage. Ensure it compiles and runs successfully.

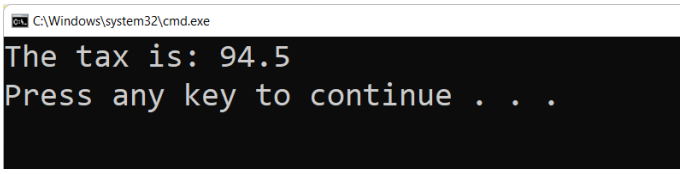
Create a new file called *CalcTax* for this exercise. Following the guide below, create a program that has the following specification:

- Declare a constant variable named *TAX_RATE* of the data type ***double***
- Declare a variable named *wage* of the data type ***double.***
- Declare a variable named *taxPaid* of the data type ***double***

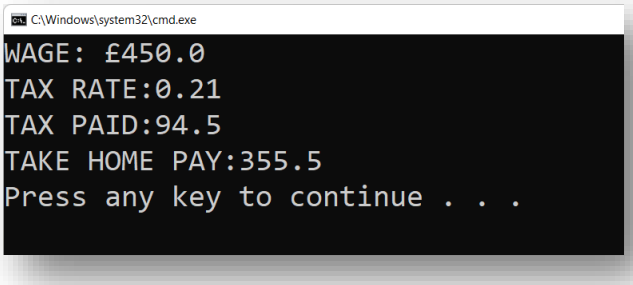
Your code can be similar to as shown below:

```
public class CalcTax
{
    public static void main(String[] args)
    {
        double wage;
        double taxPaid;
        final double TAX_RATE = 0.21;

        wage = 450;

        //Calculate the tax paid by multiplying the wage by the tax rate
        taxPaid = wage * TAX_RATE;

        System.out.println("The tax is: " + taxPaid);
    }
}
```

- Your output should be as shown below:



- Test this with multiple values to confirm that it works correctly.
- Amend your code so that the output appears as below:



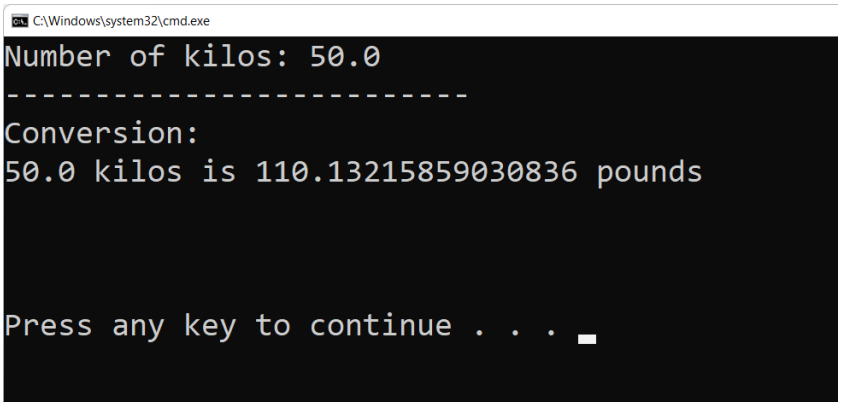
Exercise 8

Goal: Create a simple program in Java that converts kilograms to pounds. Ensure it compiles and runs successfully.

Create a new file called *KilosToPounds* for this exercise. Following the guide below, create a program that has the following specification:

- Declare a constant variable named *KILOGRAMS_PER_POUND* of the data type ***double***

  (Note that 0.454 kilograms is 1 pound)
- Declare a variable named *kilos* of the data type ***double.***
- Declare a variable named *pounds* of the data type ***double***
- Your output should be as shown below:



- Test this with multiple values to confirm that it works correctly.

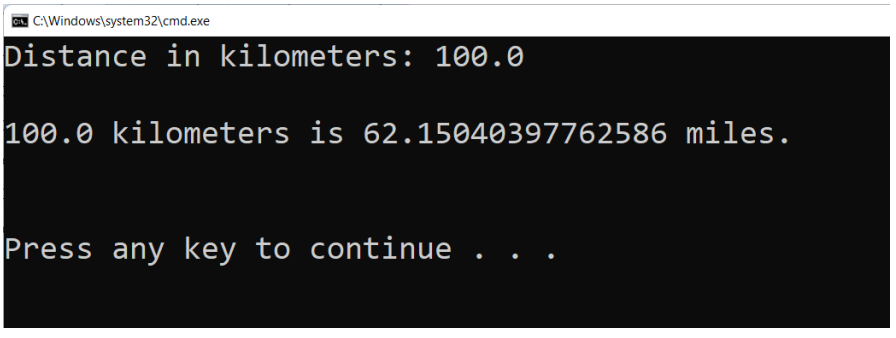
Exercise 9

Goal: Create a simple program in Java that converts kilometers to miles. Ensure it compiles and runs successfully.

Create a new file called *KmToMiles* for this exercise. Following the guide below, create a program that has the following specification:

- Declare a constant variable named *KILOMETERS_PER_MILE* of the data type ***double***

  (Note that there are 1.609 kilometers per mile)
- Declare a variable named *kilometers* of the data type ***double.***
- Declare a variable named *miles* of the data type ***double***
- Your output should be as shown below:



- Test this with multiple values to confirm that it works correctly.

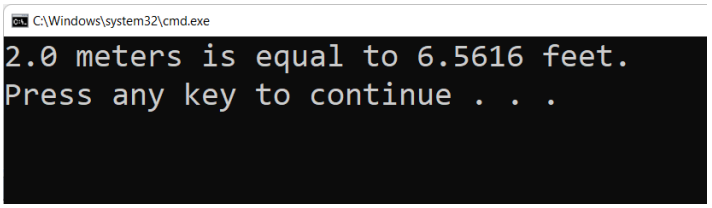
Exercise 10

Goal: Create a simple program in Java that converts kilograms to pounds. Ensure it compiles and runs successfully.

Create a new file called *MetersToFeet* for this exercise. Following the guide below, create a program that has the following specification:

- Declare a constant variable named *METERS_TO_FEET* of the data type ***double***

  (Note that there are 3.2808 feet in a meter)
- Declare a variable named *meters* of the data type ***double.***
- Declare a variable named *feet* of the data type ***double***
- Your output should be as shown below:



- Test this with multiple values to confirm that it works correctly.

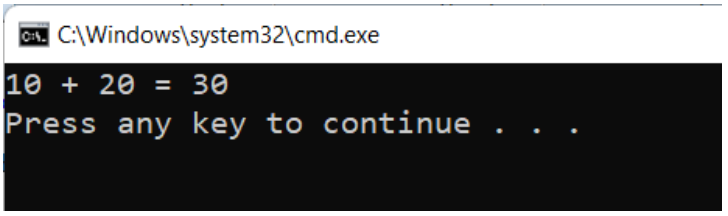
Exercise 11

Goal: Create a program in Java that adds two byte variables. Ensure it compiles and runs successfully.

Create a new file called *AddTwo* for this exercise. Following the guide below, create a program that has the following specification:

- Declare two variables named *number1* and *number2* of the data type ***byte***
- Declare a variable named *addition* of the data type ***int.*** This variable should contain the sum of the two *byte* variables.

  Your output should look similar to as shown:



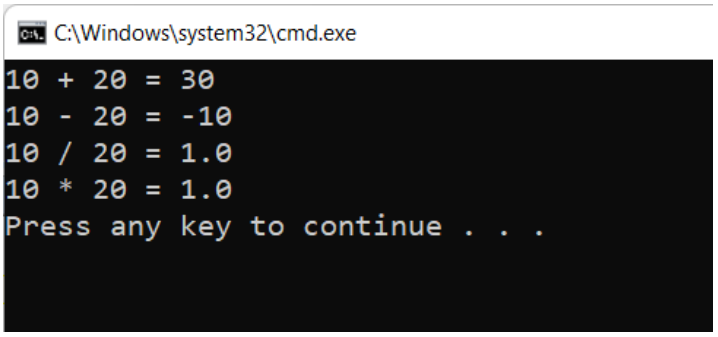
- Test this with multiple values to confirm that it works correctly.
- Amend your program so that it performs the following:

  Stores number1 minus number2 into a ***long*** variable

  Stores number1 divided by number2 into a ***double*** variable

  Stores number1 multiplied by number2 into a ***float*** variable

  Your updated output should look similar to as shown:



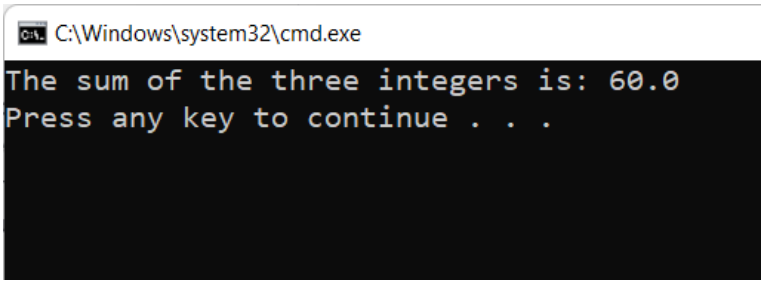
Exercise 12

Goal: Create a program in Java that performs calculation on three *Int* variables. Ensure it compiles and runs successfully.

Create a new file called *IntCalc* for this exercise. Following the guide below, create a program that has the following specification:

- Declare three variables named *num1, num2* and *num2* of the data type ***int***
- Declare a variable named *result* of the data type ***double.*** This variable should contain the sum of the three int variables.

  Your output should look similar to as shown:



Amend your program so that it also produces multiply, subtract, divide and modulus. Output your results to the console.