

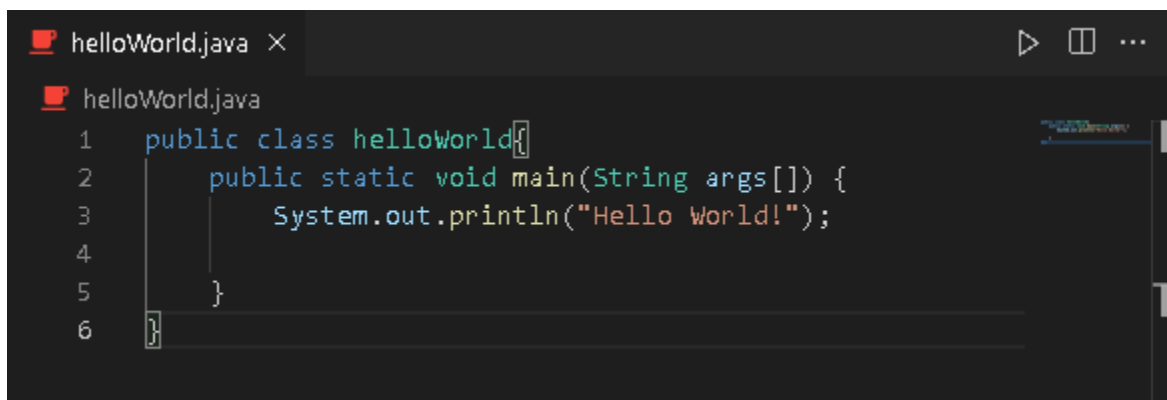
## WEEK - 01

HAFEEZ MOHAMAD

B172195

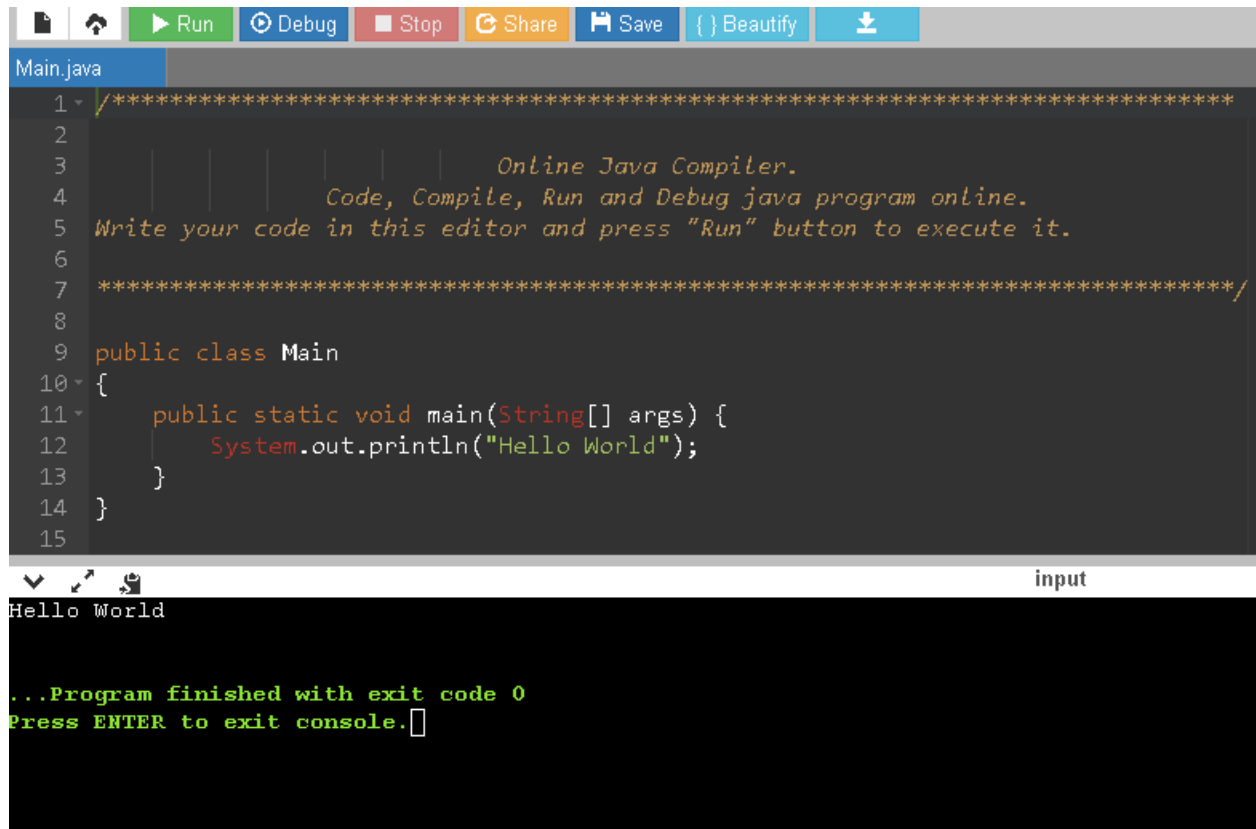
**01. Write a Java program to print “Hello World”.**

**CODE:**

A screenshot of a Java IDE window titled 'helloWorld.java'. The code is as follows:

```
1 public class helloWorld {  
2     public static void main(String args[]) {  
3         System.out.println("Hello World!");  
4     }  
5 }  
6 }
```

## OUTPUT:



The screenshot displays an online Java compiler interface. At the top, there is a toolbar with buttons for Run, Debug, Stop, Share, Save, Beautify, and a download icon. Below the toolbar, the file name 'Main.java' is shown. The code editor contains a Java program with 15 lines. Lines 1-7 are comments enclosed in a multi-line comment block. Lines 8-15 contain the code for a 'Main' class with a 'main' method that prints 'Hello World'. Below the code editor, there is a console window. The console shows the output 'Hello World' and a message indicating the program finished with exit code 0, prompting the user to press ENTER to exit the console.

```
1 - /*****
2
3         Online Java Compiler.
4         Code, Compile, Run and Debug java program online.
5         Write your code in this editor and press "Run" button to execute it.
6
7 *****/
8
9 public class Main
10 {
11     public static void main(String[] args) {
12         System.out.println("Hello World");
13     }
14 }
15
```

input

Hello World

...Program finished with exit code 0  
Press ENTER to exit console.

**02. Write a Java program that prints all the real and imaginary solutions to the quadratic equations  $ax^2 + bx + c = 0$ . Read in a, b, c and use the quadratic formula.**

**CODE:**

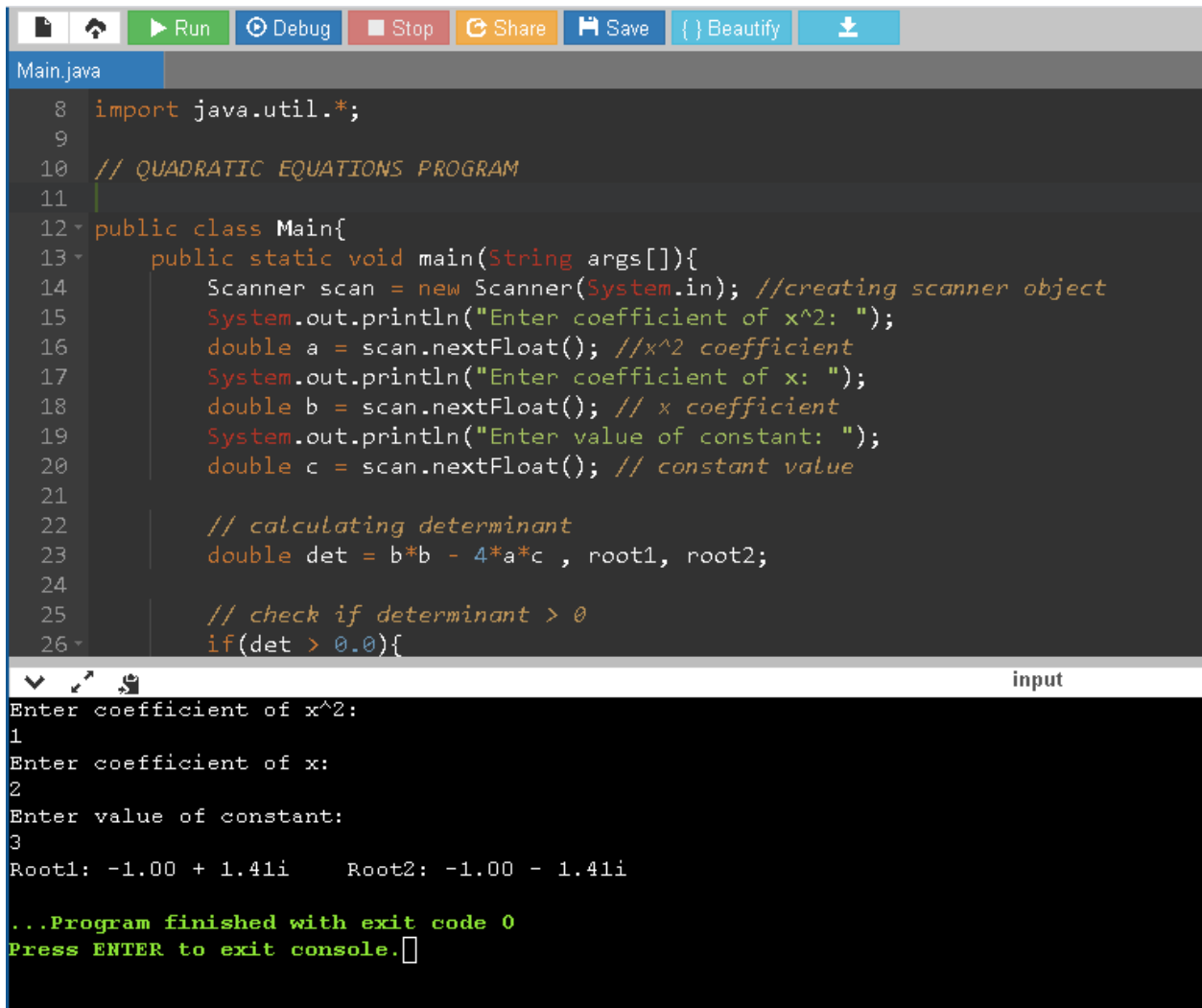
helloWorld.java quadraticEquations.java ×

quadraticEquations.java

```
1  import java.util.*;
2
3  // QUADRATIC EQUATIONS PROGRAM
4
5  public class quadraticEquations{
6      public static void main(String args[]) {
7          Scanner scan = new Scanner(System.in); //creating scanner object
8          System.out.println("Enter coefficient of x^2: ");
9          double a = scan.nextFloat(); //x^2 coefficient
10         System.out.println("Enter coefficient of x: ");
11         double b = scan.nextFloat(); // x coefficient
12         System.out.println("Enter value of constant: ");
13         double c = scan.nextFloat(); // constant value
14
15         // calculating determinant
16         double det = b*b - 4*a*c , root1, root2;
17
18         // check if determinant > 0
19         if(det > 0.0){
20             // roots are real and distinct
21             root1 = (-b - Math.sqrt(det)) / (2*a);
22             root2 = (-b + Math.sqrt(det)) / (2*a);
23
24             // roots output
25             System.out.format("Root1: %.2f\tRoot2: %.2f",root1,root2);
26         }
27
28         // check if determinant = 0
29         else if(det == 0.0){
30             // roots are real and equal
31             root1 = root2 = -b / (2*a);
32
33             // roots output
```

```
32
33     // roots output
34     System.out.format("Root1: %.2f\tRoot2: %.2f",root1,root2);
35 }
36
37 // check if determinant < 0
38 if(det < 0.0){
39     // roots are complex and distinct
40     double realPart = -b / (2*a);
41     double imgPart = Math.sqrt(-det) / (2*a);
42
43     // roots output
44     System.out.format("Root1: %.2f + %.2fi\tRoot2: %.2f - %.2fi",realPart, imgPart,realPart, imgPart);
45
46 }
47
48
49
```

**OUTPUT:**



The image shows a screenshot of a Java IDE. The top toolbar includes icons for Run, Debug, Stop, Share, Save, Beautify, and a download icon. The file name 'Main.java' is displayed. The code is as follows:

```
8 import java.util.*;
9
10 // QUADRATIC EQUATIONS PROGRAM
11
12 public class Main{
13     public static void main(String args[]){
14         Scanner scan = new Scanner(System.in); //creating scanner object
15         System.out.println("Enter coefficient of x^2: ");
16         double a = scan.nextFloat(); //x^2 coefficient
17         System.out.println("Enter coefficient of x: ");
18         double b = scan.nextFloat(); // x coefficient
19         System.out.println("Enter value of constant: ");
20         double c = scan.nextFloat(); // constant value
21
22         // calculating determinant
23         double det = b*b - 4*a*c , root1, root2;
24
25         // check if determinant > 0
26         if(det > 0.0){
```

The output window, titled 'input', shows the program's execution:

```
Enter coefficient of x^2:
1
Enter coefficient of x:
2
Enter value of constant:
3
Root1: -1.00 + 1.41i    Root2: -1.00 - 1.41i

...Program finished with exit code 0
Press ENTER to exit console.
```

**03. Write a Java program to implement calculator operations.**

**CODE:**

```

calculatorOperations.java ×
calculatorOperations.java
1  import java.util.*;
2
3  // CALCULATOR OPERATIONS PROGRAM
4
5  public class calculatorOperations{
6      public static void main(String args[]){
7          Scanner scan = new Scanner(System.in); //creating scanner object
8          System.out.println("Enter First value: ");
9          double a = scan.nextFloat(); // first value
10         System.out.println("Enter Second value: ");
11         double b = scan.nextFloat(); // second value
12         boolean flag = true;
13         while(flag){
14             System.out.println("0.Exit 1.Addition 2.subtraction 3.Multiplication 4.Division 5.Remainder");
15             System.out.println("Enter your Option: ");
16             int opt = scan.nextInt(); // input option
17
18             switch(opt){
19                 case 0:
20                     flag = false; // to end while loop
21                     break;
22                 case 1:
23                     System.out.format("Addition: %f + %f = %f",a,b,a+b); //Addition
24                     break;
25                 case 2:
26                     System.out.format("Subtraction: %f - %f = %f",a,b,a-b); //Subtraction
27                     break;
28                 case 3:
29                     System.out.format("Multiplication: %f x %f = %f",a,b,a*b); //Multiplication
30                     break;
31                 case 4:
32                     System.out.format("Division: %f / %f = %f",a,b,a/b); //Division
33                     break;
34                 case 5:
35                     System.out.format("Remainder: %f % %f = %f",a,b,a%b); //Remainder
36                     break;
37                 default:
38                     System.out.println("Invalid Option Encountered!!"); //Default
39                     break;
40             }
41         }
42     }
43 }

```

**OUTPUT:**

```

Main.java
32         case 2:
33             System.out.format("Subtraction: %f - %f = %f",a,b,a-b); //Subtraction
34             break;
35         case 3:
36             System.out.format("Multiplication: %f x %f = %f",a,b,a*b); //Multiplication
37             break;
38         case 4:
39             System.out.format("Division: %f / %f = %f",a,b,a/b); //Division
40             break;
41         case 5:

input
Enter First value:
10
Enter Second value:
2
0.Exit 1.Addition 2.subtraction 3.Multiplication 4.Division 5.Remainder
Enter your Option:
1
Addition: 10.000000 + 2.000000 = 12.0000000.Exit 1.Addition 2.subtraction 3.Multiplication 4.Division 5.Remainder
Enter your Option:
3
Multiplication: 10.000000 x 2.000000 = 20.0000000.Exit 1.Addition 2.subtraction 3.Multiplication 4.Division 5.Remainder
Enter your Option:
10
Invalid Option Encountered!!
0.Exit 1.Addition 2.subtraction 3.Multiplication 4.Division 5.Remainder
Enter your Option:
0

...Program finished with exit code 0
Press ENTER to exit console.

```

**04. Write a Java program to find prime factors of a given number.**

**CODE:**

```
primeFactors.java X
primeFactors.java
1  import java.util.*;
2
3  // FINDING PRIME FACTORS PROGRAM
4
5  public class primeFactors{
6      public static void main(String args[]){
7          Scanner scan = new Scanner(System.in); //creating scanner object
8          System.out.println("Enter your number: ");
9          int num = scan.nextInt(); // input number
10
11          int isPrime;// flag to check if number is a prime
12
13          // loop from 2 to the num (not from 1 because, 1 is not prime nor composite)
14          for(int i=2; i<=num; i++){
15              if(num%i == 0){
16                  int factor = i; // i is a factor
17                  // check if factor is prime
18                  isPrime = 1;
19
20                  for(int j=2; j<=(factor/2); j++){ //optimised code
21                      if(factor%j == 0){ //if other than 1 and itself is a factor then it is not a prime
22                          isPrime = 0;
23                          //break;
24                      }
25                  }
26                  if(isPrime == 1){ // check if num is prime
27                      System.out.format("%d is prime factor of %d\n",factor,num);
28                  }
29              }
30          }
31      }
32  }
33 }
```

Activate Windows  
Go to Settings to activate

**OUTPUT:**



The screenshot shows an online Java compiler interface. At the top, there is a toolbar with buttons for Run, Debug, Stop, Share, Save, Beautify, and a download icon. Below the toolbar, the file name 'Main.java' is displayed. The code editor contains the following Java code:

```
1  /*****
2
3      Online Java Compiler.
4      Code, Compile, Run and Debug java program online.
5      Write your code in this editor and press "Run" button to execute it.
6
7      *****/
8  import java.util.*;
9
10 // FINDING PRIME FACTORS PROGRAM
11
12 public class Main{
13     public static void main(String args[]){
14         Scanner scan = new Scanner(System.in); //creating scanner object
15         System.out.println("Enter your number: ");
```

Below the code editor, there is an input field labeled 'input' with the value '770'. The output console shows the following text:

```
Enter your number:
770
2 is prime factor of 770
5 is prime factor of 770
7 is prime factor of 770
11 is prime factor of 770

...Program finished with exit code 0
Press ENTER to exit console.
```

**05. Write a Java program to find whether given number is palindrome or not.**

**CODE:**

```
palindrome.java ×
palindrome.java
1  import java.util.*;
2
3  // PALINDROME PROGRAM
4
5  public class palindrome{
6      public static void main(String args[]){
7          Scanner scan = new Scanner(System.in); //creating scanner object
8          System.out.println("Enter your number: ");
9          int num = scan.nextInt(); // input number
10
11          int temp = num; //saving num in temporary variable
12          int rem; // to save last digits
13          int rev=0; // to save reversing numbers
14
15          //Reversing input number
16          while(temp!=0){
17              rem = temp%10; // taking out last digits
18              rev = (rev*10) + rem; // placing in rev variable
19              temp = temp/10; // removing last digit from given number
20          }
21
22          // checking if reverse number is equal to given number
23          if(num == rev)
24              System.out.format("%d is a palindrome\n", num);
25          else // if reverse is not equal to given number
26              System.out.format("%d is not a palindrome\n", num);
27      }
28  }
```

**OUTPUT:**

The screenshot shows an online Java compiler interface. At the top, there is a toolbar with buttons for Run, Debug, Stop, Share, Save, Beautify, and a download icon. Below the toolbar, the file name 'Main.java' is displayed. The code editor contains the following Java code:

```
3      Online Java Compiler.
4      Code, Compile, Run and Debug java program online.
5      Write your code in this editor and press "Run" button to execute it.
6
7      *****/
8      import java.util.*;
9
10     // PALINDROME PROGRAM
11
12     public class Main{
13     public static void main(String args[]){
14         Scanner scan = new Scanner(System.in); //creating scanner object
15         System.out.println("Enter your number: ");
16         int num = scan.nextInt(); // input number
17     }
```

Below the code editor, there is a console window. The console shows the output of the program:

```
input
Enter your number:
121
121 is a palindrome

...Program finished with exit code 0
Press ENTER to exit console.
```

**06. Write an application that declares 5 integers, determines and prints the largest and smallest in the group.**

**CODE:**

```
largestAndSmallest.java X
largestAndSmallest.java
1  import java.util.*;
2
3  // LARGEST AND SMALLEST PROGRAM
4
5  public class largestAndSmallest{
6      public static void main(String args[]){
7          Scanner scan = new Scanner(System.in); //creating scanner object
8
9          System.out.println("Enter number of values: ");
10         int n = scan.nextInt(); // number of values
11         int a[] = new int[n]; // declaring array with size n
12         int smallest = 9999; //smallest number
13         int largest = -1; // largest number
14
15         for(int i=0; i<n; i++){
16             System.out.format("Enter value of a[%d]: ",i);
17             a[i] = scan.nextInt();
18
19             // finding smallest
20             if(a[i] < smallest)
21                 smallest = a[i];
22
23             // finding largest
24             if(a[i] > largest)
25                 largest = a[i];
26         }
27         System.out.format("Smallest value: %d\n", smallest);
28         System.out.format("Largest value: %d\n", largest);
29     }
30 }
```

**OUTPUT:**

