JAVA PROGRAMMING

Dr. Hasin A Ahmed
Assistant Professor
Department of Computer Science
Gauhati University

First Java Program

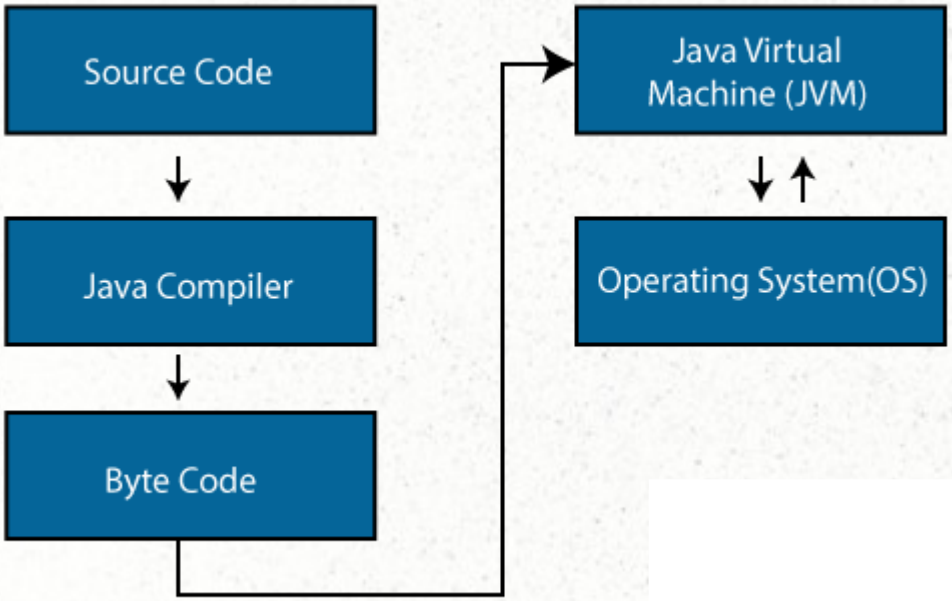
```
public class Main {
  public static void main(String[] args) {
    System.out.println("Hello World");
  }
}
```

```
C:\Users\Your Name>javac Main.java
```

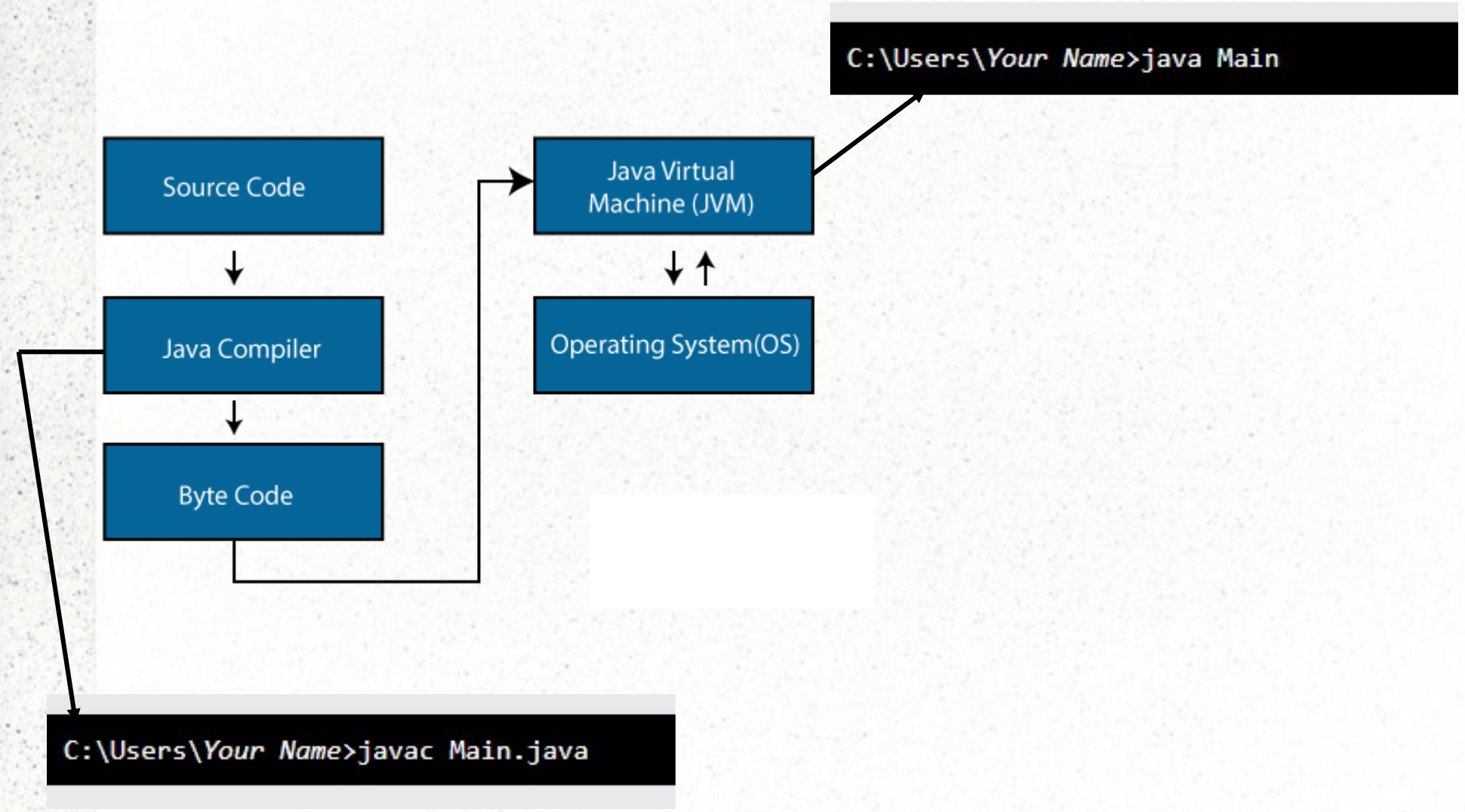
```
C:\Users\Your Name>java Main
```

```
Hello World
```

Compilation Process



Compilation Process



Comments

```
// This is a comment
System.out.println("Hello World");
System.out.println("Hello World"); // This is a comment
/* The code below will print the words Hello World
to the screen, and it is amazing */
System.out.println("Hello World");
```

Variables

- **int - stores integers (whole numbers), without decimals, such as 123 or -123**
- **float - stores floating point numbers, with decimals, such as 19.99 or -19.99**
- **char - stores single characters, such as 'a' or 'B'. Char values are surrounded by single quotes**
- **boolean - stores values with two states: true or false**
- **String - stores text, such as "Hello". String values are surrounded by double quotes**

Primitive data types

Data Type	Size	Description
`byte`	1 byte	Stores whole numbers from -128 to 127
`short`	2 bytes	Stores whole numbers from -32,768 to 32,767
`int`	4 bytes	Stores whole numbers from -2,147,483,648 to 2,147,483,647
`long`	8 bytes	Stores whole numbers from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
`float`	4 bytes	Stores fractional numbers. Sufficient for storing 6 to 7 decimal digits
`double`	8 bytes	Stores fractional numbers. Sufficient for storing 15 decimal digits
`boolean`	1 bit	Stores true or false values
`char`	2 bytes	Stores a single character/letter or ASCII values

Variables

- **Create a variable called myNum of type int and assign it the value 15**

```
int myNum = 15;

System.out.println(myNum);
```

If you don't want others (or yourself) to overwrite existing values, use the final keyword

```
final int myNum = 15;

myNum = 20;  // will generate an error
```

Variables

- To combine both text and a variable in print, use the + character:

```
int num = 34;

System.out.println("Value is: " + num);
```

If you don't want others (or yourself) to overwrite existing values, use the final keyword

```
final int myNum = 15;

myNum = 20;  // will generate an error
```

Variables

```
public class Main {
  public static void main(String[] args) {
    int x = 5;
    int y = 6;
    System.out.println( x + y);
  }
}
```

Variables

```
public class Main {
  public static void main(String[] args) {
    int x = 5;
    int y = 6;
    System.out.println( “Value is:”+x + y);
  }
}
```

Variables

```
int x = 5, y = 6, z = 50;
System.out.println(x + y + z);
```

```
int x, y, z;
x = y = z = 50;
System.out.println(x + y + z);
```

Variables

The general rules for naming variables are:

- **Names can contain letters, digits, underscores, and dollar signs**
- **Names must begin with a letter**
- **Names should start with a lowercase letter and it cannot contain whitespace**
- **Names can also begin with $ and _ (but we will not use it in this tutorial)**
- **Names are case sensitive ("myVar" and "myvar" are different variables)**
- **Reserved words (like Java keywords, such as int or boolean) cannot be used as names**

Variables

Dealing with boolean variables:

```
boolean isJavaFun = true;

boolean isFishTasty = false;

System.out.println(isJavaFun);     // Outputs true

System.out.println(isFishTasty);   // Outputs false
```

Variables

Dealing with characters and strings:

```
char myGrade = 'B';
System.out.println(myGrade);

char myVar1 = 65, myVar2 = 66, myVar3 = 67;
System.out.println(myVar1);
System.out.println(myVar2);
System.out.println(myVar3);

String greeting = "Hello World";
System.out.println(greeting);
```

Variables

In Java, there are two types of casting:

- **Widening Casting (automatically) - converting a smaller type to a larger type size**
- **byte -> short -> char -> int -> long -> float -> double**
- **Narrowing Casting (manually) - converting a larger type to a smaller size type**
- **double -> float -> long -> int -> char -> short -> byte**

Variables

Widening casting:

```
public class Main {
  public static void main(String[] args) {
    int myInt = 9;
    double myDouble = myInt; // Automatic casting: int to double
    System.out.println(myInt);      // Outputs 9
    System.out.println(myDouble);   // Outputs 9.0
  }
}
```

Variables

Narrowing casting:

```
public class Main {
  public static void main(String[] args) {
    double myDouble = 9.78d;
    int myInt = (int) myDouble; // Manual casting: double to int

    System.out.println(myDouble);   // Outputs 9.78
    System.out.println(myInt);      // Outputs 9
  }
}
```

Operators

Operator	Name	Description	Example
+	Addition	Adds together two values	x + y
-	Subtraction	Subtracts one value from another	x - y
*	Multiplication	Multiplies two values	x * y
/	Division	Divides one value by another	x / y
%	Modulus	Returns the division remainder	x % y
++	Increment	Increases the value of a variable by 1	++x
--	Decrement	Decreases the value of a variable by 1	--x

Comparison Operators

Operator	Name	Example
==	Equal to	x == y
!=	Not equal	x != y
>	Greater than	x > y
<	Less than	x < y
>=	Greater than or equal to	x >= y
<=	Less than or equal to	x <= y

Logical Operators

Operator	Name	Description	Example
&&	Logical and	Returns true if both statements are true	x < 5 && x < 10
||	Logical or	Returns true if one of the statements is true	x < 5 || x < 4
!	Logical not	Reverse the result, returns false if the result is true	!(x < 5 && x < 10)

String

A String variable (object) contains a collection of characters surrounded by double quotes:

- **Create a variable of type String and assign it a value:**

```
String greeting = "Hello";
```

- **the length of a string can be found with the length() method**

```
String txt = "ABCDEFG";

System.out.println("Length: " + txt.length());
```

String

More String methods:

```
String txt = "Hello World";
System.out.println(txt.toUpperCase());
// Outputs "HELLO WORLD"
System.out.println(txt.toLowerCase());
// Outputs "hello world"
String txt = "Please locate where 'locate' occurs!";
System.out.println(txt.indexOf("locate"));
// Outputs 7
```

String

```
String firstName = "John";
String lastName = "Doe";
System.out.println(firstName + " " + lastName);
```

```
String firstName = "John ";
String lastName = "Doe";
System.out.println(firstName.concat(lastName));
```

Math class

Math.max(5, 10);

Math.min(5, 10);

Math.sqrt(64);

Math.abs(-4.7);

Math.random(); returns a random number between 0.0 (inclusive), and 1.0 (exclusive)

If else statement

```
public class Main {
  public static void main(String[] args) {
    int time = 22;
    if (time < 10) {
      System.out.println("Good morning.");
    } else if (time < 18) {
      System.out.println("Good day.");
    }  else {
      System.out.println("Good evening.");
    }
  }
}
```

Switch case statement

```
public class Main {
  public static void main(String[] args) {
    int day = 4;
    switch (day) {
      case 6:
        System.out.println("Today is Saturday");
        break;
      case 7:
        System.out.println("Today is Sunday");
        break;
      default:
        System.out.println("Looking forward to the Weekend");
    }
  }
}
```

While statement

```
public class Main {
  public static void main(String[] args) {
    int i = 0;
    while (i < 5) {
      System.out.println(i);
      i++;
    }
  }
}
```

For statement

```
public class Main {
  public static void main(String[] args) {
    // Outer loop.
    for (int i = 1; i <= 2; i++) {
      System.out.println("Outer: " + i); // Executes 2 times

      // Inner loop
      for (int j = 1; j <= 3; j++) {
        System.out.println(" Inner: " + j); // Executes 6 times (2 * 3)
      }
    }
  }
}
```

Do while statement

```
public class DoWhileExample {
public static void main(String[] args) {
    int i=1;
    do{
        System.out.println(i);
    i++;
    }while(i<=10);
}
}
```

For each statement

```
public class Main {
  public static void main(String[] args) {
    String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};
    for (String i : cars) {
      System.out.println(i);
    }
  }
}
```

Break statement

```
public class Main {
  public static void main(String[] args) {
    for (int i = 0; i < 10; i++) {
      if (i == 4) {
        break;
      }
      System.out.println(i);
    }
  }
}
```

Continue statement

```
public class Main {
  public static void main(String[] args) {
    int i = 0;
    while (i < 10) {
      if (i == 4) {
        i++;
        continue;
      }
      System.out.println(i);
      i++;
    }
  }
}
```

Console input

```
import java.util.Scanner;
class GetInputFromUser {
    public static void main(String args[])
    {
        // Using Scanner for Getting Input from User
        Scanner in = new Scanner(System.in);
        String s = in.nextLine();
        System.out.println("You entered string " + s);
        int a = in.nextInt();
        System.out.println("You entered integer " + a);
        float b = in.nextFloat();
        System.out.println("You entered float " + b);
    }
}
```

Thank you

Any Question

??????????