

1.

•
•
•

•
•

•
•
•

-
-
-
-
-

•
•
•
-

•
•
•
•
•
•

•
•
•

2.

```
public class Simple {  
    public static void main (String[] V) {}  
}
```

-

-

-

-

-
-
-
-
-

Example

```
int a, b, c; // Declares three ints, a, b, and c.  
int a = 10, b = 10; // Example of initialization  
byte B = 22; // initializes a byte type variable B.  
double pi = 3.14159; // declares and assigns a value of PI.  
char a = 'a'; // the char variable a is initialized with value 'a'
```

-
-

○

-
-
-
-

-

-



```
public class OperatorTest {
    public static void main(String[] args) {
        System.out.println(1/3); // 0
        System.out.println(1.0/3); // 0.333
        System.out.println(1/3.0); // 0.333
        System.out.println(1.0/3.0); // 0.333
        System.out.println((double)1/3); // 0.333
        System.out.println((double)(1/3)); // 0.0
        System.out.println((int)1.67); // 1
        System.out.println((double)3); // 3.0
        System.out.println((double)5); // 5.0
        System.out.println((int)1.34); // 1
        System.out.println((int)1.8); // 1

        double number = 1.67;
        int nearestInt = (int)(number + 0.5); // (int)2.17 = 2
        System.out.println(nearestInt); //

        double value = -3.64;
        int nearestNegInt = (int)(value - 0.5); // (int)-3.64
        System.out.println(nearestNegInt); //
    }
}
```

Operator's Precedence in Java

Operators	Precedence
<code>!, +, - (unary Operators)</code>	First (Highest)
<code>* , / , %</code>	Second
<code>+ , -</code>	Third
<code><, >, >=, ></code>	Fourth
<code>= =, !=</code>	Fifth
<code>&&</code>	Sixth
<code> </code>	Seventh
<code>= (assignment Operator)</code>	Lowest

```
import java.util.Scanner;
public class ReadingName {
    public static void main(String args[]) {
        Scanner scan = new Scanner(System.in);

        System.out.println("Please Enter your name: ");
        // The following statement gets the string value
        // enter as program input and then stores the val
        String name = scan.nextLine();
        System.out.println("Hello " + name);
    }
}
```

3.

```
public class BoolTest1 {
    public static void main(String[] args) {
        int age = 15;
        int year = 14;
        // Will this print true or false?
        System.out.println(age == year); // true or false
        year = 15;
        // Will this print true or false?
        System.out.println(age === year); // true or false
        // Will this print true or false?
        System.out.println(age != year); // true or false
    }
}
```

```
public class BoolTest2 {
    public static void main(String[] args) {
        int age = 15;
        int year = 14;
        // Will this print true or false?
        System.out.println(age < year); // false
        System.out.println(age > year); // true
        System.out.println(age <= year + 1); // true
        System.out.println(age - 1 <= year); // true
    }
}
```

```
// A single if statement
if (boolean expression)
    Do statement;
// A block if statement
if (boolean expression)
{
    Do Statement1;
    Do Statement2;
    ...
    Do StatementN;
}

public class MidtermTest {
    public static void main(String[] args) {
        boolean passedExam = false;
        if(passedExam) {
            System.out.println("Good job studying!");
        }
        if(!passedExam) {
            System.out.println("Study harder next time!");
        }
        System.out.println("Goodbye!");
    }
}
```

```
public class NestedConditional {
    public static void main(String[] args) {
        boolean sunny = false;
        boolean hot = false;
        if(sunny) {
            if(hot)
                System.out.println("Head for the beach!");
            else
                System.out.println("Bring your umbrella!");
        }
        System.out.println("Done");
    }
}
```

```
// 3 way choice with else if
if (boolean expression)
{
    statement1;
}
else if (boolean expression)
{
    statement2;
}
else
{
    statement3;
}
```