Every **variable** in a Java program must be declared before it is used for the first time

```
String firstName;
```

```
firstName;
```

```
firstName = "Bob";
```

`firstName`

Bob

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

Output:

Bob

```
String firstName;
```

data Type variable name

```
String firstName;
```

Code Refactoring

```
public class SpongeBob {
    public static void main(String[] args) {
        System.out.println("SpangeBob: Hello Star");
        System.out.println("Patrik: Hi yellow boy");
        System.out.println("SpangeBob: Are you learning coding?");
        System.out.println("Patrik: Yes,but i have one problem");
        System.out.println("SpangeBob: Which?");
        System.out.println("Patrik: My computer is not working under the sea!))");
    }
}
```

```
 public static void main(String args[]) {
      String student1 = "Anna";
      String student2 = "Ari";
      String student3 = "Rocio";
      String student4 = "Selvin";
      String student5 = "Frank";

      String javaClass = student1 + student2 + student3 + student4 + student5;

      System.out.println(javaClass);
}
```

TODO:

1. Create a java class *Box*
2. Declare String variables: *box*
3. Assign values as *flowers*
4. Print variable *box*
5. Reassign value as *books*
6. Print variable *box*

```java
import java.util.Scanner;

public class Input {
public static void main(String[] args) {

    Scanner sc = new Scanner(System.in);
    String input = sc.nextLine();

    System.out.println(input);

    }
}
```

TODO:

1. Create a Java program *SimpleSiri*
2. It should ask:
   Hi there, what's your name?
   your answer..
   Nice to meet you <answer> !
3. It should ask:
   What are you studying now?
   your answer..
   Oh, <answer> is great subject to study!
4. It should ask:
   What's your favorite movie?
   your answer..
   I heard about it. <answer> is a great movie..

Java Identifiers

- An identifier (a name) can contain only letters , digits, _ (underscore)

Java Identifiers

- An identifier (a name) can contain only letters , digits, _ (underscore)
- An identifier must begin with a letter or the symbols $ or _ (underscore)

Java Identifiers

- An identifier (a name) can contain only letters , digits, _ (underscore)
- An identifier must begin with a letter or the symbols $ or _ (underscore)
- Java is case sensitive
- Good practice to follow – names of classes start with an uppercase letter, and the names of variables and methods start with a lowercase letter

Java Identifiers

- An identifier (a name) can contain only letters , digits, _ (underscore)
- An identifier must begin with a letter or the symbols $ or _ (underscore)
- Java is case sensitive
- Good practice to follow – names of classes start with an uppercase letter, and the names of variables and methods start with a lowercase letter
- Keywords or reserved words cannot be used as the names of variables, classes or methods.

Assignment Statements

- Use an **assignment statement** to give a variable value or to change it

Assignment Statements

- Use an **assignment statement** to give a variable value or to change it
- The equal sign, =, is called the **assignment operator**

Assignment Statements

- Use an **assignment statement** to give a variable value or to change it
- The equal sign, =, is called the **assignment operator**
- **Variable = Expression;**

```java
public class Box {
  public static void main(String args[]) {

    String item1 = "book\n";
    String item2 = "cup\n";
    String item3 = "glasses\n";
    String item4 = "pen";

    String box = item1 + item2 + item3 + item4;
    System.out.println(box);

    }
}
```

```
public class EggBasket
{
   public static void main(String[] args)
   {
      int numberOfBaskets, eggsPerBasket, totalEggs;

      numberOfBaskets = 10;
      eggsPerBasket = 6;

      totalEggs = numberOfBaskets * eggsPerBasket;

      System.out.println("If you have");
      System.out.println(eggsPerBasket + " eggs per basket and");
      System.out.println(numberOfBaskets + " baskets, then");
      System.out.println("the total number of eggs is " + totalEggs);
   }
}
```

Variable declarations

Assignment statement

Sample Screen Output

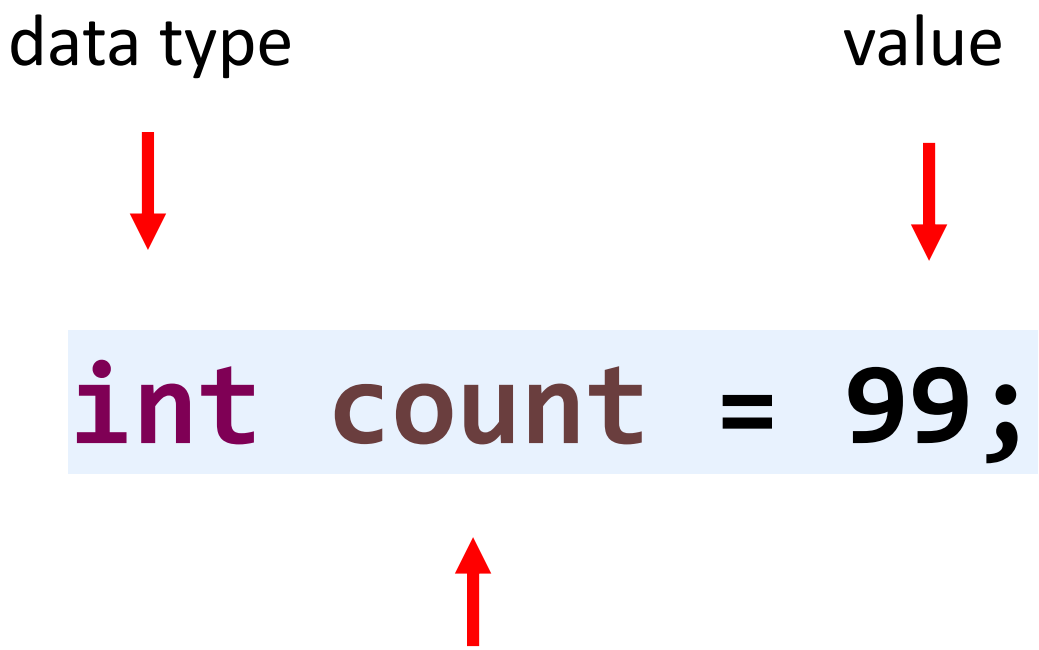
```
If you have
6 eggs per basket and
10 baskets, then
the total number of eggs is 60
```

Data Types

Characters	Hello World, ‘A’, !, $
Numeric	0, 1, 1.5, 10e2
Logical Conditions	true, false

Primitive Types

Type Name	Kind of Value	Memory Used	Range of Values
`byte`	Integer	1 byte	–128 to 127
`short`	Integer	2 bytes	–32,768 to 32,767
`int`	Integer	4 bytes	–2,147,483,648 to 2,147,483,647
`long`	Integer	8 bytes	–9,223,372,036,8547,75,808 to 9,223,372,036,854,775,807
`float`	Floating-point	4 bytes	$\pm 3.40282347 \times 10^{+38}$ to $\pm 1.40239846 \times 10^{-45}$
`double`	Floating-point	8 bytes	$\pm 1.79769313486231570 \times 10^{+308}$ to $\pm 4.94065645841246544 \times 10^{-324}$
`char`	Single character (Unicode)	2 bytes	All Unicode values from 0 to 65,535
`boolean`		1 bit	True or false


data type
value
int count = 99;
identifier

Primitive variables for whole numbers:

```
byte
short
int
long
```

Primitive variables for floating numbers:

```
float
double
```

Character

`char`

```
char h = 'H';
char i = 'I';
```

Boolean

```
boolean
```

arithmetic operators

\+ - * / %

TODO:

1. Create a Java program *Addition*
2. Declare and assign values for two int variables

```
int num1 = 5;
int num2 = 6;
```

3. Declare third int variable and assign value as sum of two previous variables

```
int result = num1 + num2;
```

4. Print result variable

TODO:

- For previous class *Addition* add Scanner to take user input for two int variables

```
.nextInt();
```

TODO:

1. Create a Java program *Subtraction*
2. Declare and assign values for two int variables
3. Declare third int variable and assign value as result of subtraction of two previous variables

```
int result = num1 - num2;
```

4. Print result variable

TODO:

- For previous class *Subtraction* add Scanner to take user input for two int variables

TODO:

- Create Java program to multiply two number from user

output ex:

```
6
*
7
=
42
```

TODO:

- Create Java program to divide two number from user

output ex:

```
25
/
5
=
5
```

whole number is for whole numbers

```
int num1 = 89;
int num2 = 10;

System.out.println(num1/num2);

Output: 8
```

Specialized Assignment Operators

+= -= *= /=

```
public class EggBasket
{
   public static void main(String[] args)
   {
      int numberOfBaskets, eggsPerBasket, totalEggs;

      numberOfBaskets = 10;
      eggsPerBasket = 6;

      totalEggs = numberOfBaskets * eggsPerBasket;

      System.out.println("If you have");
      System.out.println(eggsPerBasket + " eggs per basket and");
      System.out.println(numberOfBaskets + " baskets, then");
      System.out.println("the total number of eggs is " + totalEggs);
   }
}
```

Variable declarations

Assignment statement

Sample Screen Output

```
If you have
6 eggs per basket and
10 baskets, then
the total number of eggs is 60
```

TODO:

- Add Scanner to *EggBasket* class
- Get input from user for numberOfBaskets variable

Remainder

%

A Change-Making Program

Requirement

- Write a program that accepts from a user whole number from 1 to 99
- The program responds by telling the user one combination of coins that equals that amount of change

```
Enter a whole number from 1 to 99.
I will find a combination of coins
that equals that amount of change.
87
87 cents in coins:
3 quarters
1 dime
0 nickels and
2 pennies
```