ELSEWEDY UNIVERSITY OF TECHNOLOGY
- POLYTECHNIC OF EGYPT -

Object Oriented Programming

Lecture 1: Introduction Java

Table of Contents

- Introduction to Java
- Java Programming Basics
- Packages
- Conditions and Loops
- Strings
- Arrays
- Collections
- Files
- Methods

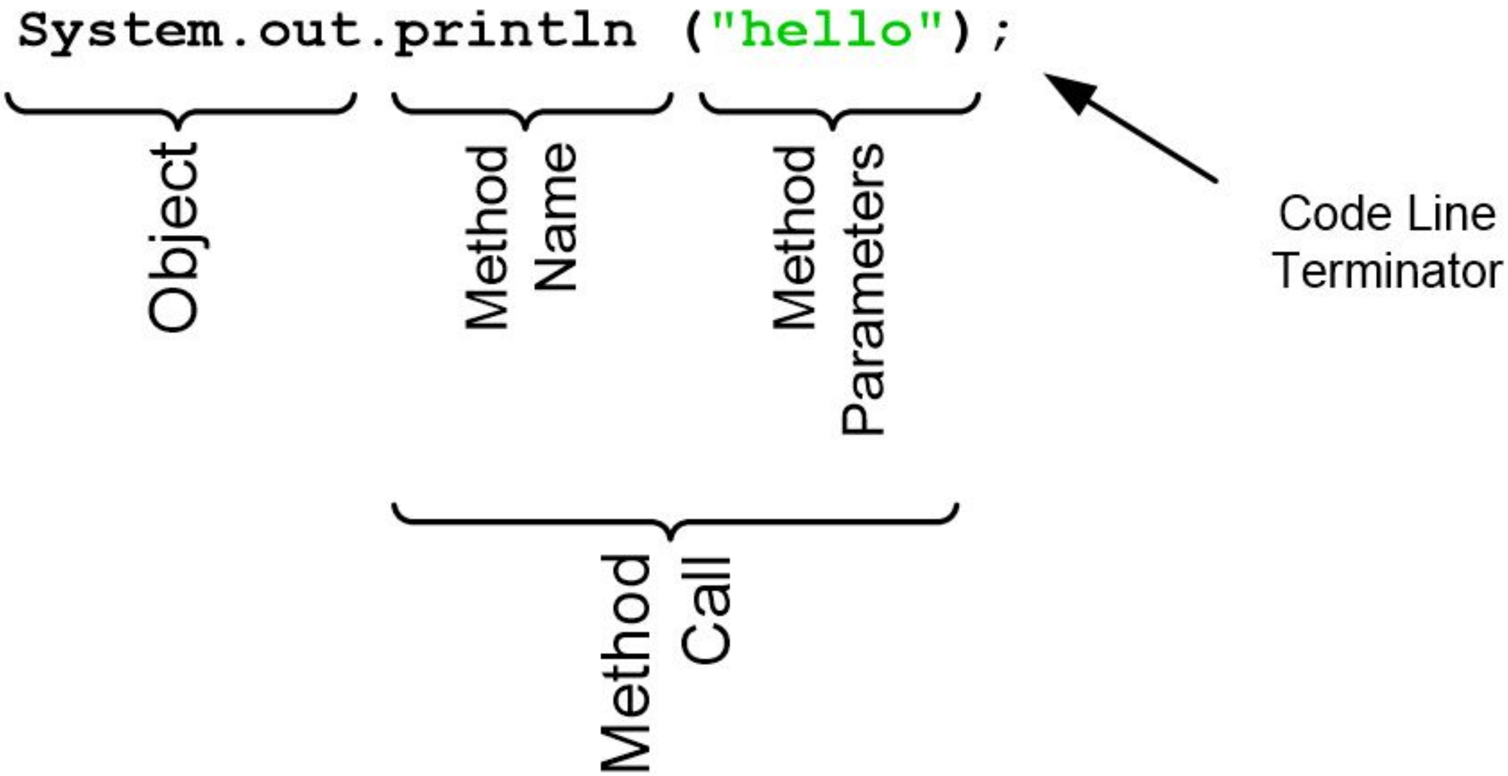
ELSEWEDY UNIVERSITY OF TECHNOLOGY
- POLYTECHNIC OF EGYPT -

Introduction to Java

Java Hello Program

```java
public class StudyJava
{
    /*program main should be static and
    defined once in the whole program*/
    public static void main (String[] args)
    {
        //print to console using println method
        System.out.println("hello");
    }
}
```

Java Hello Program



```
System.out.println ("hello");
```

Object

Method Name

Method Parameters

Code Line Terminator

Method Call

Class, Object, Variable and Methods

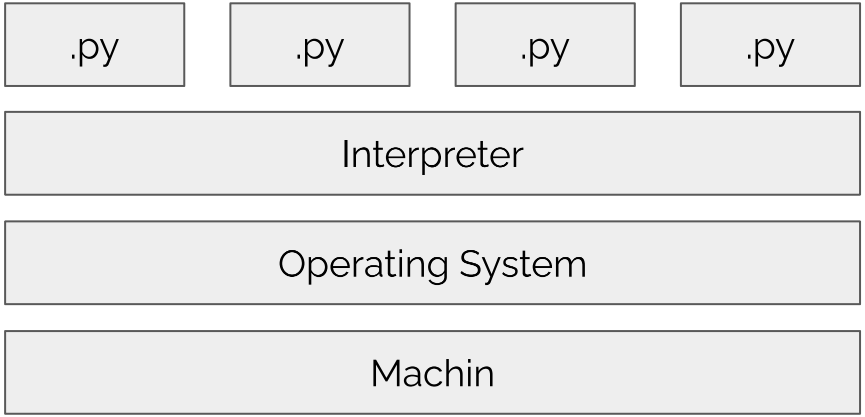
```java
public class StudyJava {
    public static void main (String[] args) {
        String a = new String("Hello Java"); // Create object a of class String
        String b = "Java is OOP language"; // Another simple way to create string object
        System.out.println(a); // Call method println in System.out object
        System.out.println(b);
        int length; // Define native variable length of type int
        length = a.length(); // call method length() in object a
        System.out.println(length);
    }
}
```

Java Programming Language

- Object Oriented language
- Java program consists of classes only
- There is no way to define standalone function
- Java is multi-platform language for Windows, Linux, Unix, Mac OS, Android
- Java is non-native, means it does not produce executable files like C/C++
- Java produces .class files instead of .exe
- Class files works under any operating system while EXE couldn't
- Class files requires Java Virtual Machine (JVM) to run while EXE files can run alone

Python Programming

Python is script based language needs interpreter to run.



.py

.py

.py

.py

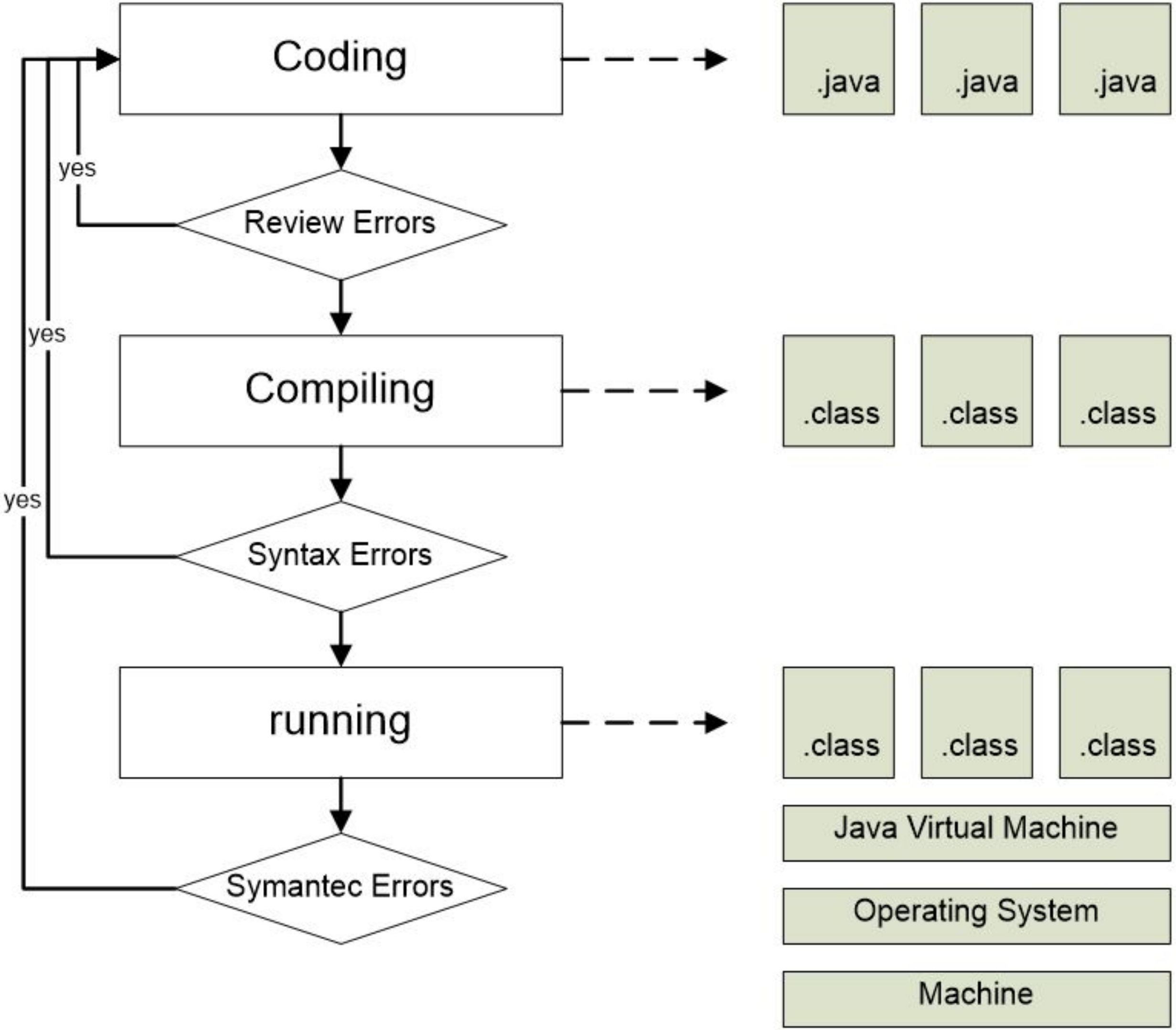
Interpreter

Operating System

Machin

Java Programming

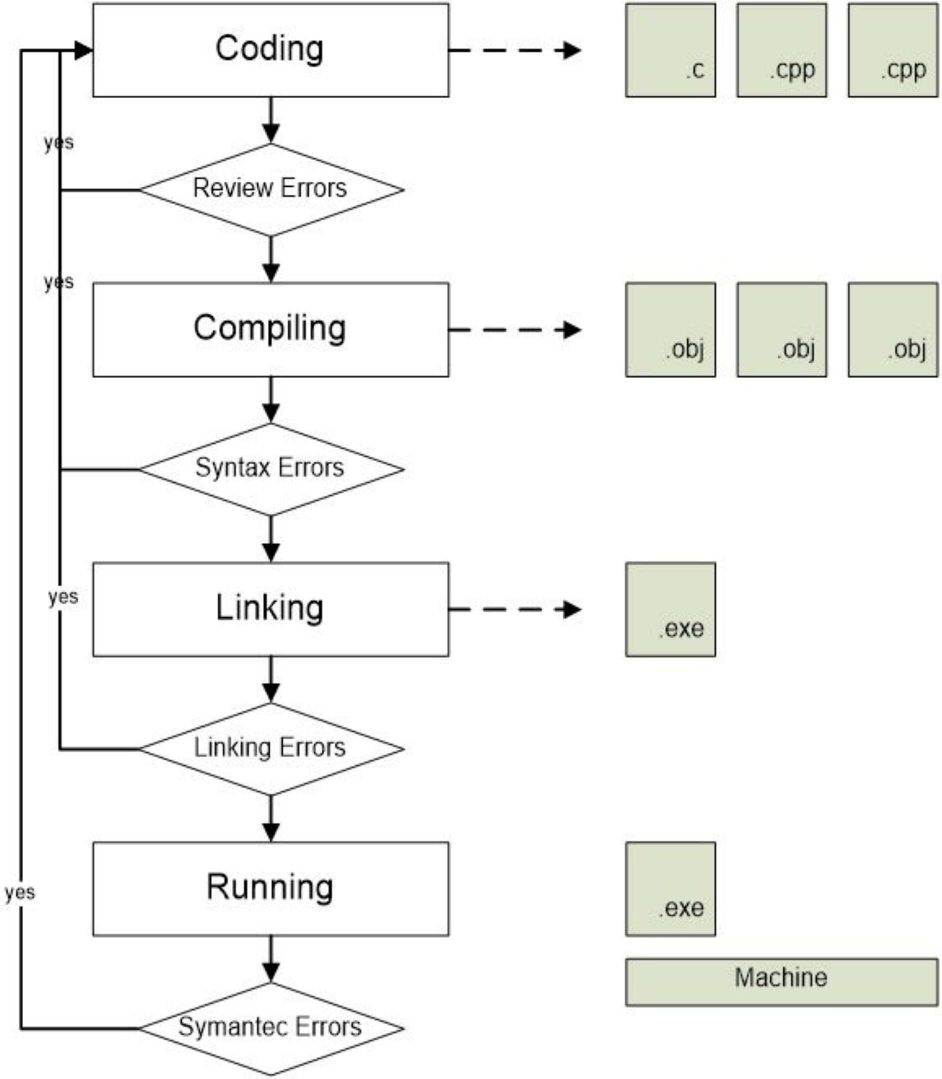
Normally all program class files are compressed in one file .jar in order to exchange the program easily.



C/C++ Programming

In C/C++ the compiler convert the program to machine code. This allow direct/fast execution on machine without any middleware layer.

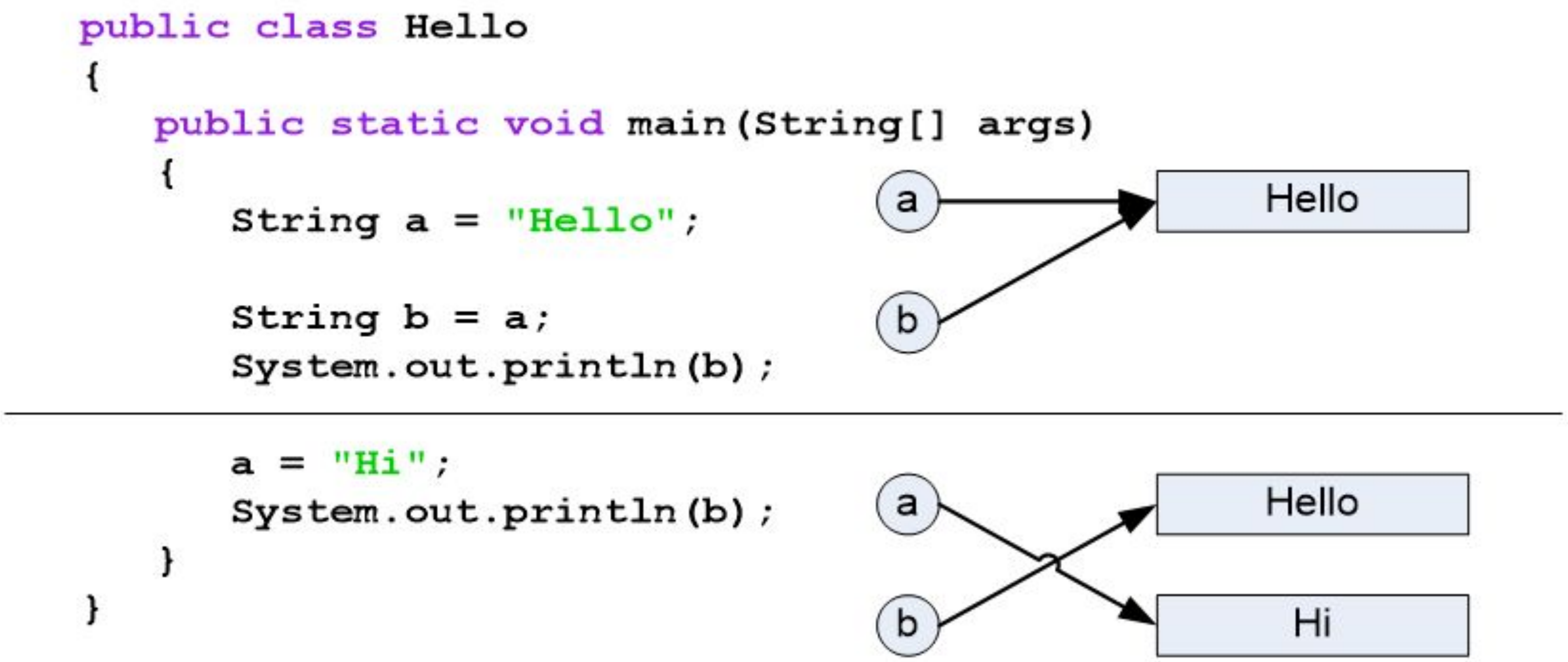
Hint: Now, python has transparent compiler that compile programs to run directly on machine.



Understanding Value and Reference

```java
public class StudyJava {
    public static void main(String[] args) {
        String a = "Hello"; // Assign to a reference to value "Hello"
        String b = a; // Now b has the same reference inside a
        System.out.println(b); // Print b --> "Hello"
        a = "Hi"; // Now a is assigned another reference rence to value "Hi"
        System.out.println(b); //Print b --> "Hello"
    }
}
```

Understanding Value and Reference



```
public class Hello
{
    public static void main(String[] args)
    {
        String a = "Hello";

        String b = a;
        System.out.println(b);
```

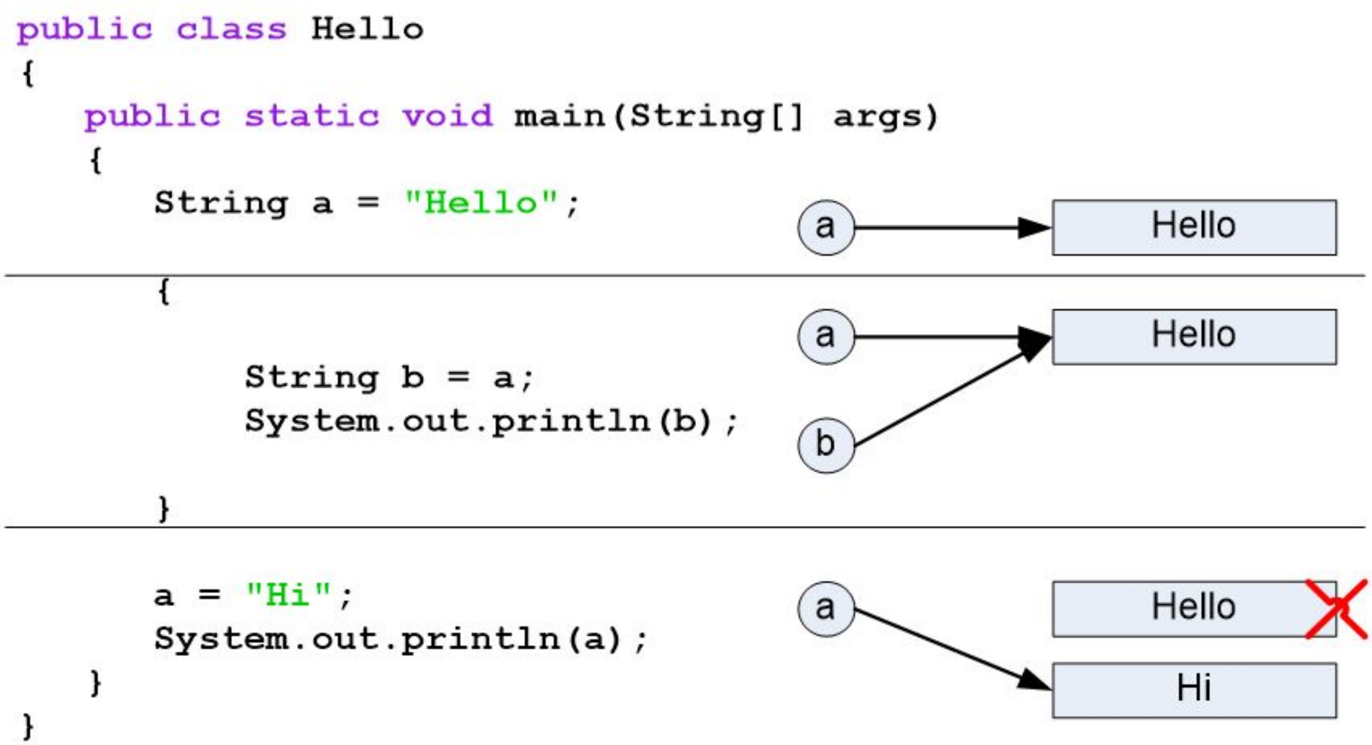
```
        a = "Hi";
        System.out.println(b);
    }
}
```

Variable Scope

```java
public class StudyJava {
 public static void main(String[] args) {
    String a = "Hello";
    {
      String b = a;
    }

    System.out.println(b);
    //Out of Scope
  }
}
```

```java
public class StudyJava {
 public static void main(String[] args) {
    String a = "Hello";
    {
      String b = a;
      System.out.println(b);
    }
  }
}
```

Garbage Collector



```
public class Hello
{
    public static void main(String[] args)
    {
        String a = "Hello";
        {

            String b = a;
            System.out.println(b);

        }

        a = "Hi";
        System.out.println(a);
    }
}
```

ELSEWEDY UNIVERSITY OF TECHNOLOGY
- POLYTECHNIC OF EGYPT -

Java Programming Basics

Java Programming Basics

- Printing Strings
- Variables and Constants
- Primitive Data Types
- Expressions
- Data Conversion
- Interactive Programs
- Working with Packages

Printing Strings

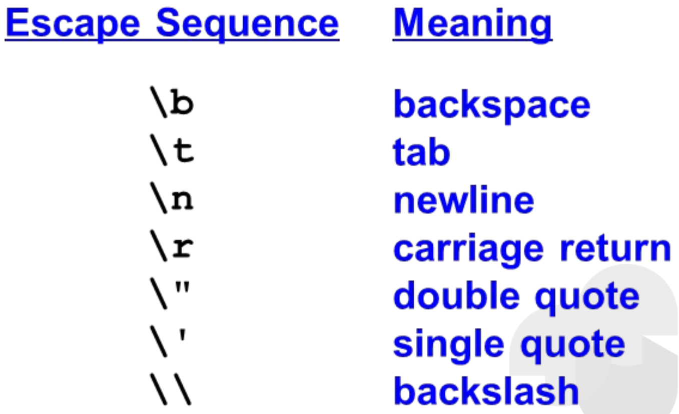
```java
public class StudyJava {
    public static void main(String[] args) {
        System.out.print("Welcome ");
        System.out.println("to Java World");
        System.out.println("Java is OOP Language");
    }
}
```

Printing Strings and Numbers

```java
public class StudyJava {
    public static void main(String[] args) {
        System.out.println ("Welcome to " + "computer world:");
        System.out.println ();
        System.out.println ("Letters in the Hawaiian alphabet: 12");
        System.out.println ("Dialing code for Antarctica: " + 672);
        System.out.println ("Leonardo da Vinci invented the parachute at: " + 1515);
        System.out.println ("Speed of ketchup: " + 40 + " km per year");
        System.out.println ("5 and 5 is " + 5 + 5);
        System.out.println ("5 plus 5 is " + (5 + 5));
        System.out.println(5 + 5 + " is ???");
    }
}
```

Printing Strings, Special Characters

```
public class StudyJava {
    public static void main(String[] args) {
        System.out.println ("Code compiles clean,\n\tNo syntax errors found,\n" +
        "Algorithms hum,\n\tBut runtime's  \"too slow\",\n\t" +
        "So I'll optimize loops\n\tFor speedier " +
        "execution.");
    }
}
```



Escape Sequence	Meaning
\b	backspace
\t	tab
\n	newline
\r	carriage return
\"	double quote
\'	single quote
\\	backslash

Variables

```java
public class StudyJava {
    public static void main(String[] args) {
        int width = 100;
        int height = 200;
        int area;
        area = width * height;
        System.out.println("Rectangle area: " + area);
    }
}
```

Variables and Constants

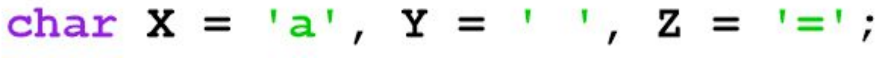
```java
public class StudyJava {
    public static void main(String[] args) {
        final double PI = 3.143;
        double radius = 100;
        double area;
        area = PI * radius * radius;
        System.out.println("Circle area:  " + area);
    }
}
```

Primitive Data Types

Type	Storage	Min Value	Max Value	Precision
byte	8 Bit	-128	127	
short	16 Bit	-32,768	32,767	
int	32 Bit	-2,147,483,648	2,147,483,647	
long	64 Bit	$-9x10^{18}$	$9x10^{18}$	
float	32 Bit	$-3.4x10^{38}$	$3.4x10^{38}$	7 Digits
double	64 Bit	$-1.7x10^{308}$	$1.7x10^{308}$	15 Digits

Characters

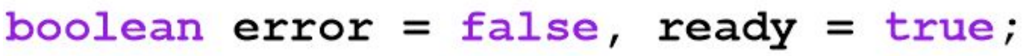
- (char) is 16 bit numeric value.
- It is usually used to store characters codes
- Characters codes is defined by UNCODE system



```
char X = 'a', Y = ' ', Z = '=';
```

Boolean

- Numeric Value used to hold either 0 or 1
- 0 represents the false state
- 1 represents the true state



```
boolean error = false, ready = true;
```

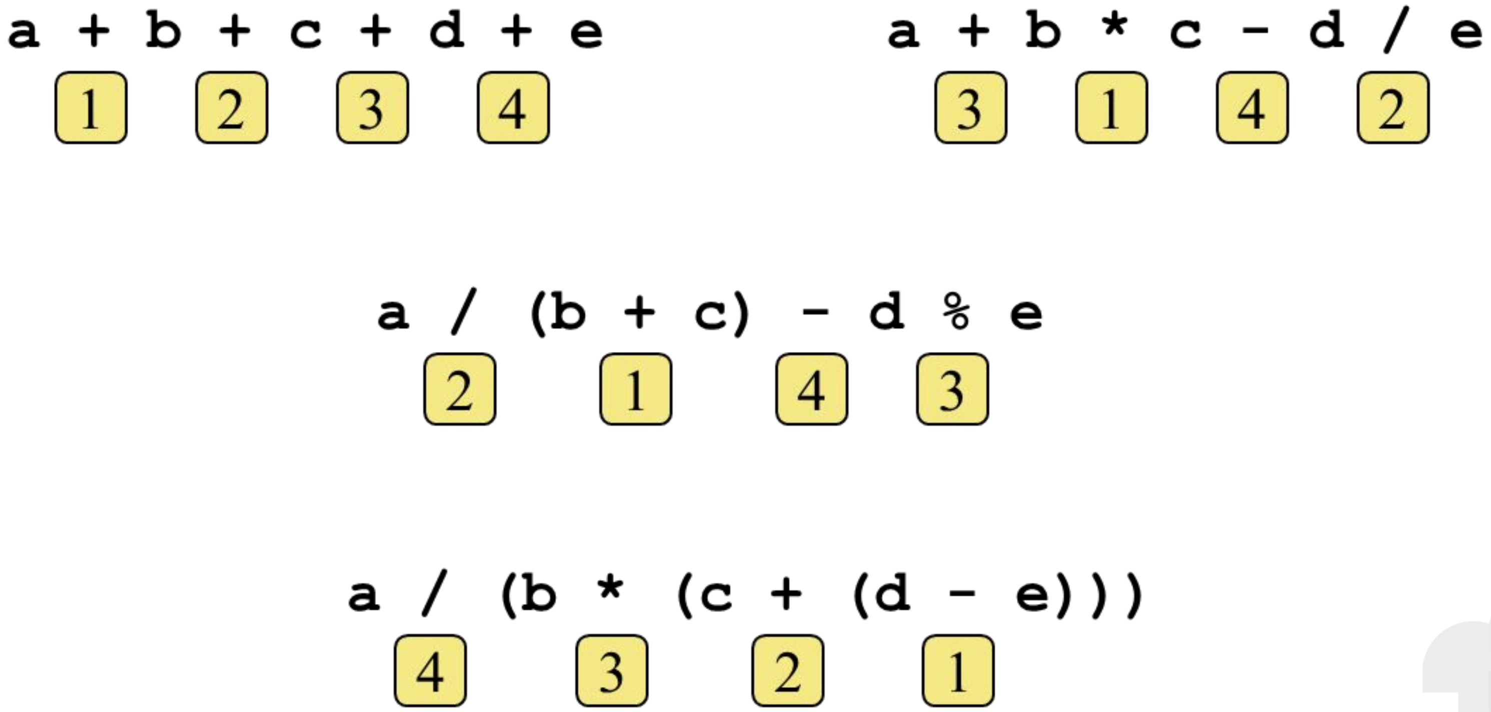
Expressions

```java
public class StudyJava {
    public static void main(String[] args) {
        int x = 5, y = 3, z;
        z = x + y; System.out.println("Z = " + z);
        z = x - y; System.out.println("Z = " + z);
        z = x * y; System.out.println("Z = " + z);
        z = x / y; System.out.println("Z = " + z);
        z = x % y; System.out.println("Z = " + z);
    }
}
```

Operator Precedence

```
public class StudyJava {
    public static void main(String[] args) {
        int x = 5, y = 3, z;
        z = x + y * 3 - 2;
        System.out.println("Z = " + z);
        z = (x + y) * (3 - 2);
        System.out.println("Z = " + z);
    }
}
```

Operator Precedence



```
a + b + c + d + e
  1   2   3   4
```

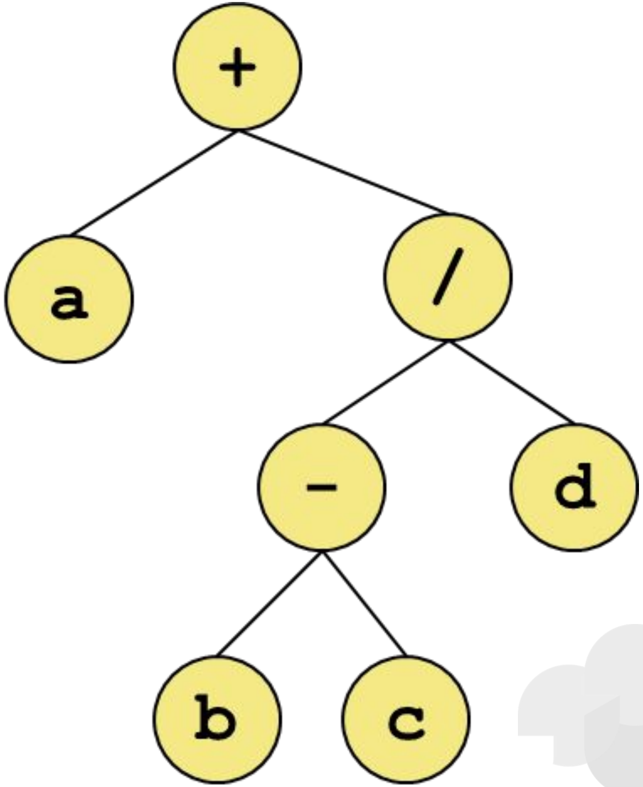
```
a + b * c - d / e
  3   1   4   2
```

```
a / (b + c) - d % e
  2    1    4   3
```

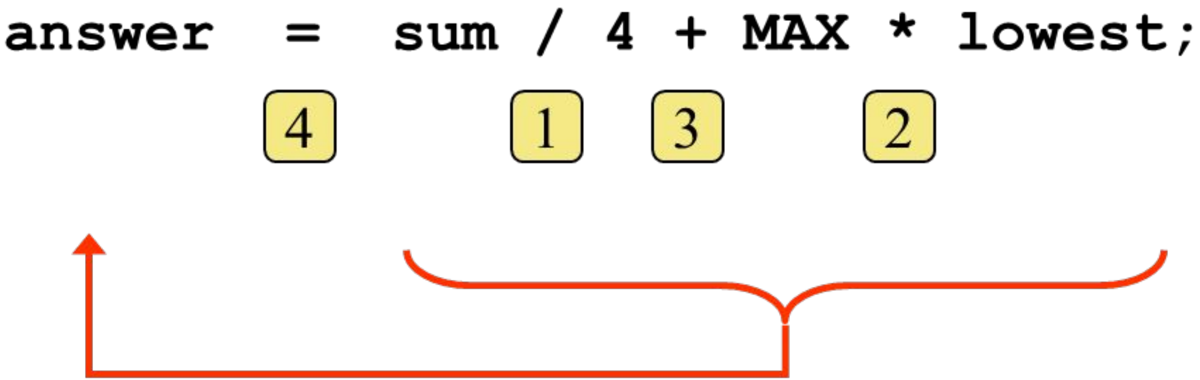
```
a / (b * (c + (d - e)))
  4    3    2    1
```

Expression Tree

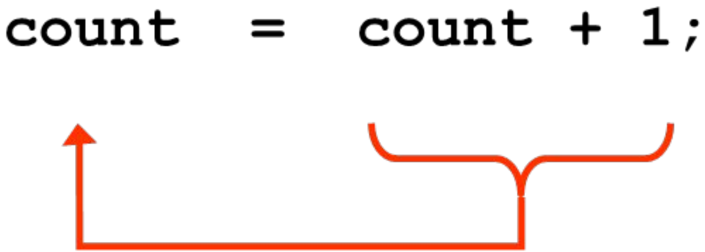
a + (b − c) / d



Assignment



```
answer  =  sum / 4 + MAX * lowest;
```



```
count  =  count + 1;
```

Self Expressions

```
public class StudyJava {
    public static void main(String[] args) {
        int x = 5;
        System.out.println("x = " + x);
        x += 2; // Add 2 to x
        System.out.println("x = " + x);
        x -= 2; // Subtract 2 from x
        System.out.println("x = " + x);
        x *= 2; // Multiply x by 2
        System.out.println("x = " + x);
        x /= 2; // Divide x by 2
        System.out.println("x = " + x);
        x %= 2; // Get remainder of divide by 2
        System.out.println("x = " + x);
    }
}
```

Increment / Decrement

```java
public class StudyJava {
    public static void main(String[] args) {
        int x = 5, y;
        System.out.println("x = " + x);
        x++;        System.out.println("x = " + x);
        x--;        System.out.println("x = " + x);
        y = x++;    System.out.println("y = " + y);
        y = ++x;    System.out.println("x = " + x);
    }
}
```

Formatting Output

```java
public class StudyJava {
    public static void main(String[] args) {
        int a = 60;
        int b = 98989;
        double c = 90.767676;
        System.out.printf("%d, %d, %f\n", a, b, c);
        System.out.printf("%8d, %08d, %.2f\n", a, b, c);
    }
}
```

Data Conversion

```java
public class StudyJava {
    public static void main(String[] args) {
        int x = 6;
        float y = 2.5f; // Use 'f' to indicate float variable
        float z;
        z = x / y;
        System.out.println("z = " + z);
        // z will be 2.4, since Java performs integer division by default
        z = x / (float) y; // Cast y to float to avoid integer division
        System.out.println("z = " + z); // z will now be 2.4
    }
}
```

Native Types Wrapper Class

```java
public class StudyJava {
    public static void main(String[] args) {
        String a = "77";
        int b = 88;
        int c = b + Integer.parseInt(a);
        System.out.println(c);

        String x = "7.7";
        double y = 8;
        double z = y + Double.parseDouble(x);
        System.out.println(z);
    }
}
```

Interactive Programs

```java
import java.util.Scanner;

public class StudyJava {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);

        System.out.print("Enter a string: ");
        String s = scan.nextLine();
        System.out.println("You have entered: " + s);

        System.out.print("Enter an integer: ");
        int i = scan.nextInt();
        System.out.println("You have entered: " + i);

        System.out.print("Enter a double: ");
        double d = scan.nextDouble();
        System.out.println("You have entered: " + d);
    }
}
```

Interactive Programs

```java
import java.util.Scanner;

public class StudyJava {
    public static void main(String[] args) {
        final double PI = 3.14159;
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the circle's radius: ");
        double radius = scanner.nextDouble();
        double area = PI * radius * radius;
        System.out.println("The area of the circle is: " + area);
    }
}
```

Packages

```java
public class StudyJava {
    public static void main(String[] args) {
        java.util.Scanner scanner = new java.util.Scanner(System.in);
        String a = scanner.nextLine();
        System.out.println(a);
        scanner.close();
    }
}
```

Packages

```java
import java.util.Scanner;

public class StudyJava {
    public static void main(String[] args) {
        java.util.Scanner scanner = new Scanner(System.in);
        String a = scanner.nextLine();
        System.out.println(a);
        scanner.close();
    }
}
```

Packages

```
import java.util.*;

public class StudyJava {
    public static void main(String[] args) {
        java.util.Scanner scanner = new Scanner(System.in);
        String a = scanner.nextLine();
        System.out.println(a);
        scanner.close();
    }
}
```

Random Package

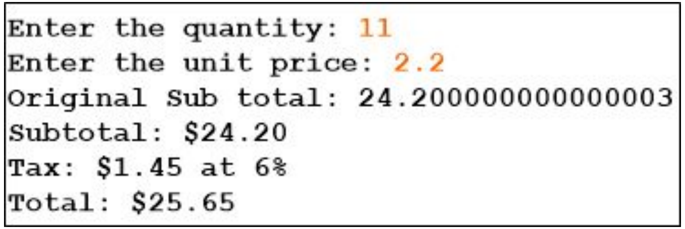
```java
import java.util.Random;
public class StudyJava {
    public static void main(String[] args) {
        Random generator = new Random();
        // Generate a random integer
        int r = generator.nextInt(); System.out.println("A random integer: " + r);
        // Generate a random integer from 0 to 9
        r = generator.nextInt(10); System.out.println("From 0 to 9: " + r);
        // Generate a random integer from 1 to 10
        r = generator.nextInt(10) + 1; System.out.println("From 1 to 10: " + r);
        // Generate a random integer from 20 to 34
        r = generator.nextInt(15) + 20; System.out.println("From 20 to 34: " + r);
        // Generate a random float between 0 and 1
        float f = generator.nextFloat(); System.out.println("A random float (between 0-1): " + f);
        // Generate a random float from 0 to 6
        f = generator.nextFloat() * 6; System.out.println("From 0 to 6: " + f);
    }
}
```

Math Package

```java
import java.util.Scanner;
public class StudyJava {
    public static void main(String[] args) {
        int a, b, c; // ax^2 + bx + c
        double discriminant, root1, root2;
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter the coefficient of x squared: ");
        a = scan.nextInt();
        System.out.print("Enter the coefficient of x: ");
        b = scan.nextInt();
        System.out.print("Enter the constant: ");
        c = scan.nextInt();
        discriminant = Math.pow(b, 2) - (4 * a * c);
        root1 = (-b + Math.sqrt(discriminant)) / (2 * a);
        root2 = (-b - Math.sqrt(discriminant)) / (2 * a);
        System.out.println("Root #1: " + root1);
        System.out.println("Root #2: " + root2);
    }
}
```

Formatter Package

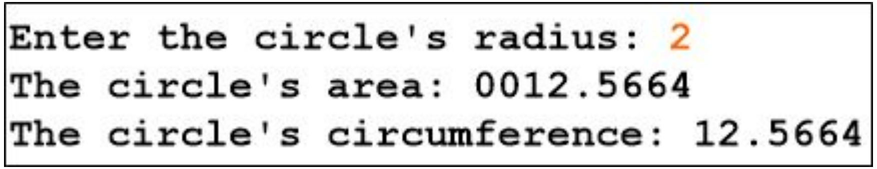
```java
import java.text.NumberFormat;
import java.util.Scanner;
public class StudyJava {
    public static void main(String[] args) {
        final double TAX_RATE = 0.06; // 6% sales tax
        int quantity; double subtotal, tax, totalCost, unitPrice;
        Scanner scan = new Scanner(System.in);

        NumberFormat currencyFormat = NumberFormat.getCurrencyInstance();
        NumberFormat percentFormat = NumberFormat.getPercentInstance();

        System.out.print("Enter the quantity: "); quantity = scan.nextInt();

        System.out.print("Enter the unit price: "); unitPrice = scan.nextDouble();

        subtotal = quantity * unitPrice;
        tax = subtotal * TAX_RATE;
        totalCost = subtotal + tax;

        System.out.println("Original Sub total: " + subtotal);
        System.out.println("Subtotal: " + currencyFormat.format(subtotal));
        System.out.println("Tax: " + currencyFormat.format(tax) + " at " + percentFormat.format(TAX_RATE));
        System.out.println("Total: " + currencyFormat.format(totalCost));
    }
}
```



```
Enter the quantity: 11
Enter the unit price: 2.2
Original Sub total: 24.200000000000003
Subtotal: $24.20
Tax: $1.45 at 6%
Total: $25.65
```

Formatter Package

```java
import java.text.DecimalFormat;
import java.util.Scanner;
public class StudyJava {
    public static void main(String[] args) {
        int radius;
        double area, circumference;

        Scanner scan = new Scanner(System.in);

        System.out.print("Enter the circle's radius: ");
        radius = scan.nextInt();

        area = Math.PI * Math.pow(radius, 2);
        circumference = 2 * Math.PI * radius;

        DecimalFormat decimalFormat = new DecimalFormat("0000.0000");

        System.out.println("The circle's area: " + decimalFormat.format(area));
        System.out.println("The circle's circumference: " + decimalFormat.format(circumference));
    }
}
```



```
Enter the circle's radius: 2
The circle's area: 0012.5664
The circle's circumference: 12.5664
```

Conditions and Loops

Conditions

```
public class StudyJava {
    public static void main(String[] args) {
        int a = 60;
        int b = 90;
        boolean c;
        c = a > b; System.out.println(c);

        c = a > 50 && b >= 90; System.out.println(c);
        c = a != 50 && b > 90; System.out.println(c);

        c = a > 50 || b > 100; System.out.println(c);
        c = a > 100 || b > 100; System.out.println(c);

        c = !(a > 60); System.out.println(c);
    }
}
```

If Statement and Conditions

```java
System.out.print("Enter student grade: ");
double grade = scan.nextDouble();

if (grade >= 90) {
    System.out.println("Excellent");
} else if (grade >= 80) {
    System.out.println("Very Good");
} else if (grade >= 70) {
    System.out.println("Good");
} else if (grade >= 60) {
    System.out.println("Passed");
} else {
    System.out.println("Failed");
}
```

If Statement and Conditions

```java
System.out.print("Enter three numbers: ");
int a = scan.nextInt();
int b = scan.nextInt();
int c = scan.nextInt();
int max;
if (a > b && a > c) {
    max = a;
}
else if (b > a && b > c) {
    max = b;
}
else {
    max  = c;
}
System.out.println("The maximum number is: " + max);
```

If Statement and Conditions

```
if(a>b) {
    if(a>c) {
        max = a;
    }
    else {
        max = c;
    }
}
else {
    if(b>c) {
        max = b;
    }
    else {
        max = c;
    }
}
```

```
int max = (a>b)?a:b;
```

Inline Conditions: Maximum of Two

```
System.out.print ("Enter three values:");
double a = scan.nextDouble();
double b = scan.nextDouble();
double maximum = (a>b) ? a : b;
System.out.print ("Maximum is :" + maximum);
```

Inline Conditions: Maximum of Three

```
System.out.print ("Enter three values:");
double a = scan.nextDouble();
double b = scan.nextDouble();
double c = scan.nextDouble();
double maximum = (a>b) ?((a>c) ?a:c): ((b>c) ?b:c);
System.out.print ("Maximum is :" + maximum);
```

Switch Statement

```
final double PI = 3.143;
System.out.print("Enter circle radius: ");
double radius = scan.nextDouble();
System.out.print("Enter 1 for circumference 2 for area:");
switch (scan.nextInt()) {
    case 1: {
        double C = 2 * PI * radius;
        System.out.println("Circumference = " + C);
    }
    break;
    case 2: {
        double A = PI * radius * radius;
        System.out.println("Area = " + A);
    }
    break;
    default: {
        System.out.println("Wrong choice");
    }
    break;
}
```

For Statement

```
int sum = 0;
for(int i=0; i<=100;i++) {
    sum += i;
}
System.out.println(sum);
```

Nested Statement

```
for(int i=0; i<=10;i++) {
    for(int j=0;j<=i;j++) {
        System.out.print("*");
    }
    System.out.println();
}
```

While Statement

```java
int count = scanner.nextInt();
int i = 0;
double total = 0;
while(i<count) {
    total += scanner.nextDouble();
    i++;
}
System.out.println("Average Temperature: " + (total/count));
```

Endless while and break

```java
int count = 0;
double total = 0;
while(true) {
    System.out.print("Enter degree value or (-ev) to exit:");
    double degree = scanner.nextDouble();
    if(degree<0) break;
    total += degree;
    count++;
}
System.out.println("Average Degree: " + (total/count));
```

Do .. While

```
do {
    System.out.print("Enter degree value:");
    double degree = scanner.nextDouble();
    System.out.print("Enter max degree value:");
    double maxDegree = scanner.nextDouble();
    double percentage = 100 * degree / maxDegree;
    System.out.println("Percentage is: " + Math.round(percentage) + "%");
    System.out.println("Do you want to continue (1:yes, 2:no)?");
} while(scanner.nextInt() == 1);
```

Strings

Strings

- ❑ Format
- ❑ Case
- ❑ Trim
- ❑ Compare
- ❑ Concat

```
String text = "Hello";
System.out.println(text);
int x = 7;
double y = 8;
int v = 199;
text = String.format("x = %08d, y = %8.4f, v = %d", x, y, v);
System.out.println(text);
text = new String("Welcome to Java");
System.out.println(text.toUpperCase());
System.out.println(text.toLowerCase());
text = new String(" Hello ");
System.out.println(text.trim());
text = "Computer";
System.out.println(text.compareTo("computer") == 0);
System.out.println(text.compareToIgnoreCase("computer") == 0);
text = "Computer";
text += " World";
System.out.println(text);
```

Strings

```java
String text = ""; System.out.println(text.isEmpty());
text = "hello"; System.out.println(text.length());
text = "Welcome to Java World. Java is Simple.";
System.out.println(text.substring(11, 15));
System.out.println(text.replace("Java", "Computer"));
System.out.println(text.charAt(1));
System.out.println(text.indexOf('J'));
System.out.println(text.indexOf("Java"));
System.out.println(text.lastIndexOf("Java"));
int index = -1;
while (true) {
    index = text.indexOf("Java", index + 1);
    if (index < 0) break;
    System.out.println("Java is found at:" + index);
}
System.out.println(text.contains("World"));
```

❑ Check
❑ Length
❑ Substring
❑ Replace
❑ Characters
❑ Search

Splitting

```java
import java.util.Scanner;

public class StudyJava {

    public static void main(String[] args) {
        Scanner scan = new Scanner("Welcome to java world");
        while (scan.hasNext()) {
            System.out.println(scan.next());
        }
    }
}
```

Splitting

```java
import java.util.Scanner;
public class StudyJava {
    public static void main(String[] args) {
        Scanner scan = new Scanner("21,39,87,49,82,98,24");
        scan.useDelimiter(",");
        int sum = 0;
        while (scan.hasNext()) {
            sum += scan.nextInt();
        }
        System.out.println(sum);
    }
}
```

Arrays

Array Initialization and Iteration using “for” statement

```
public class StudyJava {
    public static void main(String[] args) {
        int[] primes = {2, 3, 5, 7, 11, 13};
        for(int value:primes) {
            System.out.println(value);
        }
    }
}
```

Using Arrays

```java
Scanner scan = new Scanner(System.in);
System.out.println("Enter 5 values:");
int[] values = new int[5];
for (int i = 0; i < values.length; i++) {
    values[i] = scan.nextInt();
}
System.out.println("You entered: ");
for (int i = 0; i < values.length; i++) {
    System.out.println(values[i]);
}
System.out.println("You entered: ");
for (int value : values) {
    System.out.println(value);
}
```

Program Arguments

```java
public class StudyJava {
    public static void main(String[] args) {
        if (args.length < 3) { System.out.println("Invalid args"); return; }
        if (args[0].compareToIgnoreCase("sum") == 0) {
            double sum = 0;
            for (int i = 1; i < args.length; i++)
                sum += Double.parseDouble(args[i]);
            System.out.println("summation is " + sum);
        } else if (args[0].compareToIgnoreCase("mul") == 0) {
            double product = 1;
            for (int i = 1; i < args.length; i++)
                product *= Double.parseDouble(args[i]);
            System.out.println("product is " + product);
        } else {
            System.out.println("Invalid operation");
        }
    }
}
```

2D Arrays (Matrix)

```java
double[][] X = { { 3.5, 4.1 }, { 1.5, 2.3 }, { 7.2, 8.3 } };
for (int r = 0; r < X.length; r++) {
    for (int c = 0; c < X[r].length; c++) {
        System.out.print(X[r][c] + "\t");
    }
    System.out.println();
}
```

3.5	4.1
1.5	2.3
7.2	8.3

Adding Two Matrices

```java
double[][] X = { { 3.5, 4.1 }, { 1.5, 2.4 } };
double[][] Y = { { 6.3, 3.1 }, { 2.5, 2.3 } };
double[][] Z = new double[2][2];
for (int r = 0; r < X.length; r++) {
    for (int c = 0; c < X[r].length; c++) {
        Z[r][c] = X[r][c] + Y[r][c];
    }
}
for (int r = 0; r < Z.length; r++) {
    for (int c = 0; c < Z[r].length; c++) {
        System.out.print(Z[r][c] + "\t");
    }
    System.out.println();
}
```

$$\begin{array}{|c|c|}\hline 3.5 & 4.1 \\ \hline 1.5 & 2.4 \\ \hline\end{array} + \begin{array}{|c|c|}\hline 6.3 & 3.1 \\ \hline 2.5 & 2.3 \\ \hline\end{array} = \begin{array}{|c|c|}\hline 9.8 & 7.2 \\ \hline 4.0 & 4.7 \\ \hline\end{array}$$

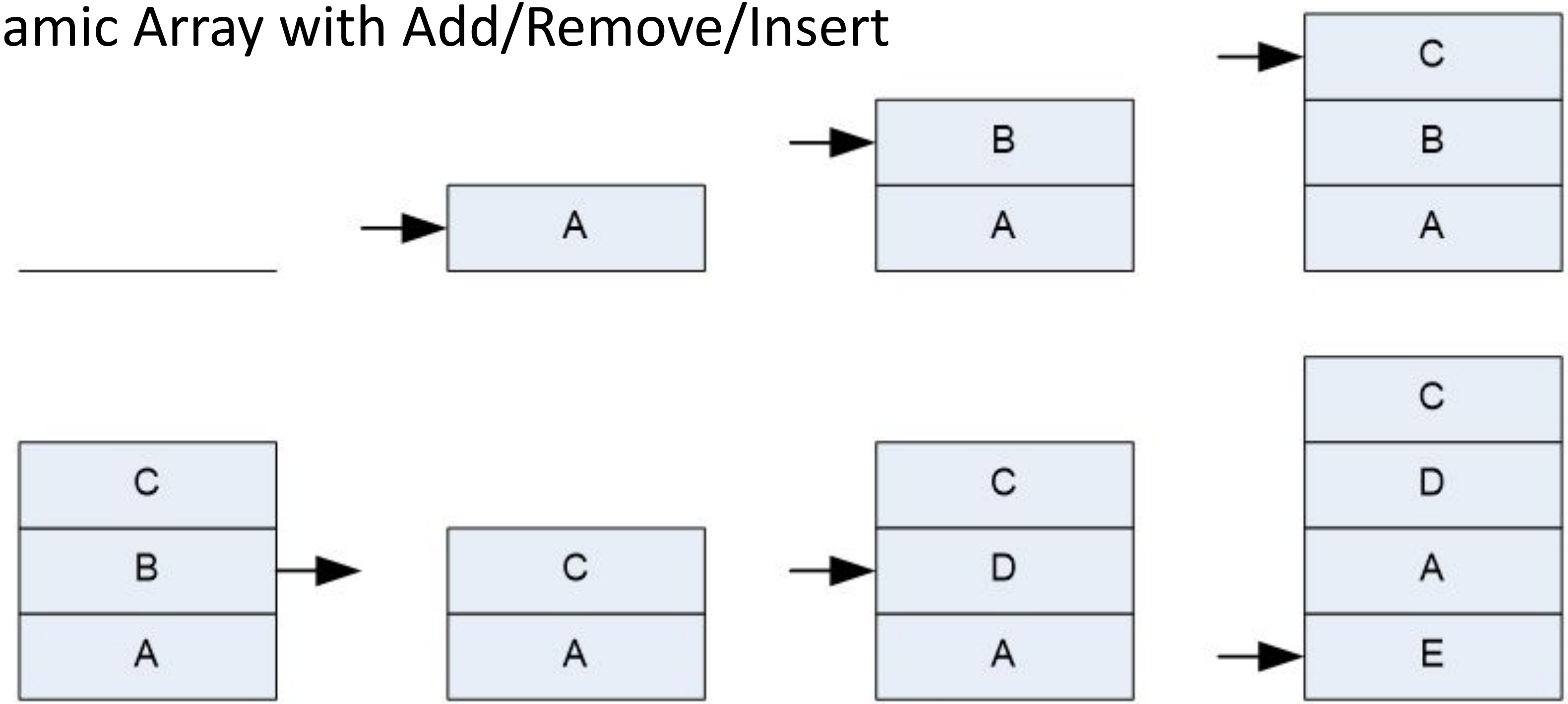
Collections

Collections Types

- Provide dynamic data entry and data removal
  - ArrayList
- Provide special data entry and data removal
  - Stack
  - Queue
- Assign key indexing
  - TreeMap

ArrayList

Dynamic Array with Add/Remove/Insert



ArrayList

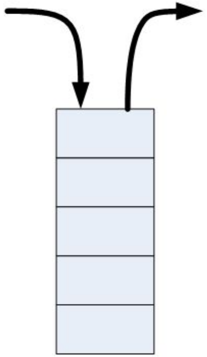
```java
ArrayList<String> myarray = new ArrayList<String>();
myarray.add("Paul");
myarray.add("Pete");
myarray.add("John");
myarray.add("George");
System.out.println(myarray);
myarray.set(1, "Shady");
int location = myarray.indexOf("Pete");
myarray.remove(location);
System.out.println(myarray);
System.out.println("At index 1: " + myarray.get(1));
myarray.add(2, "Ringo");
myarray.add(1, "Mark");
System.out.println(myarray);
System.out.println("Size of the band: " + myarray.size());
```

- □ Add
- □ Print
- □ Find
- □ Remove
- □ Index
- □ Insert

Stack

```
Stack<String> history = new Stack<String>();
history.push("https://www.google.com");
history.push("https://www.google.com/search?q=java");
history.push("https://www.w3schools.com/java/java_intro.asp");
history.push("https://www.w3schools.com/java/java_files.asp");
while (!history.empty()) {
    String url = (String) history.pop();
    System.out.println("Back to " + url);
}
```

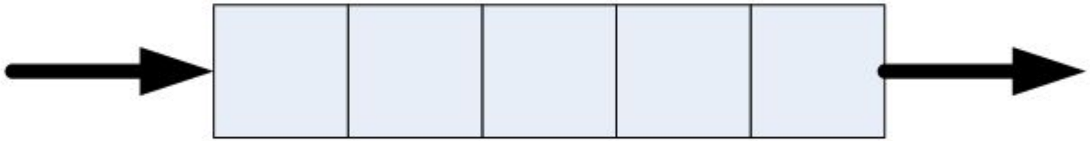
- ❑ Push
- ❑ Print
- ❑ Check
- ❑ POP



Queue

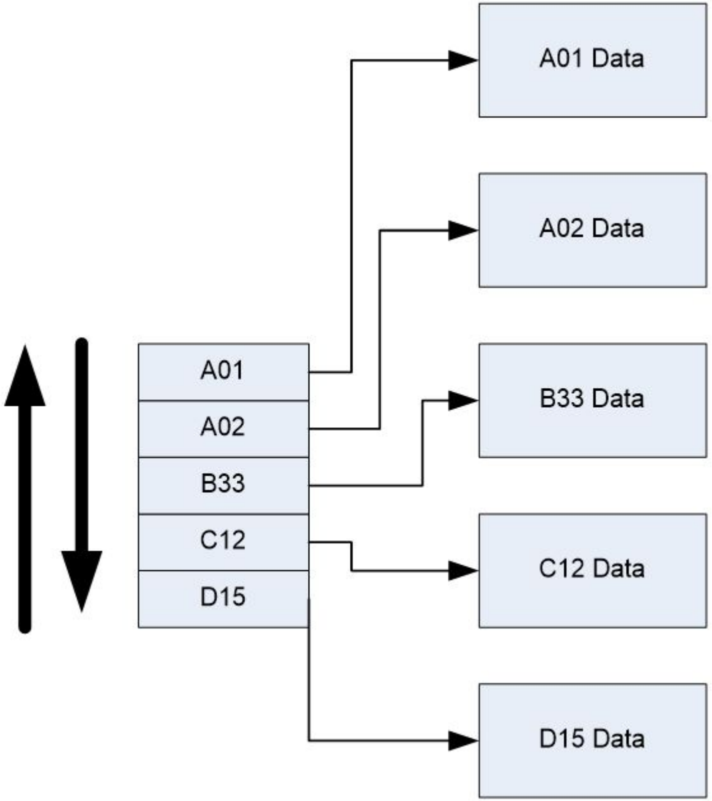
```java
LinkedList<String> queue = new LinkedList<String>();
queue.add("Noura");
queue.add("Salem");
queue.add("Safaa");
queue.add("Sherief");
while (!queue.isEmpty()) {
    String name = queue.poll();
    System.out.println("Serving " + name);
}
```

- ❑ Add
- ❑ Poll
- ❑ Check



TreeMap

- Better way to store data.
- Make data retrieval very fast.
- No need to remember the index.
- Data is retrieved using the given keys.



TreeMap

```
TreeMap<String, String>  citizens = new TreeMap<String, String>();
citizens.put("28503052102715", "Ahmed Ali Said");
citizens.put("28204122102225", "Sara Mohsen Abdelaziz");
citizens.put("27905182102895", "Shady Ahmed Malek");
citizens.put("28502121102125", "Nader Foad Seleem");
citizens.put("29901121101215", "Noura Ahmed Shady");
System.out.print("Enter Citizen Code:");
Scanner scan = new Scanner(System.in);
String code = scan.nextLine();
System.out.println("Citizen name:" + citizens.get(code));
```

❑ Put

❑ Get

Files

Exception Handling

```java
public class StudyJava {
    public static void main(String[] args) {
        try {
            String[] fruits = {"Banana", "Apple", "Orange"};
            System.out.println(fruits[4]);
        }
        catch(Exception e) {
            System.out.println("Oops, there is an error happened.");
        }
    }
}
```

Writing to Text Files

```
try {
    PrintWriter printWriter = new PrintWriter(".\\data.txt");
    int i = 5;
    double d = 5.5;
    String s = "Ahmed";
    printWriter.println(i);
    printWriter.println(d);
    printWriter.println(s);
    printWriter.close();
} catch (FileNotFoundException e) {
    System.out.println("Oops! File error happened.");
}
```

Reading from Text Files

```
try {
    Scanner SR = new Scanner(new File(".\\data.txt"));
    int i = SR.nextInt();
    SR.nextLine();
    double d = SR.nextDouble();
    SR.nextLine();
    String s = SR.nextLine();
    SR.close();
    System.out.printf("%d, %f, %s", i, d, s);
} catch (FileNotFoundException e) {
    System.out.println("Oops! File error happened.");
}
```

Writing to Text Files using Buffered Writer

```
try {
    BufferedWriter writer = new BufferedWriter(new FileWriter(".\\data.txt"));
    writer.write("hello world\n");
    writer.write("hello java\n");
    writer.close();
} catch (IOException e) {
    System.out.println("Oops! File error happened.");
}
```

Reading from Text Files using Buffered Reader

```
try {
    BufferedReader reader = new BufferedReader(new FileReader(".\\data.txt"));
    String a = reader.readLine();
    String b = reader.readLine();
    System.out.println(a);
    System.out.println(b);
    reader.close();
} catch (IOException e) {
    System.out.println("Oops! File error happened.");
}
```

Writing Data to Streams (Binary Files)

```
try {
    int a = 123; double b = 4.35; boolean c = true; String d = "hello";
    float e[] = { 98.7f, 34.2f, 8.9f, 7.1f };
    FileOutputStream fos = new FileOutputStream(".\\serial.dat");
    ObjectOutputStream oos = new ObjectOutputStream(fos);
    oos.writeInt(a);
    oos.writeDouble(b);
    oos.writeBoolean(c);
    oos.writeUTF(d);
    oos.writeObject(e);
    oos.flush();
    oos.close();
} catch (IOException e) {
    System.out.println("Oops! File error happened.");
}
```

Reading Data from Streams (Binary Files)

```java
try {
    int a; double b; boolean c; String d; float e[];
    FileInputStream fis = new FileInputStream(".\\serial.dat");
    ObjectInputStream ois = new ObjectInputStream(fis);
    a = ois.readInt();
    b = ois.readDouble();
    c = ois.readBoolean();
    d = ois.readUTF();
    e = (float[]) ois.readObject();
    ois.close();
    System.out.println(a + " - " + b + " - " + c + " - " + d);
    for (float f : e)
        System.out.print(f + " ");
    System.out.println();
} catch (Exception e) {
    System.out.println("Oops! File error happened.");
}
```

Writing Objects to Streams (Binary Files)

```java
try {
    ArrayList<String> list = new ArrayList<String>();
    list.add("Paul"); list.add("Pete");
    list.add("John"); list.add("George");
    FileOutputStream fos = new FileOutputStream(".\\serial.dat");
    ObjectOutputStream oos = new ObjectOutputStream(fos);
    oos.writeObject(list);
    oos.flush();
    oos.close();
} catch (Exception e) {
    System.out.println("Oops! File error happened.");
}
```

Reading Objects from Streams (Binary Files)

```
try {
    FileInputStream fis = new FileInputStream(".\\serial.dat");
    ObjectInputStream ois = new ObjectInputStream(fis);
    ArrayList<String> list = (ArrayList<String>) ois.readObject();
    ois.close();
    System.out.println(list);
} catch (Exception e) {
    System.out.println("Oops! File error happened.");
}
```