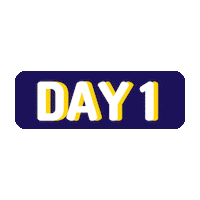
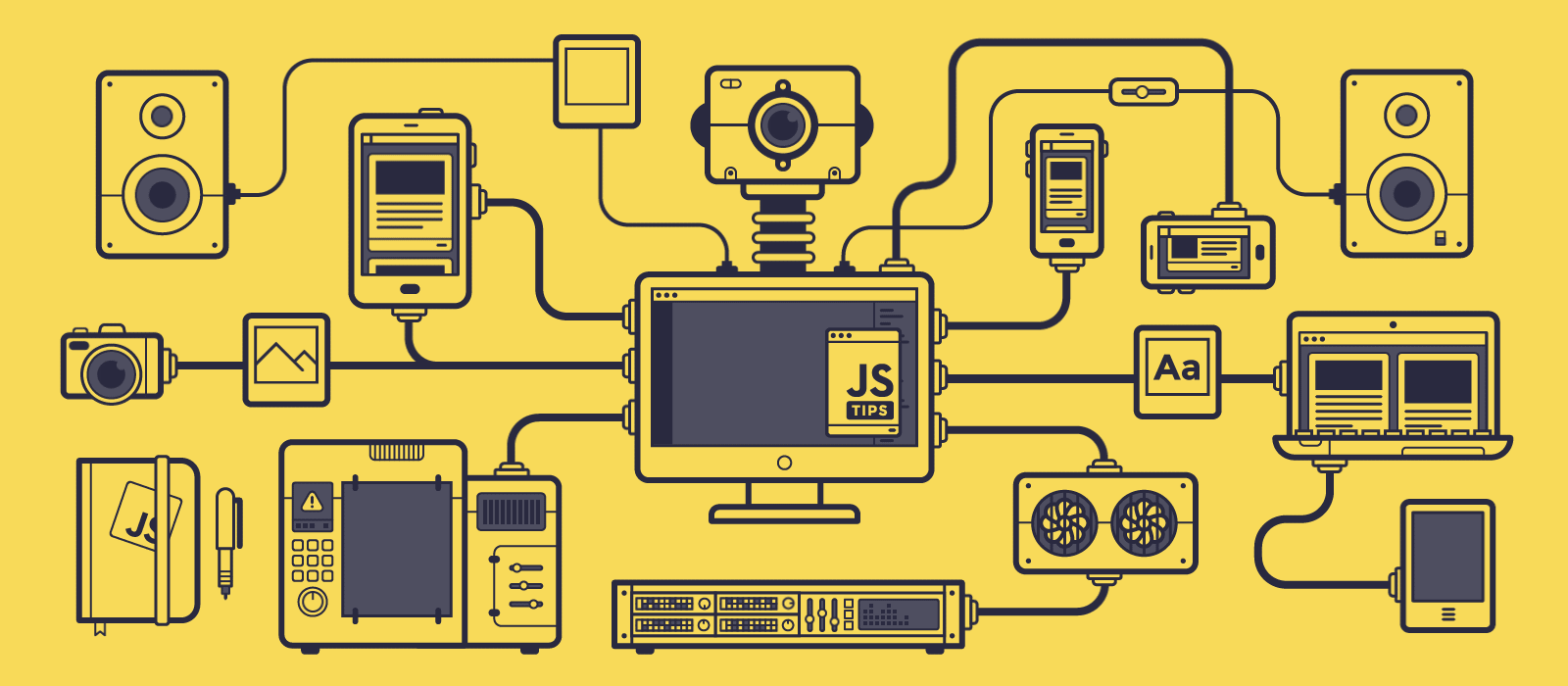
**Introduction to programming language and Operators**

**Notes / Reference material:**

* Video for set up and installation: [Click here](https://www.youtube.com/watch?v=KwnavHTOBiA)
* Interesting links
  + <https://en.wikipedia.org/wiki/The_Imitation_Game>
  + [Coding is not difficult by Bill Gates, Mark Zuckerberg](https://www.youtube.com/watch?v=hb7Q33ysCwI)
  + [Real life examples of algorithms used](https://www.invisibly.com/learn-blog/algorithm-examples-everyday-life)
* **What and why?**

How humans interact using languages, we can interact with computers using its language.

* **About Java**

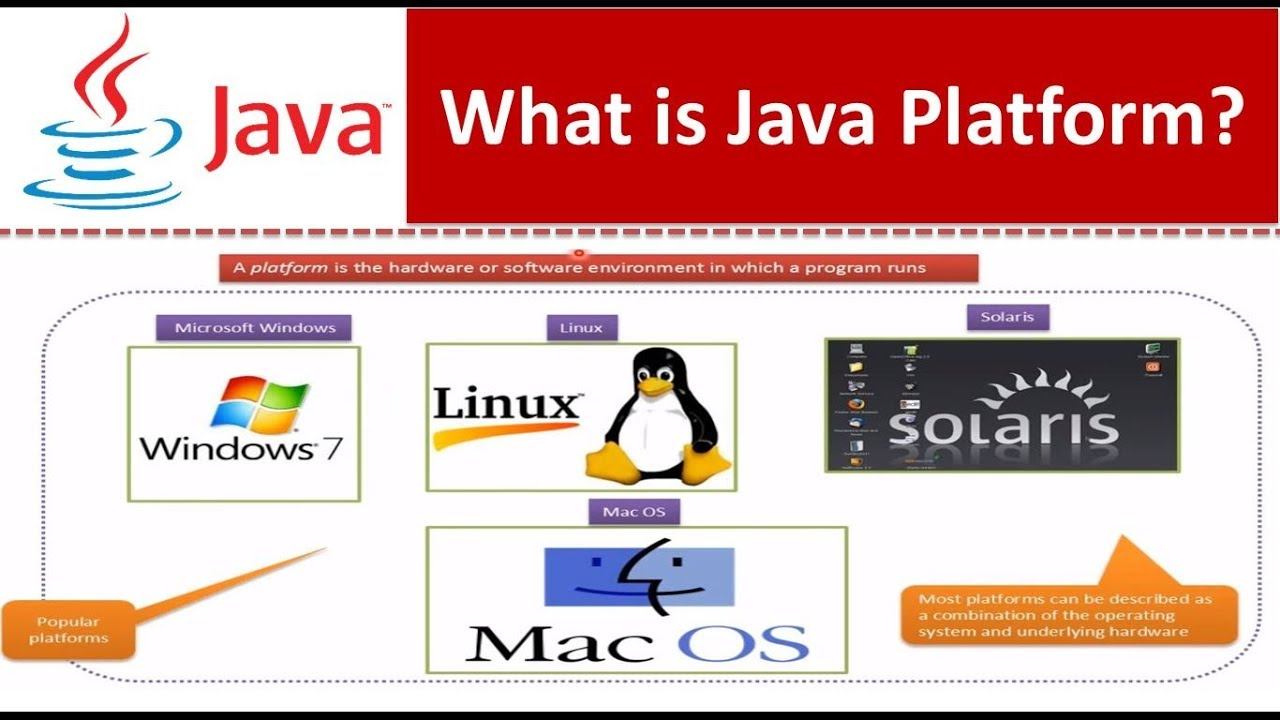
[Click here](https://www.techopedia.com/definition/3927/java#:~:text=Java%20is%20an%20object%2Doriented,Windows%2C%20Linux%20and%20Mac%20OS.)

* **What is Java? **

****

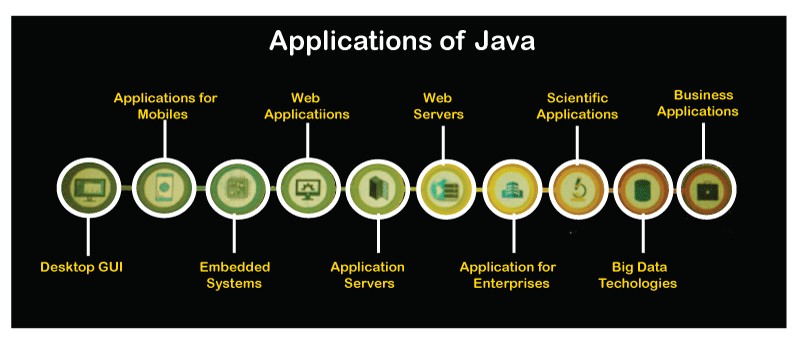
**Java** is an object-oriented, general-purpose programming language built on classes that was intended to have fewer implementation dependencies. It is an application development computing platform. Java is consequently quick, secure, and dependable. It is frequently used to create Java applications for smartphones, game consoles, supercomputers for science, laptops, data centres, etc.

* **What is Java Platform?**

****

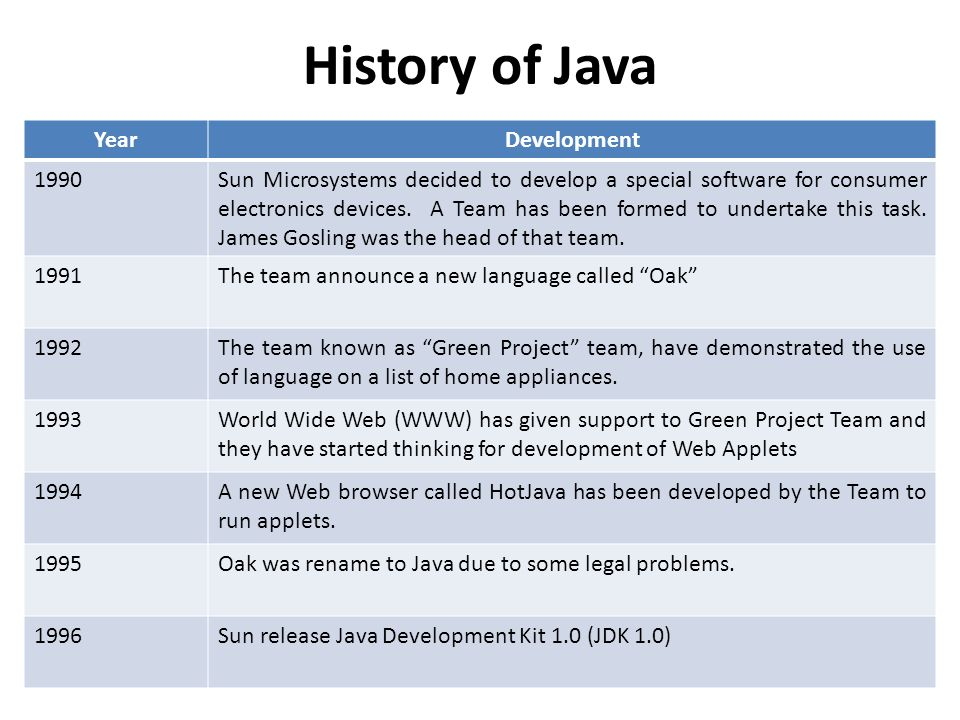
Java Platform is a collection of programs that help programmers to develop and run Java programming applications efficiently. It includes an execution engine, a compiler, and a set of libraries in it. It is a set of computer software and specifications. James Gosling developed the Java platform at Sun Microsystems, and the Oracle Corporation later acquired it.

* **What is Java used for?**

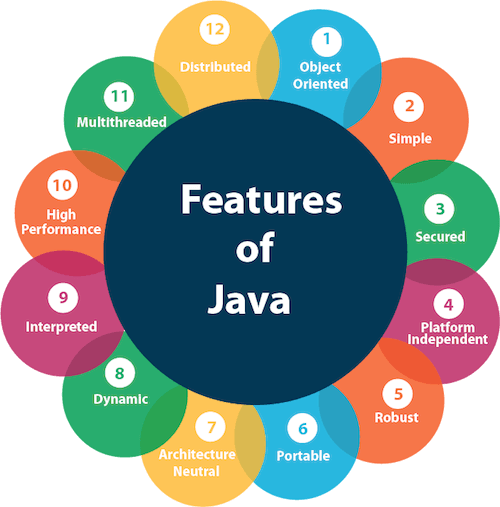
****

Here are some important Java applications:

* It is used for developing Android Apps ([Example of calculator app)](https://github.com/anshulkhattar/Beginner-Level-Android-Studio-Apps/blob/master/Calculator/app/src/main/java/com/justudin/calculator/MainActivity.java)
* Helps you to create Enterprise Software
* Wide range of Mobile java Applications
* Scientific Computing Applications
* Use for Big Data Analytics
* Java Programming of Hardware devices
* Used for Server-Side Technologies like Apache, JBoss, GlassFish, etc.
* **History of Java Programming Language**

****

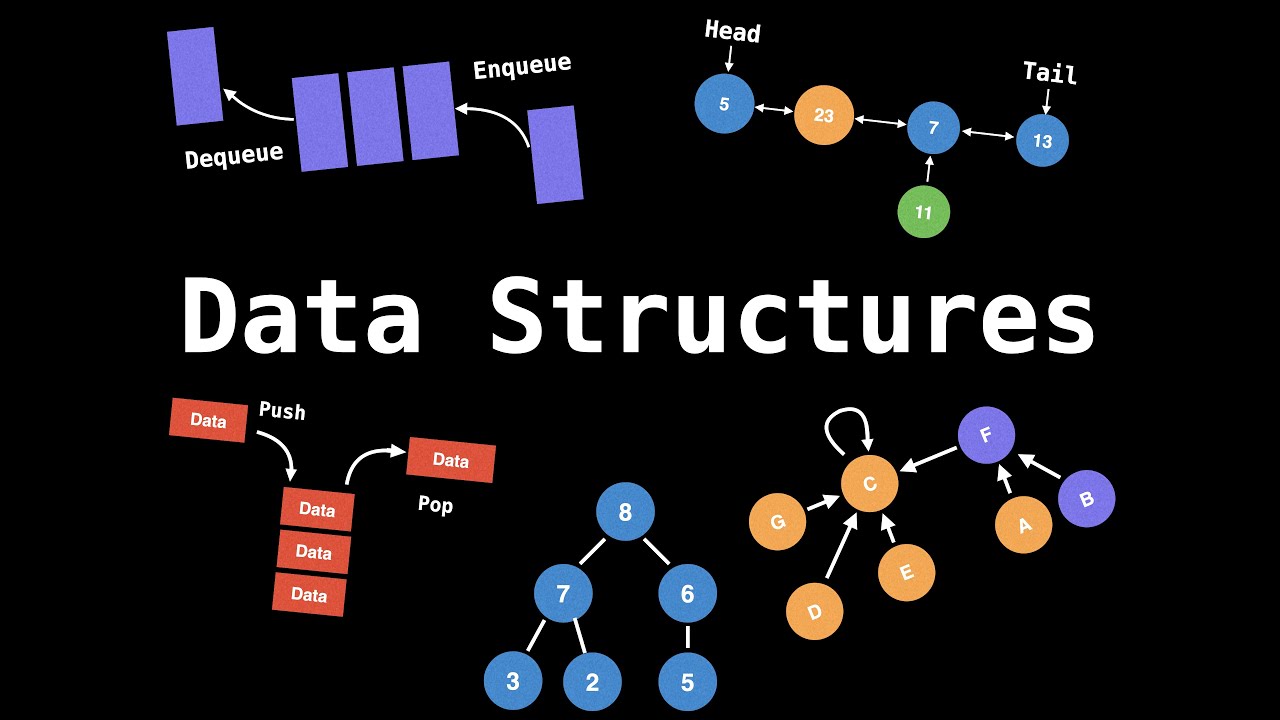
* Here are important landmarks from the history of the Java language:
* The Java language was initially called OAK.
* Originally, it was developed for handling portable devices and set-top boxes. Oak was a massive failure.
* In 1995, Sun changed the name to “Java” and modified the language to take advantage of the burgeoning www (World Wide Web) development business.
* Later, in 2009, Oracle Corporation acquired Sun Microsystems and took ownership of three key Sun software assets: Java, MySQL, and Solaris.
* **Java Features**

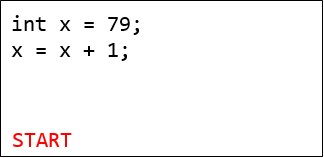
****

* Here are some important Java features:
* It is one of the easy-to-use programming languages to learn.
* Write code once and run it on almost any computing platform.
* Java is platform-independent. Some programs developed in one machine can be executed in another machine.
* It is designed for building object-oriented applications.
* It is a multithreaded language with automatic memory management.
* **What is data?**



Data can be defined as a representation of facts, concepts, or instructions in a formalised manner, which should be suitable for communication, interpretation, or processing by human or electronic machine.

* **Why do we need data structures?**
  + Data Structure is used for organising the data in memory. There are various ways of organising the data in the memory for eg. array, list, stack, queue and many more.
  + The data structure isn’t a programming language like C, C++, Java, etc. It is a set of algorithms that can be used in any programming language to organise the data in the memory.
  + As applications are becoming more complex and the amount of data is increasing day by day, which may cause problems with processing speed, searching data, handling multiple requests etc.
  + Data structure provides a way of organising, managing, and storing data efficiently. With the help of data structure, the data items can be traversed easily.
  + Data structure provides efficiency, reusability and abstraction. It plays an important role in enhancing the performance of a program because the main function of the program is to store and retrieve the user’s data as fast as possible.
* **What is an algorithm?**
* In computer programming terms, an algorithm is a set of well-defined instructions to solve a particular problem. It takes a set of input and produces a desired output. For example,
* An algorithm to add two numbers:
  + Take two number inputs
  + Add numbers using the + operator
  + Display the result.
* **Variables**

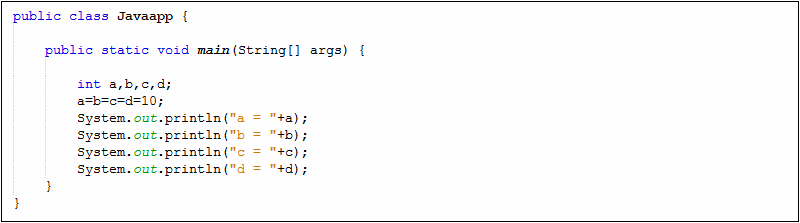
****

Variables are containers for storing data values.

Note - Variables will be more discussed in the upcoming classes too.

In Java, there are different types of variables, for example:

* String - stores text, such as "Hello". String values are surrounded by double quotes
* 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
* **Operators**



| + | 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 |

* **Other operators**

| = | x = 5 | x = 5 |
| --- | --- | --- |
| += | x += 3 | x = x + 3 |
| -= | x -= 3 | x = x - 3 |
| \*= | x \*= 3 | x = x \* 3 |
| /= | x /= 3 | x = x / 3 |
| %= | x %= 3 | x = x % 3 |

| * **Note-**   In Java, we have the following functions to print anything in the console. **System.out.print()** and **System.out.println()** But there is a slight difference between both of them, i.e.  **System.out.print()** only prints the content without switching to the next line after executing this statement whereas **System.out.println()** prints the content and switches to the next line after execution of the statement. |
| --- |
|  |

**Questions Covered In The Class:**

1. **Print “Hello World. I am here.”**

| **Step 1. Start  Step 2. Print/Display ("Hello World. I am here.") Step 3. Stop** |
| --- |

1. **Print the below pattern**

**Hello**

**World.**

**I**

**am**

**here.**

| **Step 1. Start  Step 2. Print/Display ("Hello")**  **Step 3. Print/Display ("World.")**  **Step 4. Print/Display ("I")**  **Step 5. Print/Display ("am")**  **Step 6. Print/Display ("here") Step 7. Stop** |
| --- |

1. **Print \*\*\*\*\***

| **Step 1. Start  Step 2. Print/Display ("\*\*\*\*\*") Step 3. Stop** |
| --- |

1. **Print the below pattern**

**\*\*\*\*\***

**\*\*\*\*\***

**\*\*\*\*\***

| **Step 1. Start  Step 2. Print/Display ("\*\*\*\*\*")**  **Step 3. Print/Display ("\*\*\*\*\*")**  **Step 4. Print/Display ("\*\*\*\*\*") Step 5. Stop** |
| --- |

1. **Print the below pattern**

**\*\*\*\*\***

**\***

**\***

**\***

**\*\*\*\*\***

| **Step 1. Start  Step 2. Print/Display ("\*\*\*\*\*")**  **Step 3. Print/Display ("\*")**  **Step 4. Print/Display ("\*")**  **Step 5. Print/Display ("\*")**  **Step 6. Print/Display ("\*\*\*\*\*") Step 7. Stop** |
| --- |

1. **Add two numbers 10,20.**



| **Step 1. Start  Step 2. Print/Display (10 + 20). Step 3. Stop** |
| --- |

**-Multiply three numbers 10,20,30.**

| **Step 1. Start  Step 2. Print/Display (10 \* 20 \* 30). Step 3. Stop** |
| --- |

**-Subtract two numbers 40-20.**

| **Step 1. Start  Step 2. Print/Display (40 - 20). Step 3. Stop** |
| --- |

1. **Find the sum and product of 20, 30, 50.**

| **Step 1. Start  Step 2. Print/Display (20 + 30 + 50).**  **Step 3. Print/Display (20 \* 30 \* 50). Step 4. Stop** |
| --- |

1. **Divide two numbers 25/10.**

| **Step 1. Start  Step 2. Print/Display (25 / 10). Step 3. Stop** |
| --- |

1. **Find the remainder when 438 is divided by 9.**

| **Step 1. Start  Step 2.Print/Display (438 % 9). Step 3. Stop** |
| --- |

1. **Find the remainder when 4596 is divided by 10.**

| **Step 1. Start  Step 2.Print/Display (4596 % 10). Step 3. Stop** |
| --- |

1. **Store the value 100 in x variable of int data type and then print x.**

| **Step 1. Start  Step 2. Declare Variable names x with ‘int data type’. Step 3. Assign Values of x as 100. Step 4. Print/Display x. Step 5. Stop** |
| --- |

1. **Store the value 235 in y variable of int data type and then print y+10**

| **Step 1. Start  Step 2. Declare Variable names y with ‘int data type’. Step 3. Assign Values of y as 235. Step 4. Print/Display (y + 10). Step 5. Stop** |
| --- |

1. **Store values 165, 84 in x and y variable of int data type and print x+y, x-y, x\*y, x/y.**

| **Step 1. Start  Step 2. Declare Variable names x and y with ‘int data type’. Step 3. Assign Values of x , y as 165 and 84 respectively. Step 4. Print/Display (x+y).**  **Step 5. Print/Display (x-y).**  **Step 6. Print/Display (x\*y).**  **Step 7. Print/Display (x/y). Step 8. Stop** |
| --- |

1. **Store values 20, 30, 40 in the variable x, y, z each of int data type and print x+y-z.**

| **Step 1. Start  Step 2. Declare Variable names x , y , z with ‘int data type’. Step 3. Assign Values of x,y,z as 20 , 30 , 40 respectively. Step 4. Print/Display (x+y+z). Step 5. Stop** |
| --- |

1. **Remove the last digit of a given number. Say 45983**

| **Step 1. Start  Step 2. Print/Display (45983 / 10). Step 3. Stop** |
| --- |

1. **Find the last digit of a given number. Say 45983**

| **Step 1. Start  Step 2. Print/Display (45983 % 10). Step 3. Stop** |
| --- |

**Homework Questions/Assignment Questions:**

1. **Print “Hello I am enjoying coding”**
2. **Print the below pattern:**

**Hello**

**I**

**am**

**enjoying**

**coding**

1. **Find the sum, product of 100,8,3**
2. **Find the difference between 100 and 40.**
3. **Print the below pattern**

**\*\*\*\*\***

**\*\*\*\*\***

**\*\*\*\*\***

**\*\*\*\*\***

**\*\*\*\*\***

**\*\*\*\*\***

1. **Print the below pattern**

**\* \***

**\* \***

**\* \***

**\* \***

**\* \***

**\*\*\*\***

1. **Find the last digit of the number 987653.**
2. **Store values 15, 4 in m and n variable of int data type then**

**Print value of m+n in the first line,**

**Print value of m-n in the second line,**

**Print value of m\*n in the second line.**