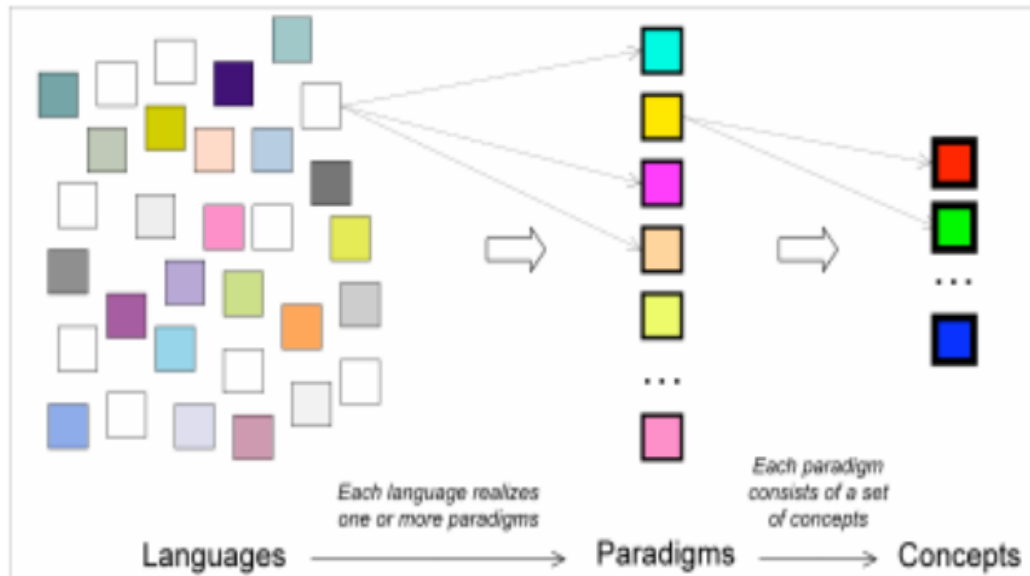


## Introduction

**Programming** is a creative process (عملية ابداعية) carried out (يقوم) by programmers to instruct a computer on how to do a task.

**A program** is a set of instructions that tells a computer what to do in order to come up with (التوصل) a solution to a particular problem.

The approaches to the programming process referred to as **programming paradigms** (نماذج).



Two of the most important approaches are :

1. procedural programming(also referred to as imperative programming or Procedure Oriented Programming (POP))
2. Object-Oriented Programming (or OOP).

The different between them shown in the following table:

	Procedure Oriented Programming	Object Oriented Programming
<b>Divided Into</b>	Program is divided into small parts called <b>functions</b> .	Program is divided into parts called <b>objects</b> .
<b>Importance</b>	Importance is not given to data but to <b>functions</b>	Importance is given to the <b>data</b> rather than functions(methods)
<b>Approach</b>	Follows <b>Top Down approach</b> .	Follows <b>Bottom Up approach</b> .
<b>Access Specifiers</b>	does not have any access specifier.	has access specifiers named Public, Private, Protected, etc.
<b>Data Moving</b>	data can move freely from function to function	objects can move and communicate with each other through member functions.
<b>Expansion</b>	To add new data and	provides an easy way to add new

	function in POP is not so easy.	data and function.
<b>Data Access</b>	most function uses global data for sharing that can be accessed freely from function to function.	data can not move easily from function to function, it can be kept public or private so we can control the access of data.
<b>Data Hiding</b>	does not have any proper way for hiding data so it is <b>less secure</b> .	provides data hiding so provides <b>more security</b> .
<b>Overloading</b>	overloading is not possible.	overloading is possible
<b>Examples</b>	Example of POP are : C, VB, FORTRAN, Pascal.	Example of OOP are : C++, JAVA, VB.NET, C#.NET.

### Fundamental Principles of OOP

In order for a programming language to be **object-oriented**, it has to enable working with **classes** and **objects** as well as the implementation and use of the fundamental object-oriented principles and concepts: inheritance, abstraction, encapsulation and polymorphism:

- **Encapsulation** Is a process of wrapping(التفاف) code and data together into a single unit
- **Inheritance** is the mechanism of making(اتخاذ) new classes from existing one.
- **Abstraction** is a process of hiding the implementation details (تفاصيل التنفيذ) and showing only functionality(وظائف) to the user.
- **Polymorphism** is a concept by which we can perform a single action (عمل) by different ways.

### Classes and Objects

The two most important concepts in object-oriented programming are the class and the object.

An object is comprised (يتكون) of data and operations that manipulate these data. For example, a Student object may consist of data such as name, gender, birth date, home address, phone number, and age and operations for assigning and changing these data values.

For the computer to be able to create an object, we must provide a definition, called a **class**. A class is a kind of template that dictates(يملئ) what objects can and cannot do. An object is called an **instance** of a class.

Once a class is defined, we can create as many instances of the class as a **program requires**. A class must be defined before you can create an instance (object) of the class.

### method and function

A **function** is a combination of instructions that are combined to achieve some results, it requires some input (called arguments) and returns some results.

A **method** in object-oriented programming is a procedure associated with a class, it defines the behavior of the objects(or an action that an object is able to perform). A method defined for a class is called a **class method**.

Q: How does function differ from a method?

Ans.: A function is independent and not associated with a class. You can use this function anywhere in your code, and you don't need to have an object to use it.

### **Java – Basics**

Java programming language was developed by Sun Microsystems which was initiated by James Gosling in 1995. Java is **object oriented where** everything is an object.