GIA DINH
UNIVERSITY

Chapter 2

Introduction to OOP

Objectives

- *Feature of Java*
- *Explain the Java Virtual Machine (JVM)*
- *Introduce some IDEs*
- *Class and object in Java*

Type of Programming

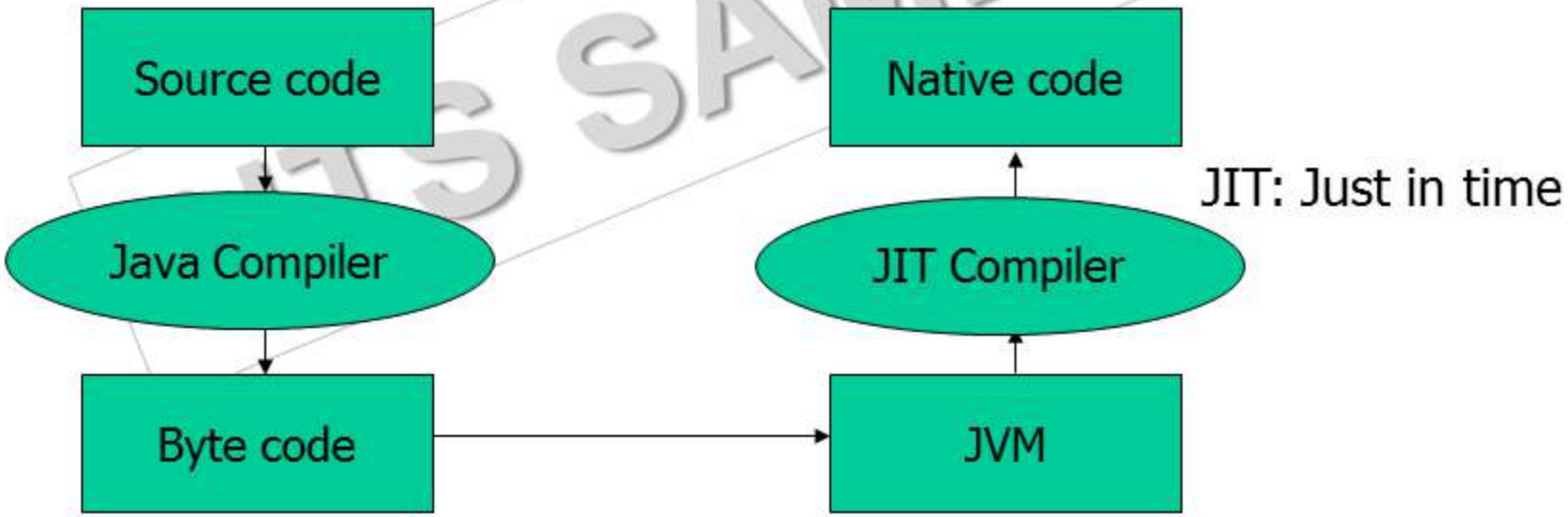
- Non-structured programming: Assembly, Basic
- Structured programming: Pascal, C
- Logic programming: Prolog
- Object-Oriented programming: C++, Java, C#...

Feature of Java

- Object - Oriented
- Platform independent
- Architecture-neutral
- Secure
- Distributed
- Multithreaded
- Dynamic

JVM-Java Virtual Machine

- Java code can run on any platform by using JVM (JRE – Java Runtime Environment)
- Recognizes only a particular binary format called a class file



JDK – Java Development Kit

- JDK: Development tools
  - Java compiler: javac
  - Java interpreter: java
  - Applet viewer: appletviewer
  - Runtime Environment (JRE)

Visual Development Tools

- Eclipse
- NetBeans
- IntelliJ IDEA
- JDeveloper
- MyEclipse
- BlueJ
- Visual Studio Code

Object

- **Object**: the representation of real world entities. Ex: car, house, dog
- **Property**: describes the characteristics of object
- Ex: House → property:
  - Room
  - Door
  - Color
- Ex: Dog → property
  - Breed
  - Color
  - Age

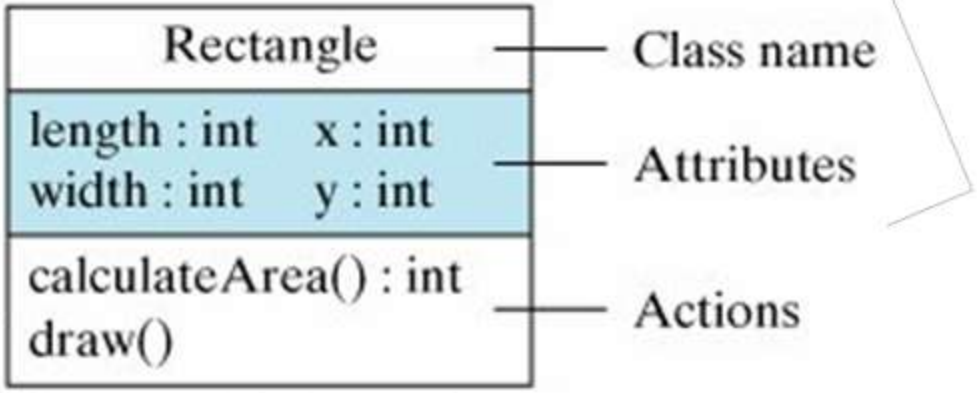
Class

- **Class**: a template defines the outline of state for all objects belonging to that class
- Ex: class: animal → object: dog, cat, tiger, sheep, monkey…
- **Instance**: a dog is just one instance of class animal
- **Attribute** and **action** of class

Comparison

Class	Object
Animal	Panther
	Donkey
Car	Mec
	Toyota

Unified Modeling Language



A UML diagram

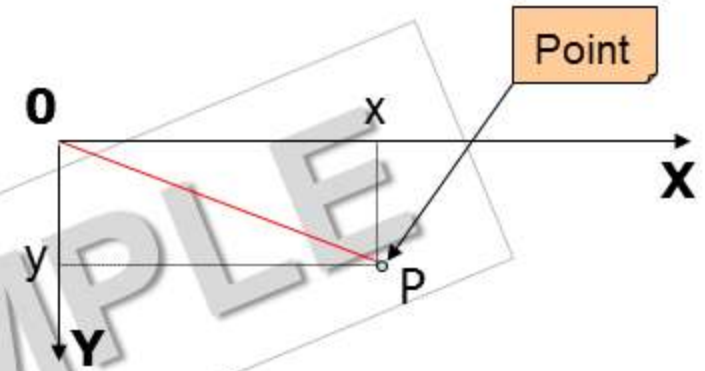
Class in JAVA

- Syntax:

```
class class_name {
        ...
}
```

- Ex:

```
class   CartesianPoint   {
        ...
}
```



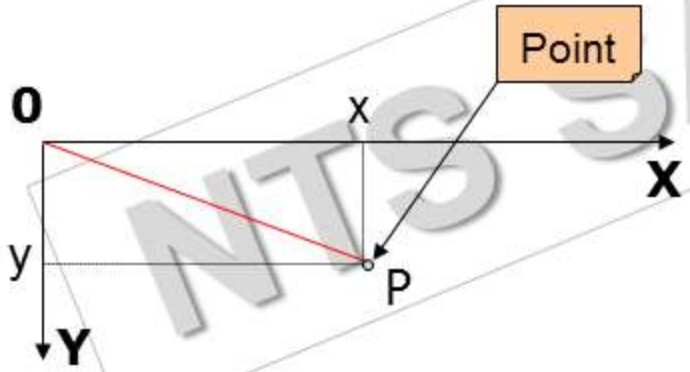
CartesianPoint
x: int y: int
distanceToO(): double

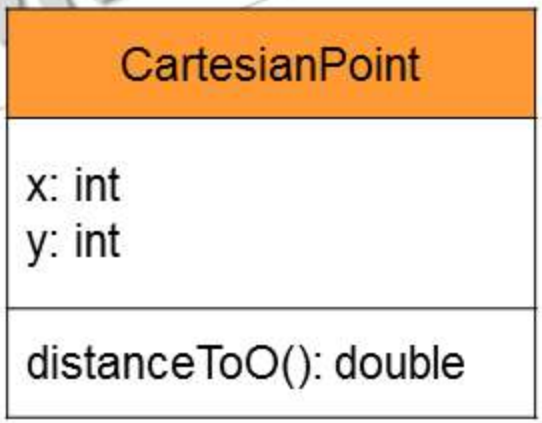
Object in Java

- Syntax

```
class_name1 object_name=new class_name2(arg1,arg2,…);
```

- Ex: CartesianPoint p = new CartesianPoint(4,3);





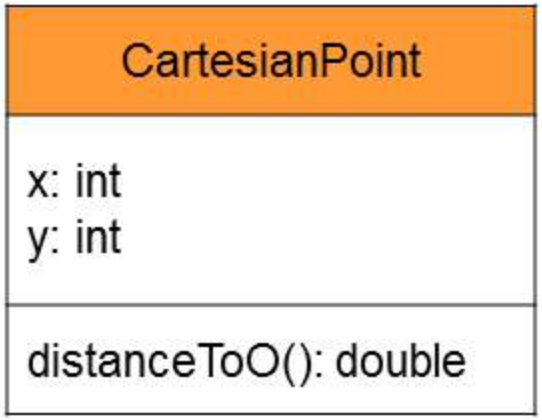
Constructor

- Constructor?
- Syntax:

```
class_name(list_of_arguments){
        ….
}
```

- Ex:

```
CartesianPoint(int x, int y) {
        this.x = x;
        this.y = y;
}
```

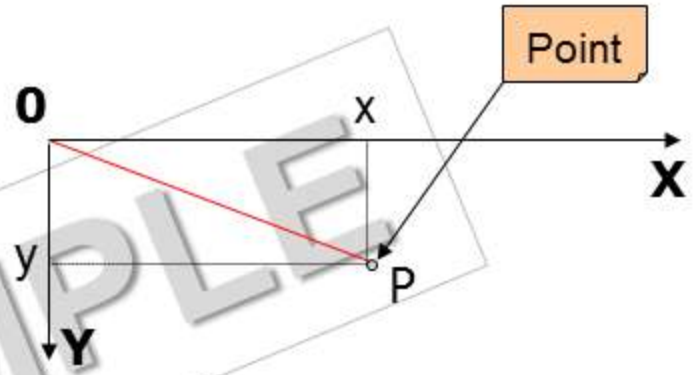
Method in Java

- Syntax:

```
Type method_name(arg1, arg2,…){
}
```

- Ex:

```
double distanceToO() {
        return (float)Math.sqrt(x*x + y*y);
}
```



CartesianPoint
x: int y: int
distanceToO(): double