



BITS Pilani
Pilani Campus

Object Oriented Programming CS F213

Dr. Amitesh Singh Rajput
Dr. Amit Dua



OOP Basics

BITS Pilani
Pilani Campus

My first program



```
class First{  
    public static void main(String args[]){  
        System.out.println("Hello World");  
    }  
}
```

Save the file as **First.java**

Parameters used

- **class** keyword is used to declare a class in java.
- **public** keyword is an access modifier which represents visibility, making it visible to all.
- **static** is a keyword. The core advantage of static method is that there is no need to create object to invoke the static method.
- **void** is the return type of the method.
- **main** represents starting point of the program, invoked by JVM.
- **String[] args** is used for command line argument.
- **System.out.println()** is used to print a string.

What is `System.out.println`?

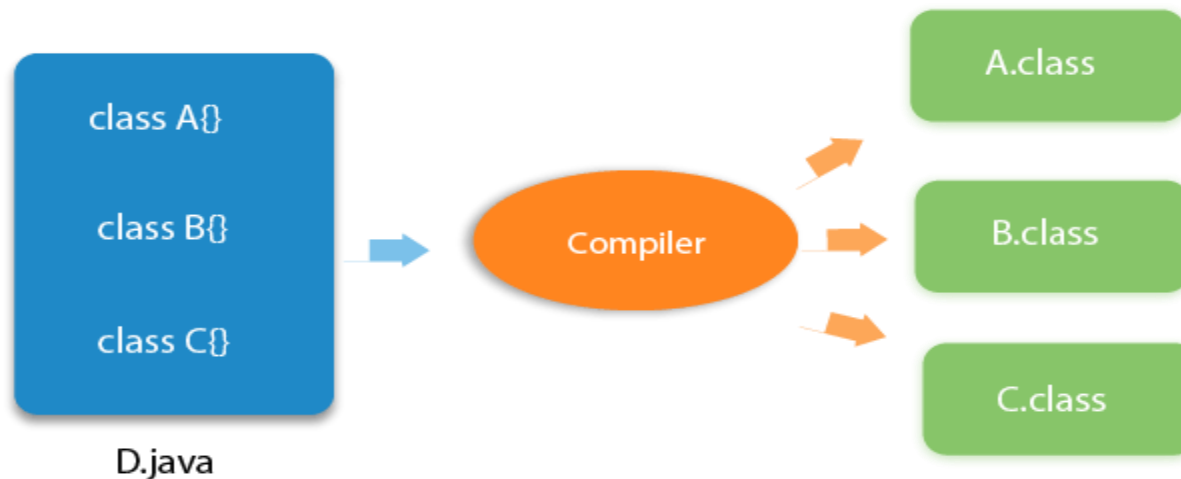
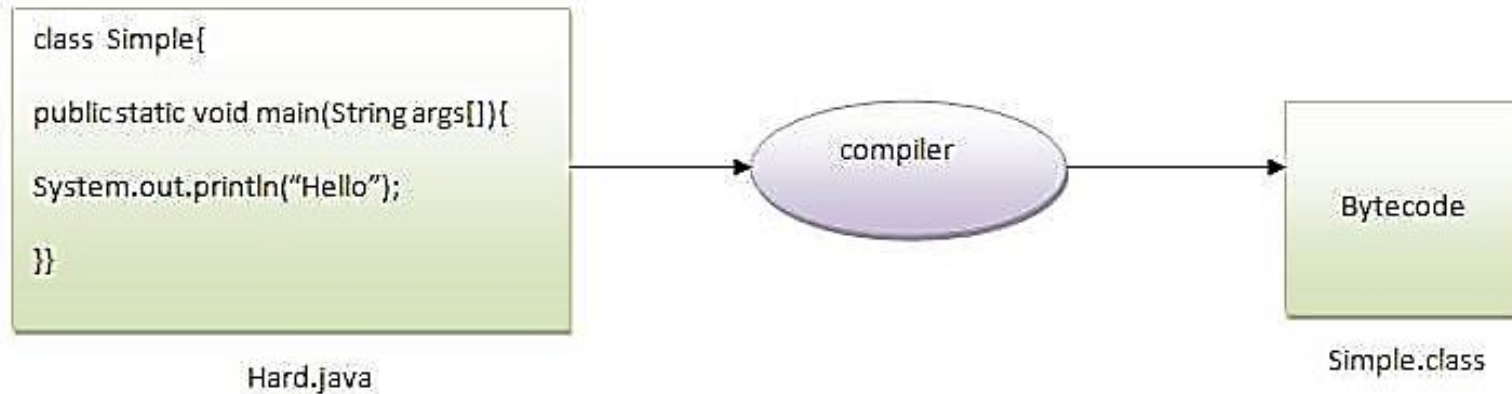


- **System** is a Class in java.lang package (standard input/output and error output streams).
- **out** is a static member field of System class and is of type `PrintStream`.
- **println()** is a method of `PrintStream` class.

```
class System {  
    public static final PrintStream out;  
    ...  
}
```

```
class PrintStream{  
    public void println();  
    ...  
}
```

Additional details

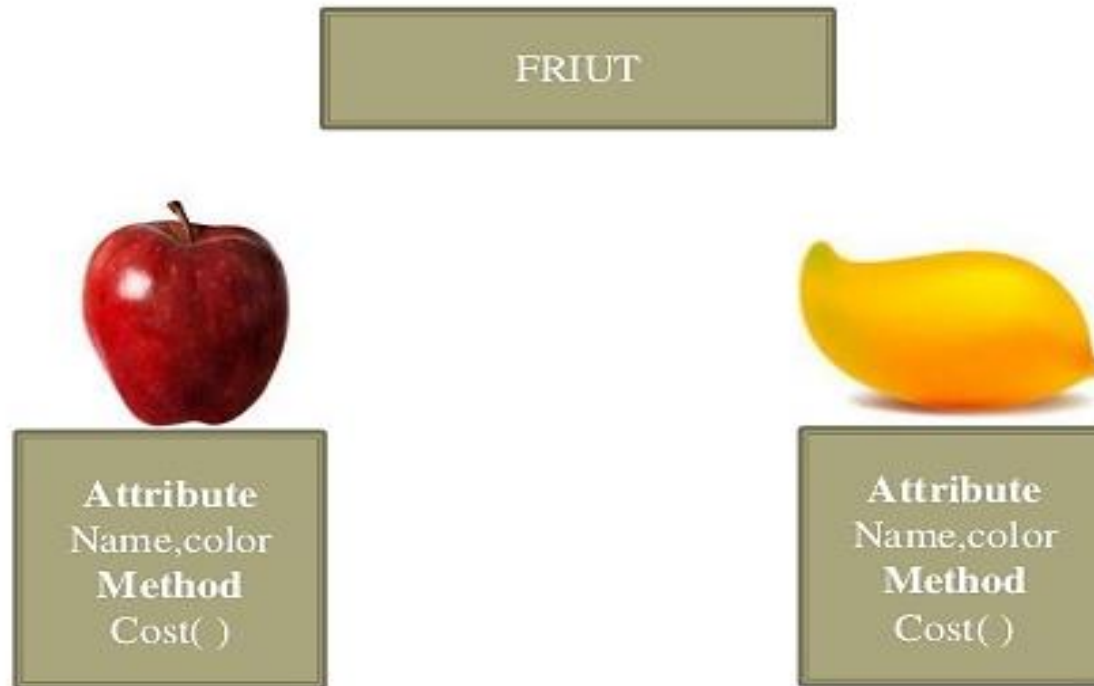


Basic OOP concepts



- Class
- Object
- Encapsulation
- Inheritance
- Polymorphism

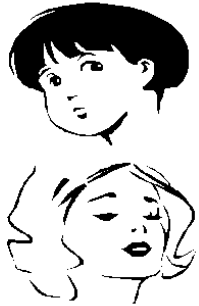
Class



ADT: Examples



Person Objects



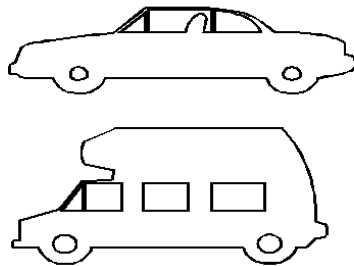
**Abstract
Into** →

Person Class

Attributes: Name, Age, Sex

Operations: Speak(), Listen(), Walk()

Vehicle Objects



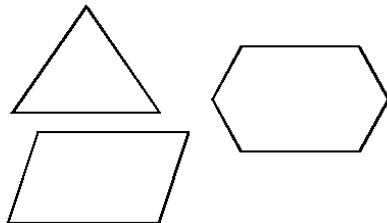
**Abstract
Into** →

Vehicle Class

Attributes: Name, Model, Color

Operations: Start(), Stop(), Accelerate()

Polygon Objects



**Abstract
Into** →

Polygon Class

Attributes: Vertices, Border,
Color, FillColor

Operations: Draw(), Erase(), Move()

Class

- Class is a set of **attributes** and **operations** that are performed on that attributes.
- It's a blueprint from which individual objects can be created.

Account
accountName accountBalance
withdraw() deposit() determineBalance()

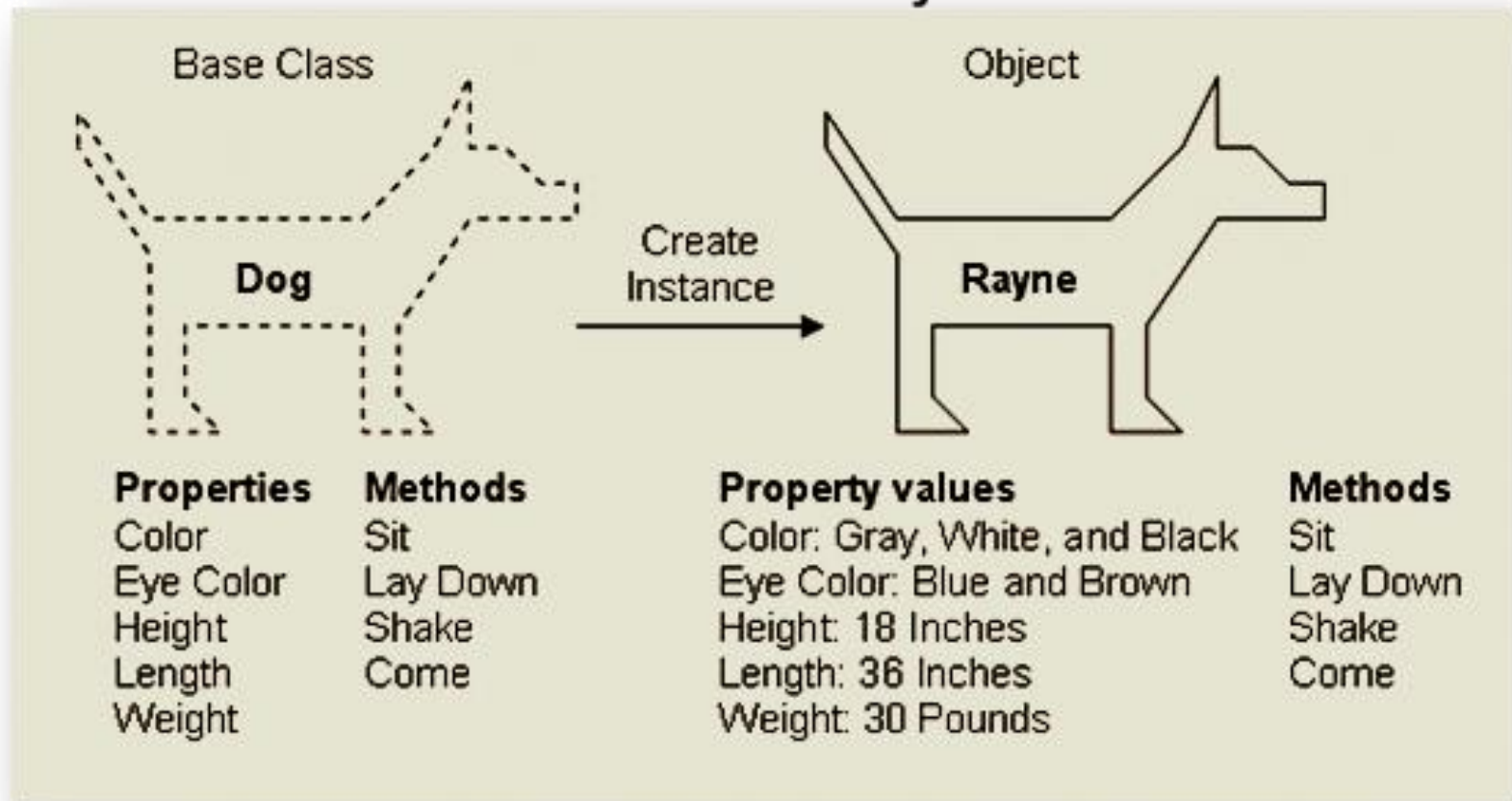
Student
name age studentId
getName() getId()

Circle
centre radius
area() circumference()

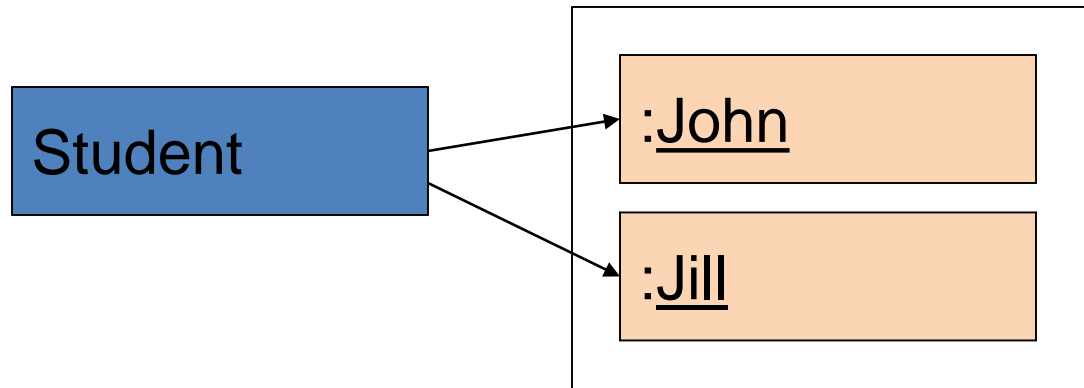
Objects

- Instance of the class.
- Entity that has state and behavior.
- Each object has an address and takes up memory.
- It can communicate without knowing other object's code or data.

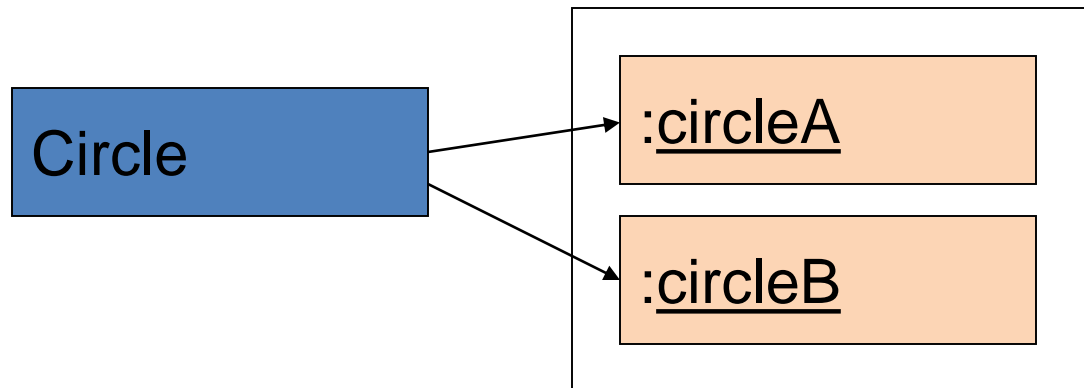
Classes vs. Objects



Classes/Objects



John and Jill are
objects of
Student class



circleA and circleB are
objects of Circle class

Class/Object Example

```
class Student{  
    int id;  
    String name;  
}
```

```
class TestStudent{  
    public static void main(String args[]){  
        Student s1 = new Student();        //Creating object  
        s1.id = 253;                        //Initializing object  
        s1.name = "Sathish";                //Initializing object  
        System.out.println(s1.id+" "+s1.name);  
    }  
}
```



BITS Pilani
Pilani Campus



Thank You!