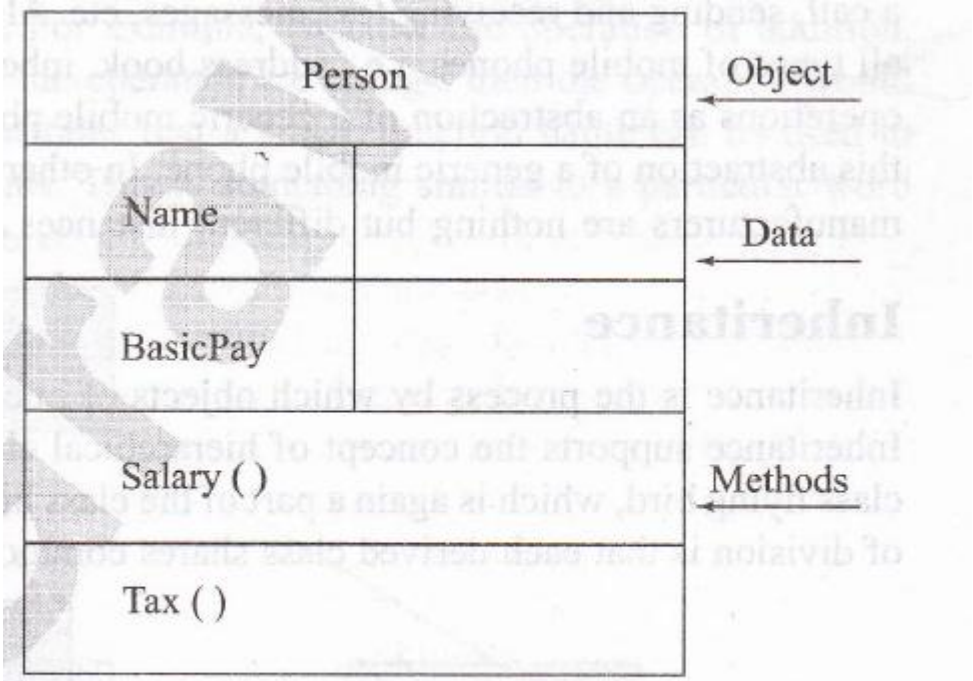
Object-Oriented Programming –A Brief Overview
Lecture-2

Basic Concepts of Object-Oriented Programming

Objects and Classes

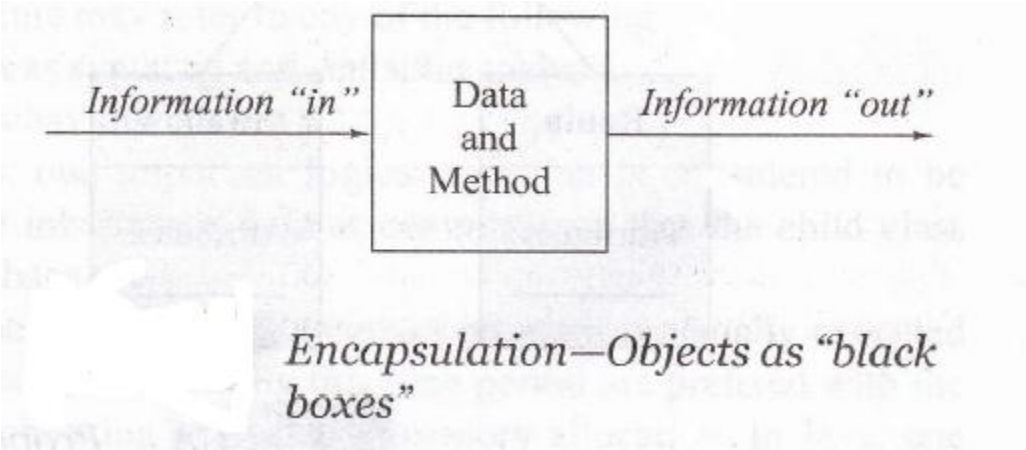
- Objects are the basic runtime entities.
- Objects may represent a person, a place, a bank account etc.
- In execution of program, objects interact by sending messages to one another.
- A class is a user-defined data type with a template that serves to define its properties.
- An object is an instance of a class.



Representation of an object

Data Abstraction and Encapsulation

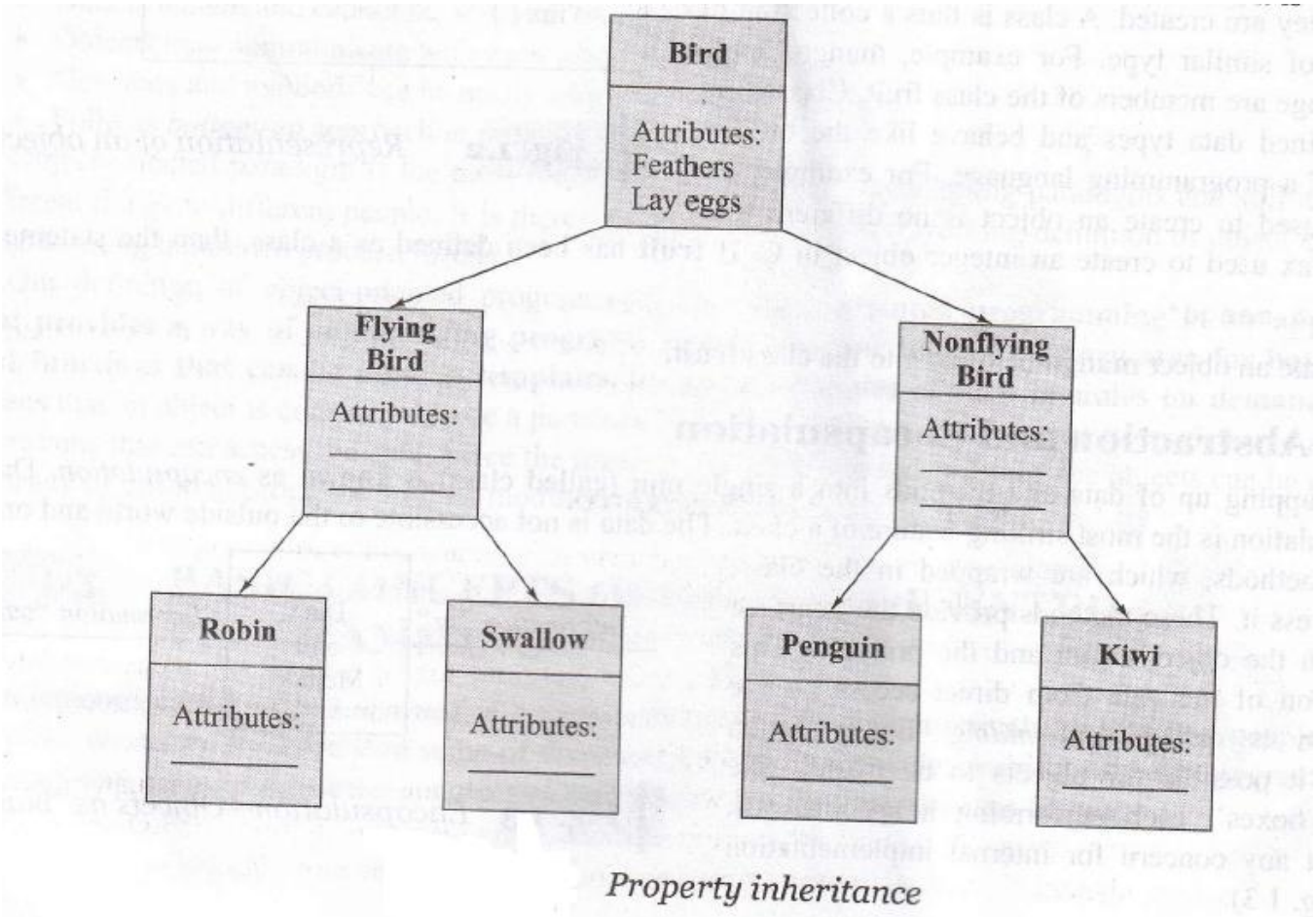
- Wrapping up of data and methods into a single unit (called class) is known as encapsulation.
- Data is not accessible to the outside world and only those methods, which are wrapped in the class, can access it.
- Abstraction refers to the act of representing essential features without including the background details or explanations.



Encapsulation—Objects as "black boxes"

Inheritance

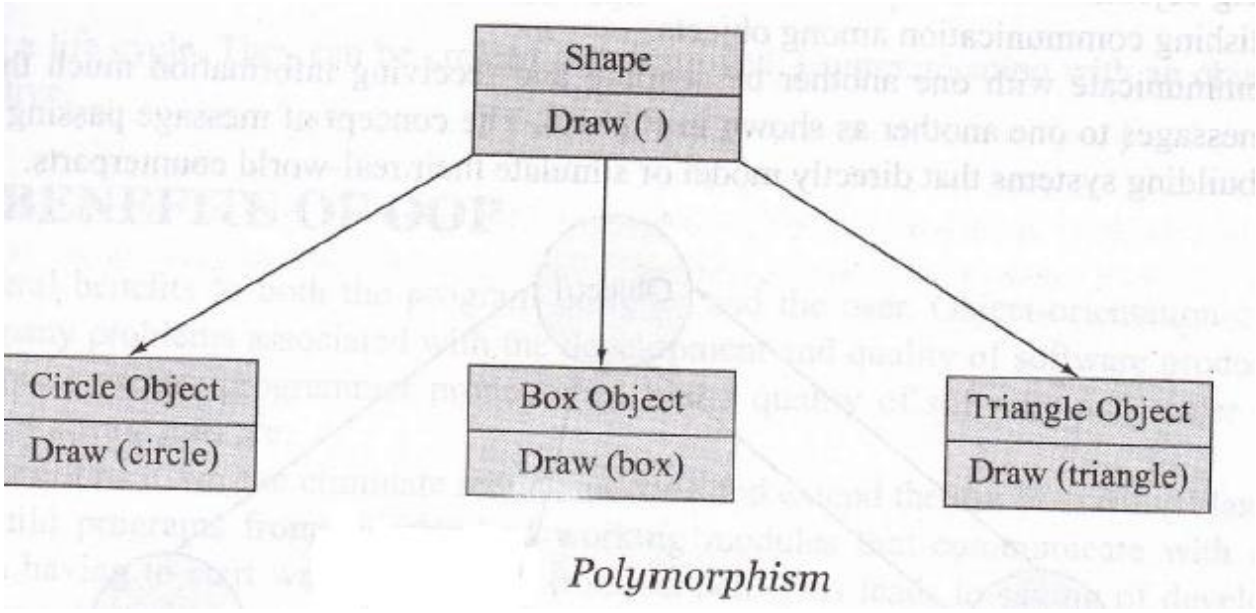
- Inheritance is the process by which objects of one class acquire the properties of objects of another class.
- Inheritance supports the concept of hierarchical classification.



Property inheritance

Polymorphism

- Polymorphism is the ability to take more than one form.
- Polymorphism plays an important role in allowing objects having different internal structures to share the same external interface.



Polymorphism

Example of Inheritance

```
package Inheritance;

class Person
{
  private String name;

  public void setName(String n)
  {
    name = n;
  }

  public String getName()
  {
    return name;
  }
}
```

```
package Inheritance;

class Student extends Person
{
  private String stuNum;

  public void setStuNum(String sn)
  {
    stuNum = sn;
  }

  public String getStuNum()
  {
    return stuNum;
  }
}
```

```
package Inheritance;

public class TestInheritance
{
  public static void main(String[] args)
  {
    Student stu = new Student();
    stu.setName("John Smith");
    stu.setStuNum("12345");
    System.out.println("Student Name: " + stu.getName());
    System.out.println("Student Number: " + stu.getStuNum());
  }
}
```