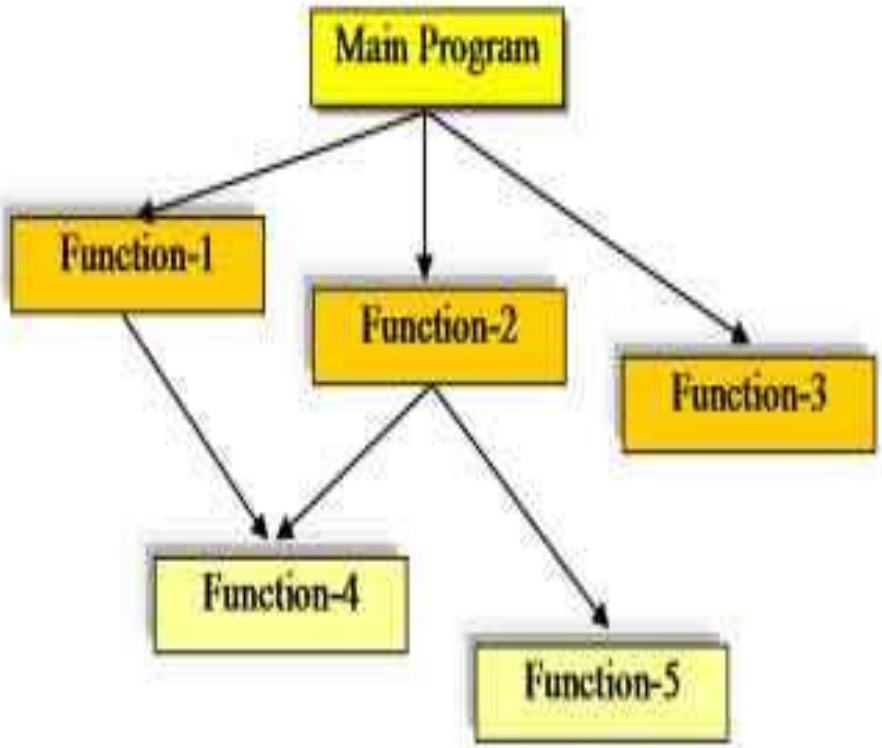
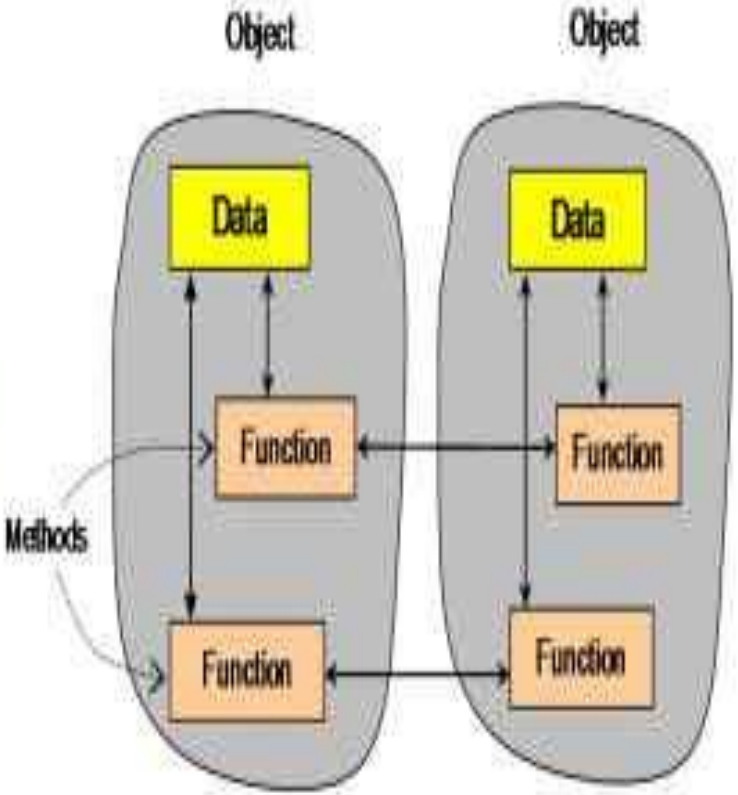
OOPS

Procedure-oriented Programming



Object-oriented Programming



Difference:	Procedure Oriented Programming	Object Oriented Programming
Divided Into	In POP, program is divided into small parts called functions.	In OOP, program is divided into parts called objects.
Importance	In POP, Importance is not given to data but to functions as well as sequence of actions to be done.	In OOP, Importance is given to the data rather than procedures or functions because it works as a real world.
Access Specifiers	POP does not have any access specifier.	OOP has access specifiers named Public, Private, Protected, etc.
Data Moving	In POP, Data can move freely from function to function in the system.	In OOP, objects can move and communicate with each other through member functions.

Difference:	Procedure Oriented	Object Oriented Programming
Expansion	To add new data and function in POP is not so easy.	OOP provides an easy way to add new data and function.
Data Access	In POP, Most function uses Global data for sharing that can be accessed freely from function to function in the system.	In OOP, data can not move easily from function to function,it can be kept public or private so we can control the access of data.
Data Hiding	POP does not have any proper way for hiding data so it is less secure.	OOP provides Data Hiding so provides more security.
Examples	Example of POP are : C, VB, FORTRAN, Pascal.	Example of OOP are : C++, JAVA, VB.NET, C#.NET.

Object-Oriented concepts

1. Object
2. Class
3. Inheritance
4. Polymorphism
5. Abstraction
6. Encapsulation

CLASSES

- **Attributes**

- **Methods:**
  - Have a behaviour
  - Contain code to implement behaviour
  - Associated using objects
  - Have a set of instructions
  - Have a unique purpose

OBJECTS

- **Object :**

–Presented physically

–Has a state

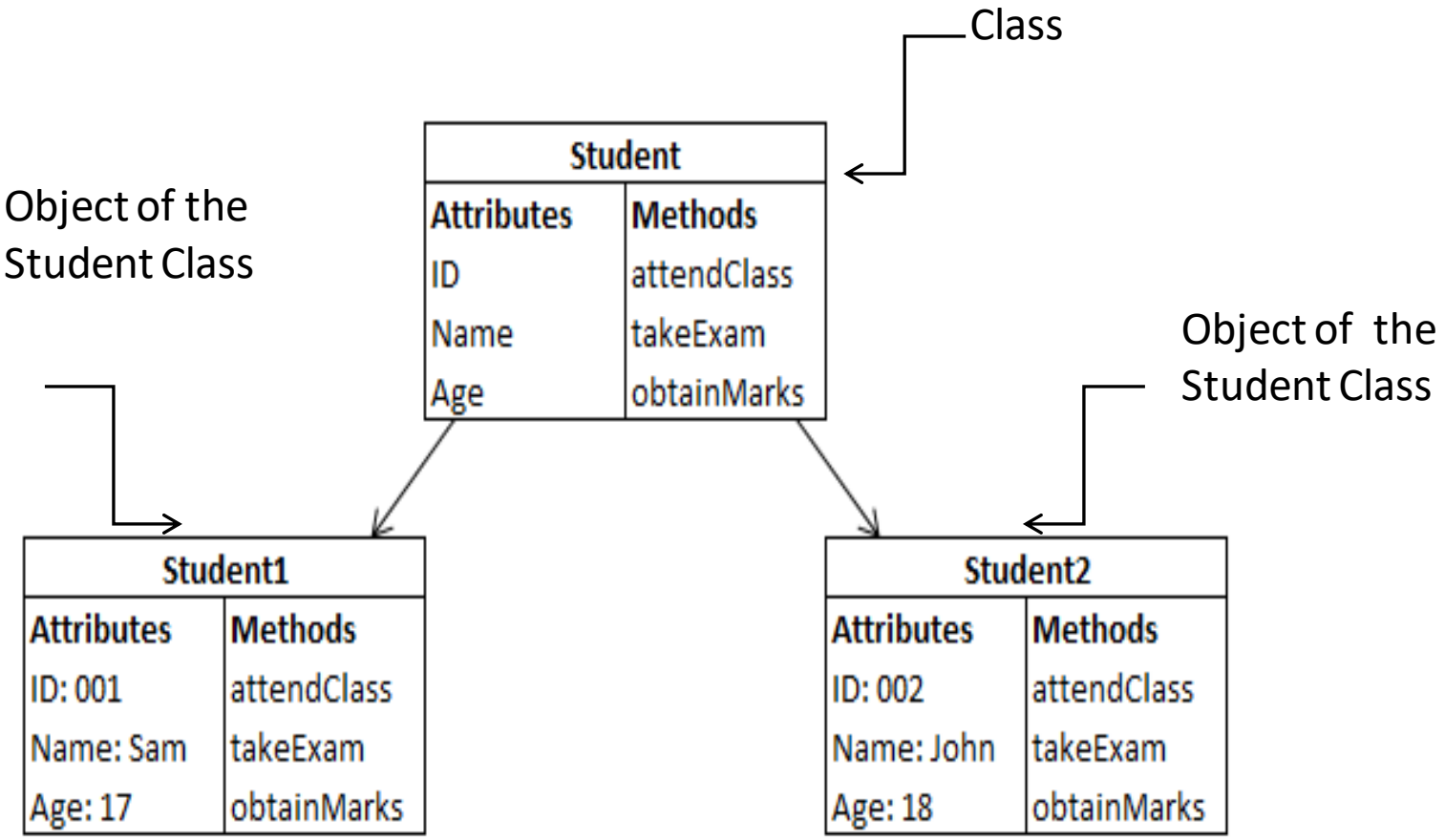
–Displays behavior

–Has a unique identity

Two objects:

–May have the same behavior and state

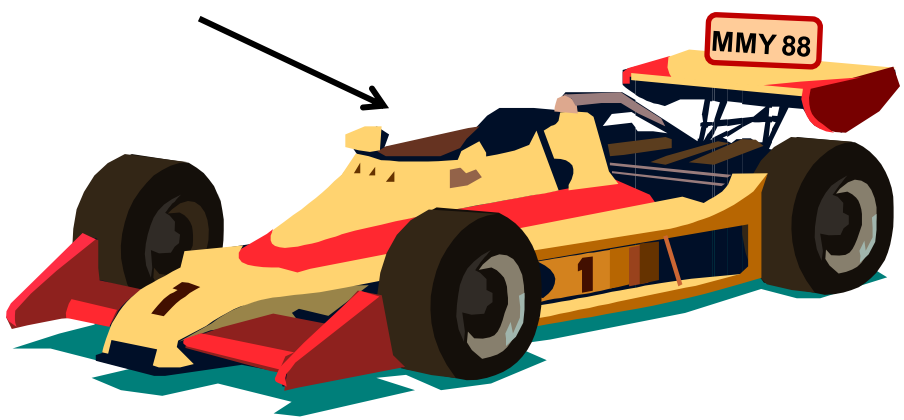
–Can never have the same identity


Class
Student
Attributes
ID
Name
Age
Methods
attendClass
takeExam
obtainMarks
Object of the Student Class
Object of the Student Class
Student1
Attributes
ID: 001
Name: Sam
Age: 17
Methods
attendClass
takeExam
obtainMarks
Student2
Attributes
ID: 002
Name: John
Age: 18
Methods
attendClass
takeExam
obtainMarks

Movement of the car defines its behaviour.

The car's registration number, MMY 88, shows its identity.

A car positioned at one place defines its state.



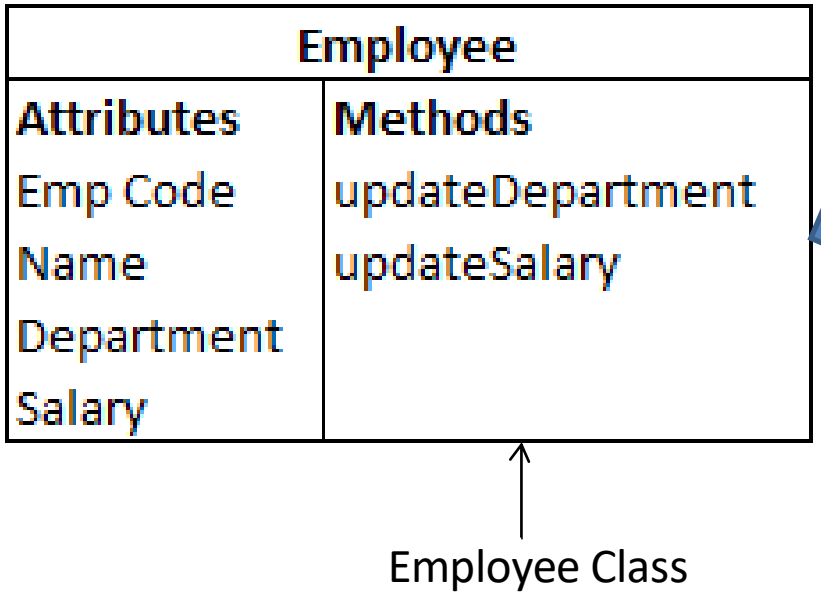
ADVANTAGES OF OBJECT ORIENTATION (CONTD.)

Real-world object

Car	
Attributes	**Methods**
Registration Number	accelerate
Model	deccelerate

Object-oriented representation of the real-world object

ADVANTAGES OF OBJECT ORIENTATION(CONTD.)



Payroll application

Leave accounting application

Employee recruitment application

INHERITANCE

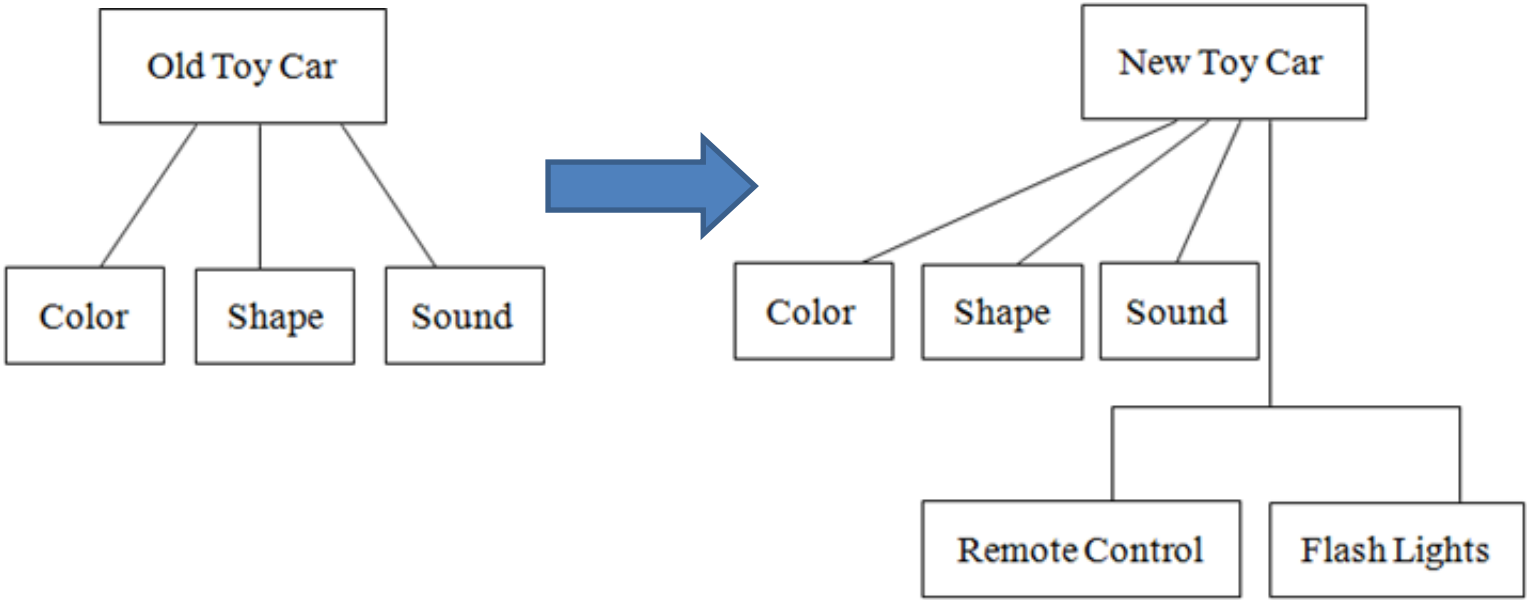
- One of the most useful aspects of object-oriented programming is code reusability.
- Inheritance is the process of forming a new class from an existing class .
- Exisiting class called as base class and new class is formed called as derived class.
- Subclass objects get the copy of the data members and member functions of the super class
- - Subclass:
- Class inherits attributes from another class

–Super class:

- Class from which attributes are derived
-

INHERITANCE

- Flexibility to change:
- Allows to create a new class
- New class can add features to an existing class



Different types of inheritance

- Single
- Multiple
- Multilevel
- Hybrid
- hierarchical

- **Single inheritance**

In which one base class and one derived class

- **Multiple inheritance**

A derived class is created from more than one base class.

- **Multilevel inheritance**

Base class and the derived class are further inherited into the new derived class. This is similar to a relationship representing a child and grandfather.

- **Hierarchical inheritance**

When more than one derived classes are created from a single base this type of inheritance is called hierarchical inheritance. In this program, we have a parent (base) class and two child (derived) classes.

DATA ABSTRACTION

- Data abstraction refers to, providing only essential information to the outside world and hiding their background details, i.e., to represent the needed information in program without presenting the details.
- For example, a database system hides certain details of how data is stored and created and maintained.

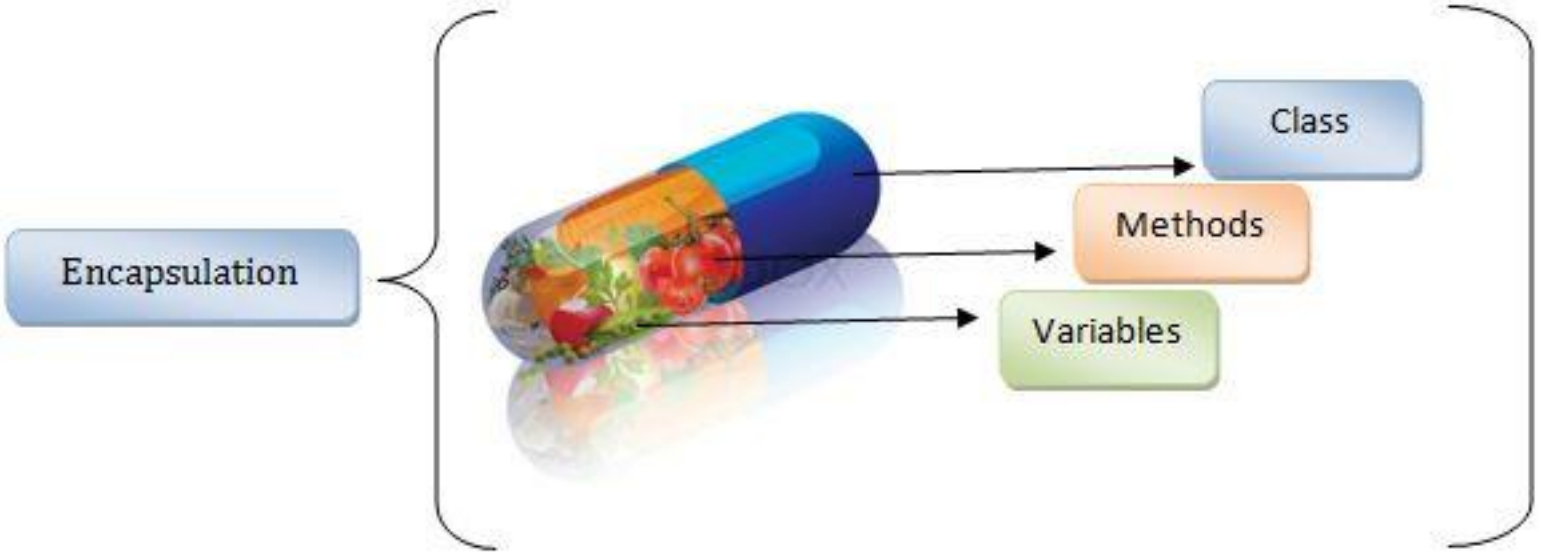
DATA ABSTRACTION

- Information hiding:
- Allows programmers to limit the access to information

A person who is saving a file on a computer does not know how the computer saves the file on the hard disk.
The person just knows that clicking the Save button will save the file.

ENCAPSULATION

- Process of enclosing data and methods within a logical package:
  - Package is called class
  - Package includes attributes and methods
- Leads to information hiding



POLYMORPHISM

- The ability to use an operator or function in different ways in other words giving different meaning or functions to the operators or functions is called polymorphism.
- Poly refers to many. That is a single function or an operator functioning in many ways different upon the usage is called polymorphism.

- Polymorphism is mainly divided into **two**:

- Compile time polymorphism/Static Polymorphism
- Runtime polymorphism /Dynamic polymorphism

POLYMORPHISM

Greek Words and Their Meanings

Poly = Many
Morphos= Forms

Polymorphism = Many Forms
The ability to take more than one form .

- Perform a single action by different ways.