Unit1: Introduction to objected oriented programming

- Programming:
  - The art and science of creating programs
  - Art: application of human creative skill and imagination.
  - Science: systematic study through observation and experiment.
- So, to be a good programmer you should do a lot of experiment and observation of programs by using your own creative skill and imagination.

Contd…

- Program:
  - Program outlines exactly the steps needed to perform a certain task.
  - E.g.,
    - Prepare coffee
    - Add two numbers

Contd…

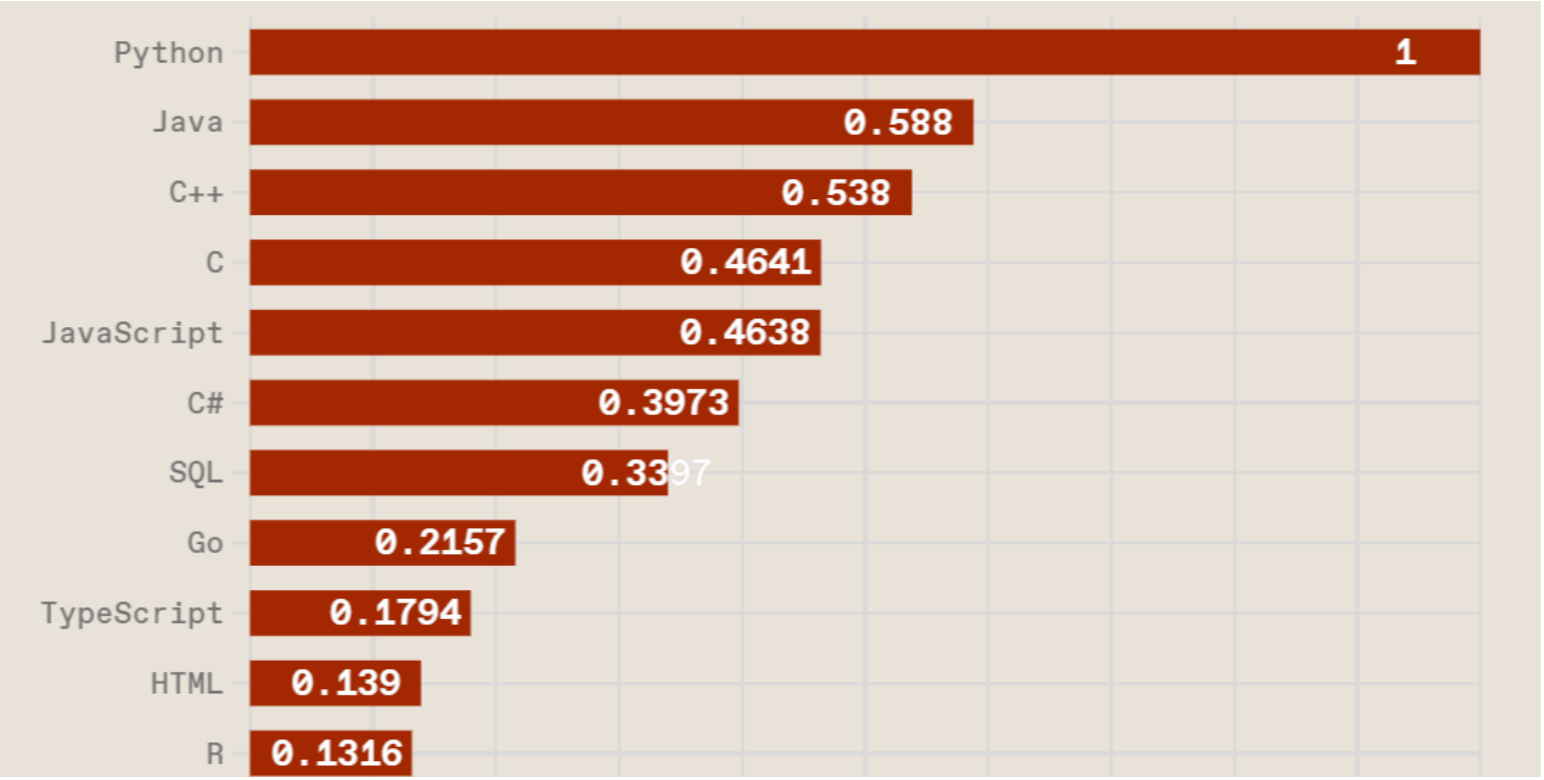
- Programming language:
  - A programming language is a tool used by programmers for instructing a computer or computing device to perform specific tasks.

Contd…

- Classification Of Programming Languages:
  - Machine Language:
    - use 0 and 1 to express the instructions
  - Assembly Language:
    - This language uses mnemonics to express the instructions
  - High-level language:
    - This language uses human understandable like language to express the instructions

Contd..

- **The Top Programming Languages?**
  - IEEE Ranking of programming language:



Programming Paradigms

- Program organizing principles are called programming paradigms.
- Following are the five types of programming paradigm:
  - Unstructured programming
  - Procedural programming
  - Structured programming
  - Object-oriented programming
  - Object-based programming

Contd…

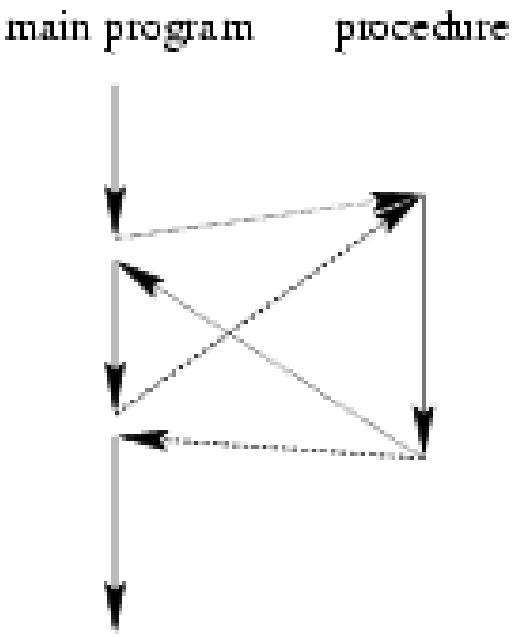
- Unstructured Programming paradigm:
  - A Technique for organizing and coding computer programs in which programmers put all their code in one program.

Contd…

- This programming technique provides tremendous disadvantages (duplicacy and redundancy of code, code maintenance problem, readability and many more) once the program gets sufficiently large
- For example, if the same statement sequence is needed at different locations within the program, the sequence must be copied.
- This has lead to the idea of extracting these sequences, naming them and offering a technique to call and return from these procedures. i.e., Reuse the same code at different places in the program without copying it.

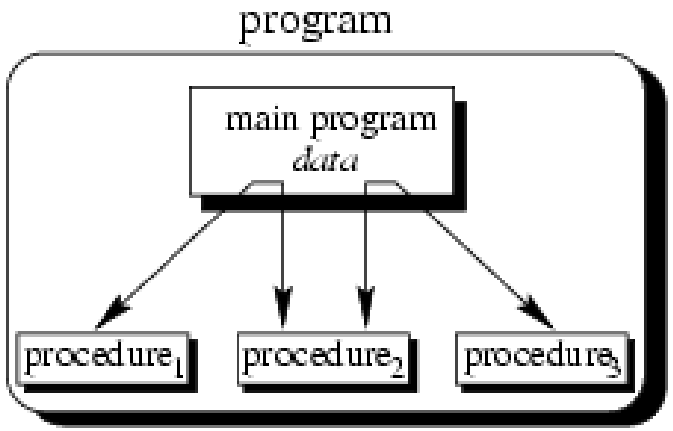
Contd…

- Procedural Programming:
  - Combine returning sequences of statements into one single place
  - A procedure call is used to invoke the procedure
  - After the sequence is processed, flow of control proceeds right after the position where the call was made



Contd…

- With the introduction of parameters as well as procedures of procedures (sub-procedures) programs can now be written more structured and error free
  - For example, if a procedure is correct, every time it is used it produces correct results
- The main program is responsible to pass data to the individual calls, the data is processed by the procedures and, once the program has finished, the resulting data is presented



Contd…

- Now we have a single program which is divided into small pieces called procedures
- To enable usage of general procedures or groups of procedures also in other programs, they must be separately available

- For that reason, modular programming allows grouping of procedures into **modules**

Contd…

- Structured Programming (Modular Programming) Paradigm:
  - During the 1970s it became clear that even well-structured programs were not enough for mastering the complexity involved in developing a large program system
  - It was also recognized that it was necessary to support the division of the program into well-defined parts or modules, that could be developed and tested independently of one another, so that several people could work together within one large programming project
  - Modular programming (subset of procedural programming)is thus concerned with the subdivision of programs into manageable "chunks"

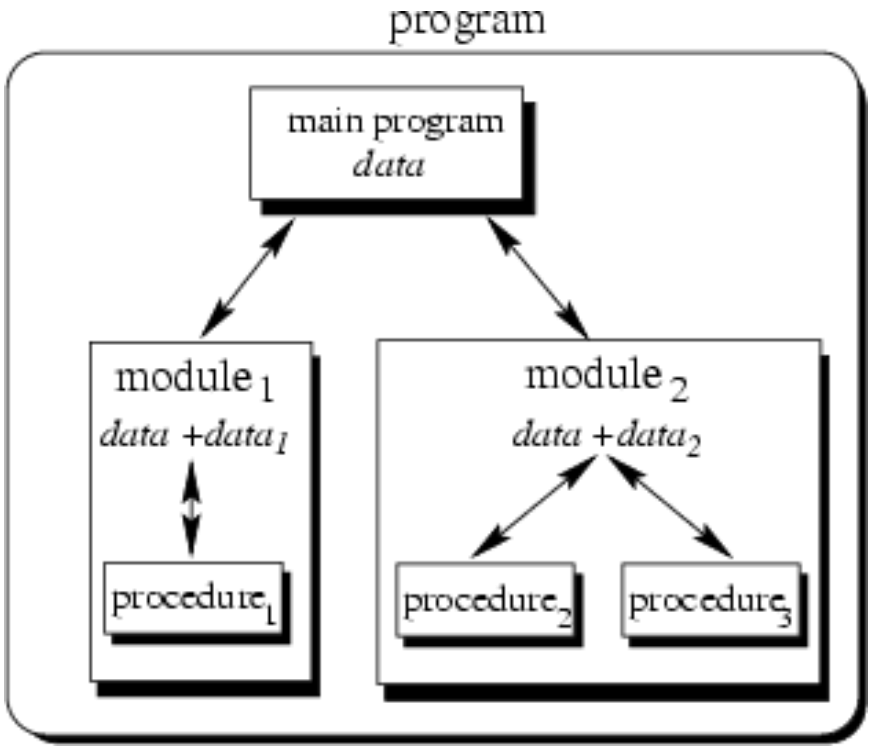
Contd…

- With modular programming procedures of a common functionality are grouped together into separate modules

- A program therefore no longer consists of only one single part.

- It is now divided into several smaller parts which interact through procedure calls and which form the whole program

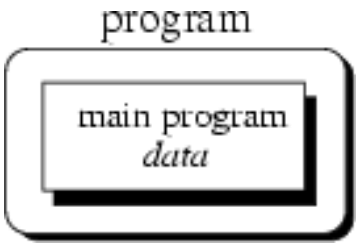
Contd…



- Each module can have its own data. This allows each module to manage an internal state which is modified by calls to procedures of this module

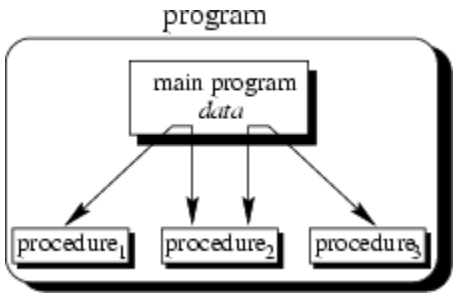
Contd…

Unstructured, procedural, modular programming



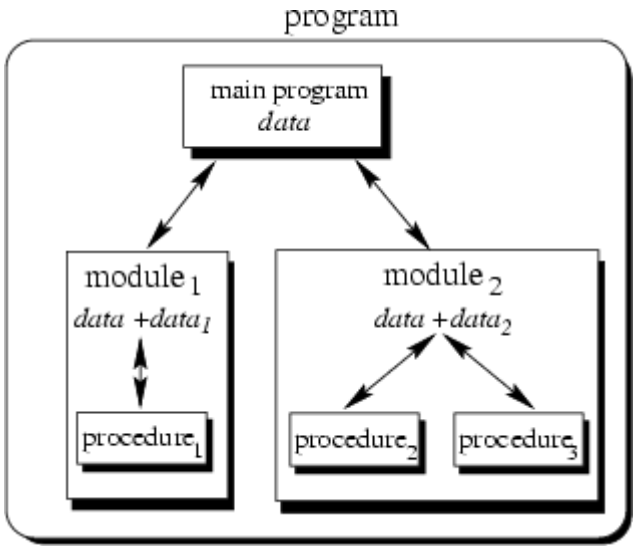
Unstructured programming.

The main program directly operates on global data



Procedural programming.

The main program coordinates calls to procedures and hands over appropriate data as parameters



Modular programming. The main program coordinates calls to procedures in separate modules and hands over appropriate data as parameters

Contd…

- Features of structured programming:
  - Emphasis is given on algorithm rather than on data.
  - Large programs are divided into smaller programs known as functions and functions are grouped into modules.
  - Most of the functions share global data.
  - Data move openly around the system from function to function.

Contd…

- Advantages of structured programming: The following are the major advantages of structured programming.
  - Reduced complexity
  - Increased productivity
  - Portability

Contd…

- Problems with structured programming: Some problems with structured programming are:
  - Complexity of managing large projects
  - Data under-valued
  - Relationship to real world
  - New data types

Contd…

- Complexity of managing large projects:
  - As programs grow larger, even structured programming approach begins to show signs of strain.
  - No matter how well the structured programming approach is implemented, the project becomes too complex, the schedule slips, more programmers are needed, and costs increase.

Contd…

- Data Undervalued:
  - Data is given second-class status in the organization of procedural languages.
  - A global data can be corrupted by functions. Since many functions access the same global data, the way the data is stored becomes critical.

Contd…

- Relationship to the Real World:
  - Procedural programs are often difficult to design because their chief components – functions and data structures – don’t model the real world very well.

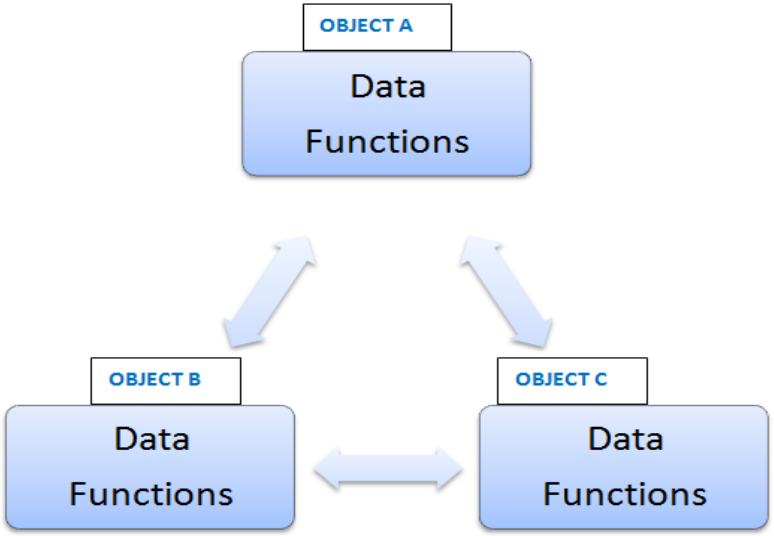
Contd…

- New Data Types:
  - It is difficult to create new data types with procedural languages.
  - Furthermore, most Procedural languages are not usually extensible and hence procedural programs are more complex to write and maintain.

The problems faced in the structured programming approach are the motivating factor in the invention of objected oriented approach.

Object Oriented Programming(OOP) Paradigm:

- In the OOP approach, data and the functions, which are supposed to have the access to the data, are packed together into a single unit known as an object
- An objected-oriented program may thus consists of number of objects .
- Each object of the program is capable of receiving message, processing data , and sending messages to other objects thus, can be viewed as an independent machine with a distinct role and responsibility.

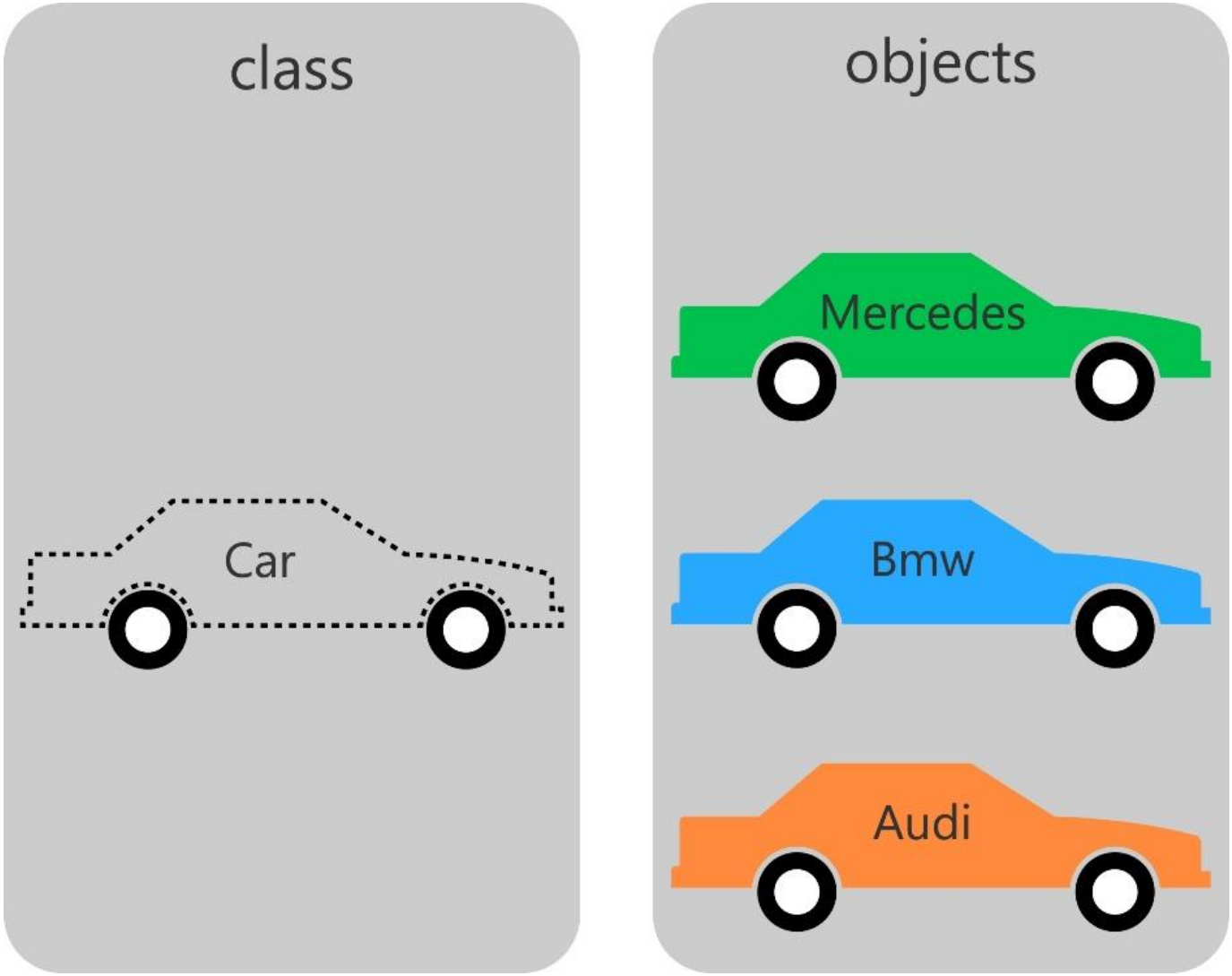


Characteristics(features) of object oriented languages:

- Different features of object oriented programming are:
  - Objects
  - Classes
  - Data abstraction and Encapsulation
  - Inheritance
  - Polymorphism and Overloading

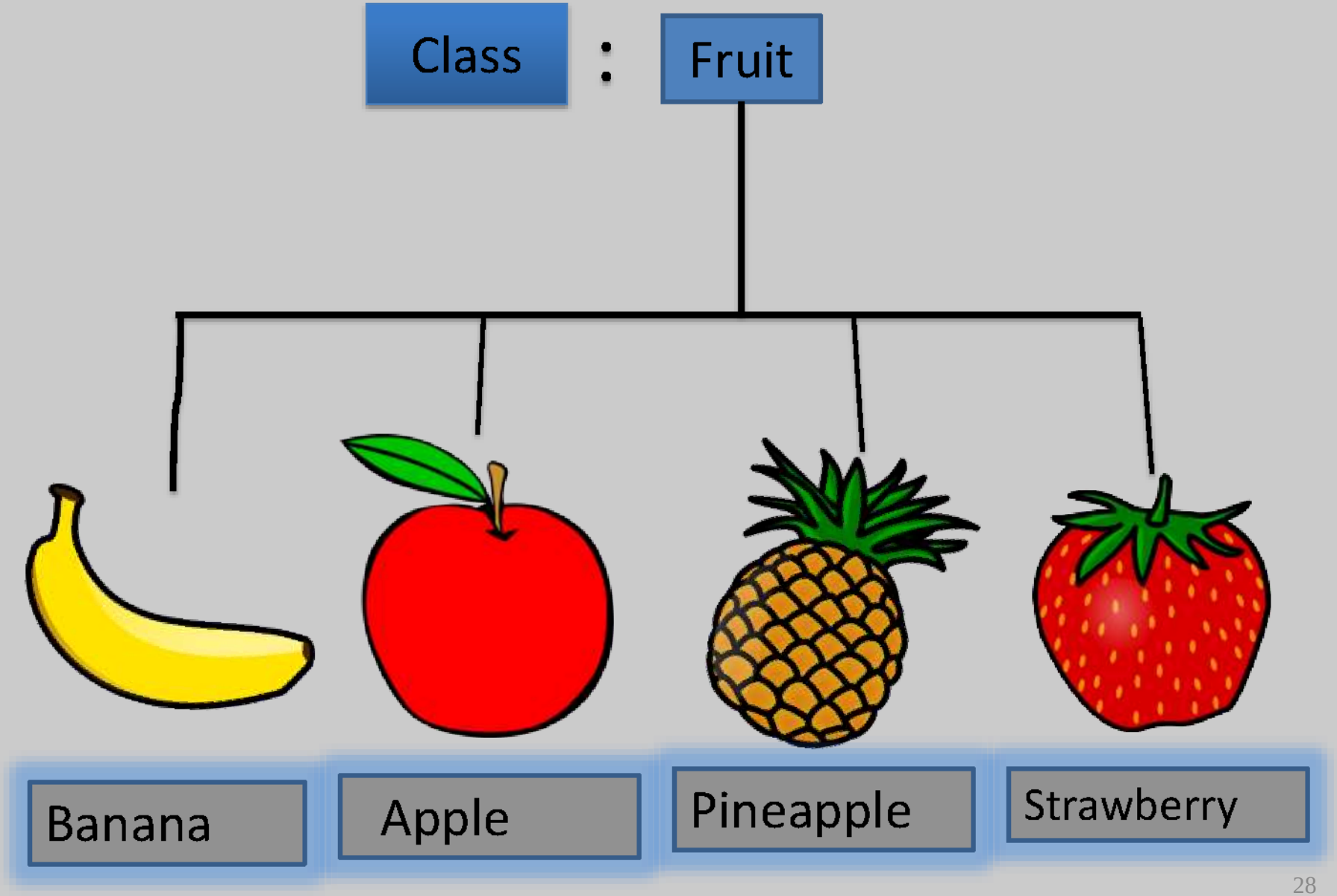
Contd…

- Objects:
  - Objects are the basic run- time entities in an object-oriented program.
  - They may represent a person, place, table etc. They are identified by its unique name.
  - An object is a particular instance of a class. There can be more than one instance.
  - Each object contains data and code to manipulate the data. So, they occupy space in memory.

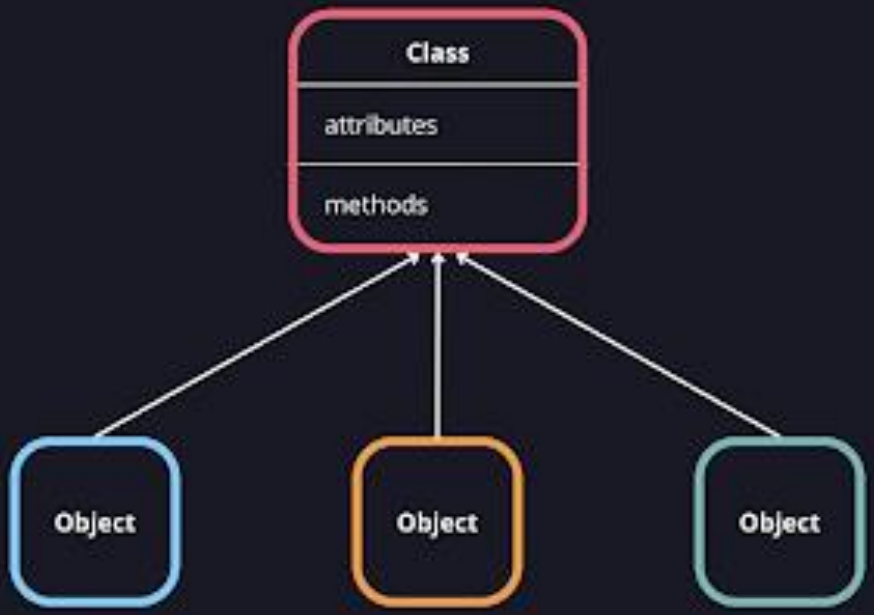

class
Car
objects
Mercedes
Bmw
Audi

Contd…

- Classes:
  - **Collection of objects** is called class. It is a logical entity. So, it does not occupy space.
  - Classes are user defined data types and objects are variables of class.
  - Once a class has been defined we can create any number of objects for that class


Class
:
Fruit
Banana
Apple
Pineapple
Strawberry

Class & Objects



Contd…

- Abstraction:
  - Abstraction means the representation of the essential features without providing the internal details and complexities. i.e. give input get output is only concern
    - For example: phone call, we don't know the internal processing.
  - In OOP, abstraction is achieved by the help of class, where data and methods are combined to extract the essential features only.

Contd…

Real Life Example of Abstraction

Contd…

- **Encapsulation:**
  - Encapsulation is the process of combining the data (called fields or attributes) and functions (called methods or behaviors) into a single framework called class.
  - Encapsulation helps prevent the modification of data from outside the class by properly assigning the access privilege to the data inside the class.
  - So the term data hiding is possible due to the concept of encapsulation, since the data are hidden from the outside world.

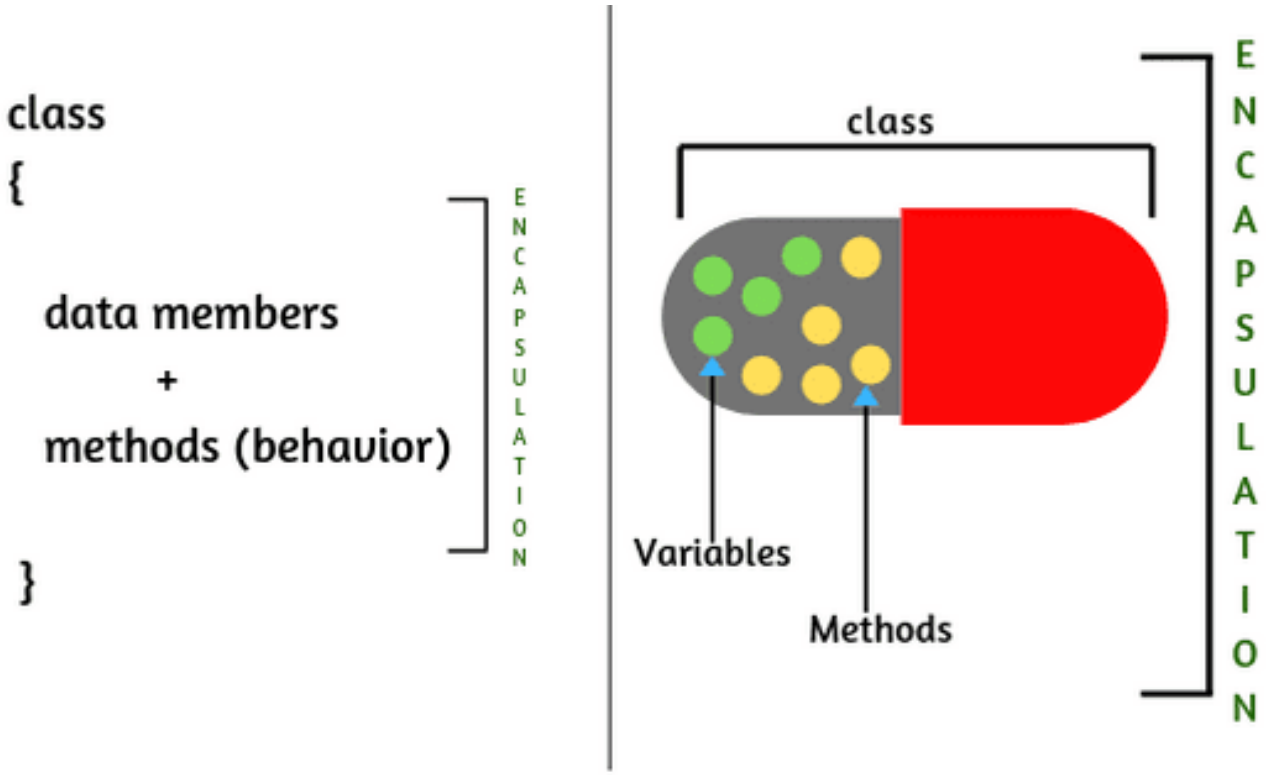
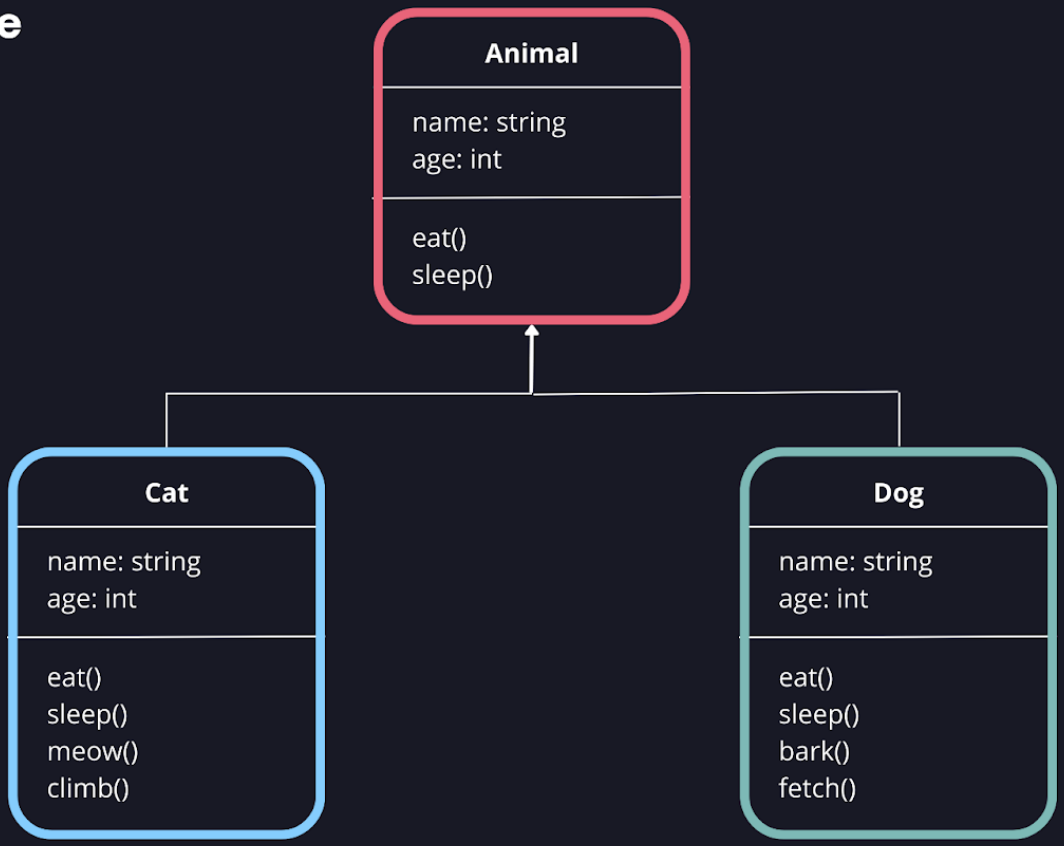


Fig: Encapsulation

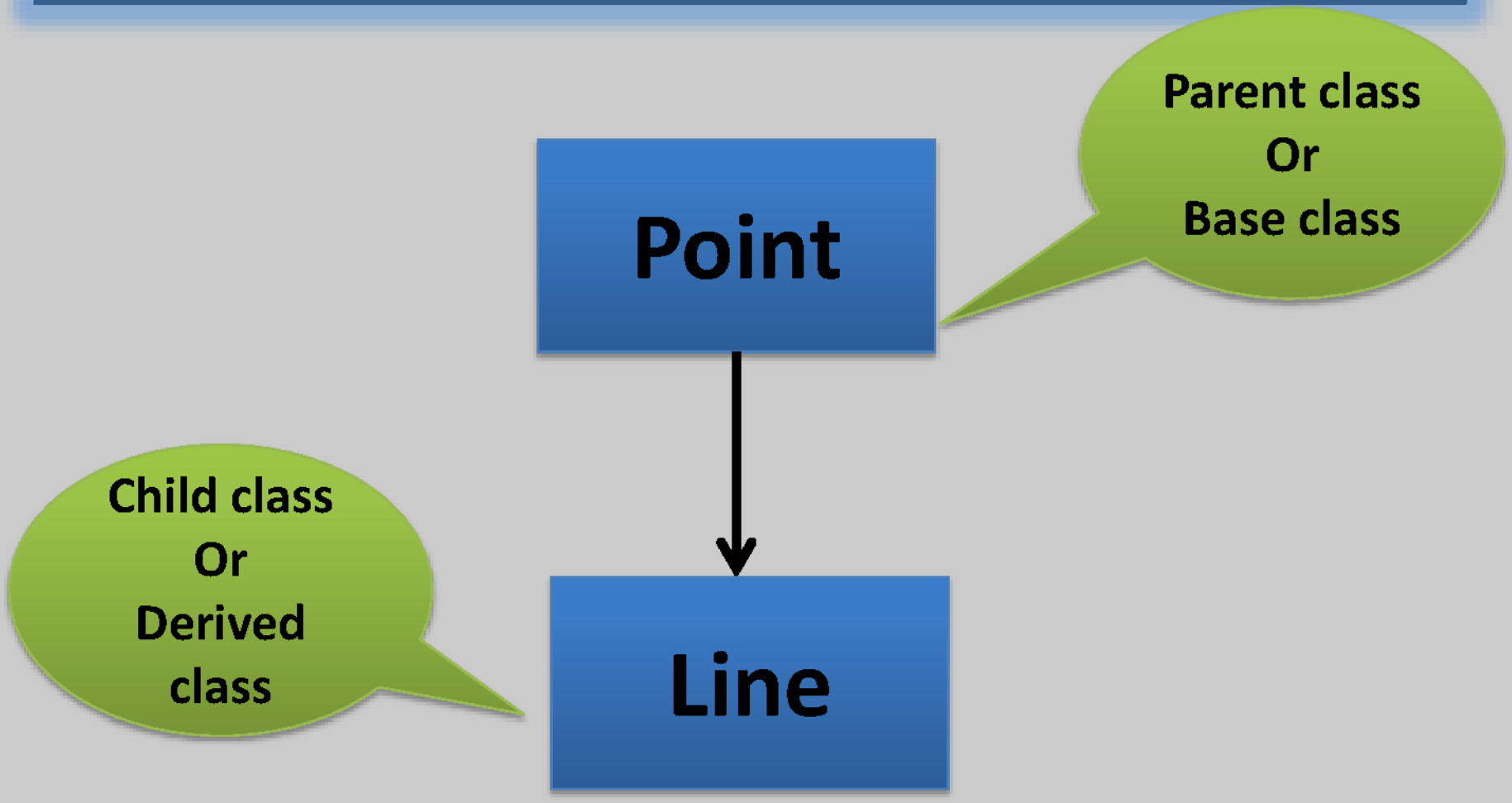
Contd…

- **Inheritance:**
  - Inheritance is the process of acquiring certain attributes and behaviors from parents.
    - For examples, cars, trucks, buses, and motorcycles inherit all characteristics of vehicles.
  - Inheritance leads to reusability:
    - Class can be inherited and modified to accommodate new features.

Inheritance



Inheritance



Contd…

- Object-oriented programming allows classes to inherit commonly used data and functions from other classes.

- If we derive a class(called derived class) from another class (called base class), some of the data and functions can be inherited so that we can reuse the already written and tested code in our program, simplifying our program.

Contd…

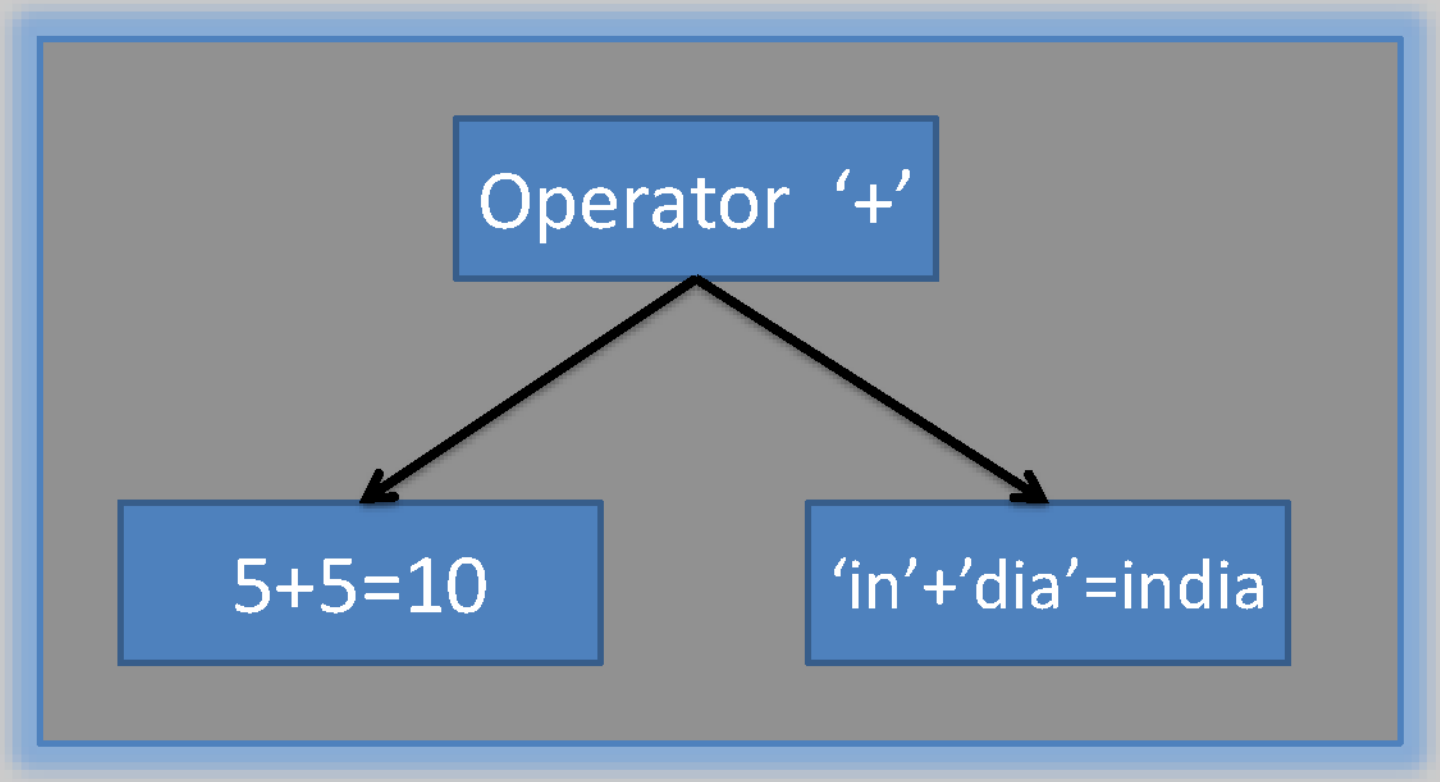
- Polymorphism and Overloading
  - The word polymorphism means having many forms.
  - Real life example of polymorphism, a person at a same time can have different characteristic. Like a man at a same time is a father, a husband, a employee. So a same person posses have different behavior in different situations. This is called polymorphism.

Contd…

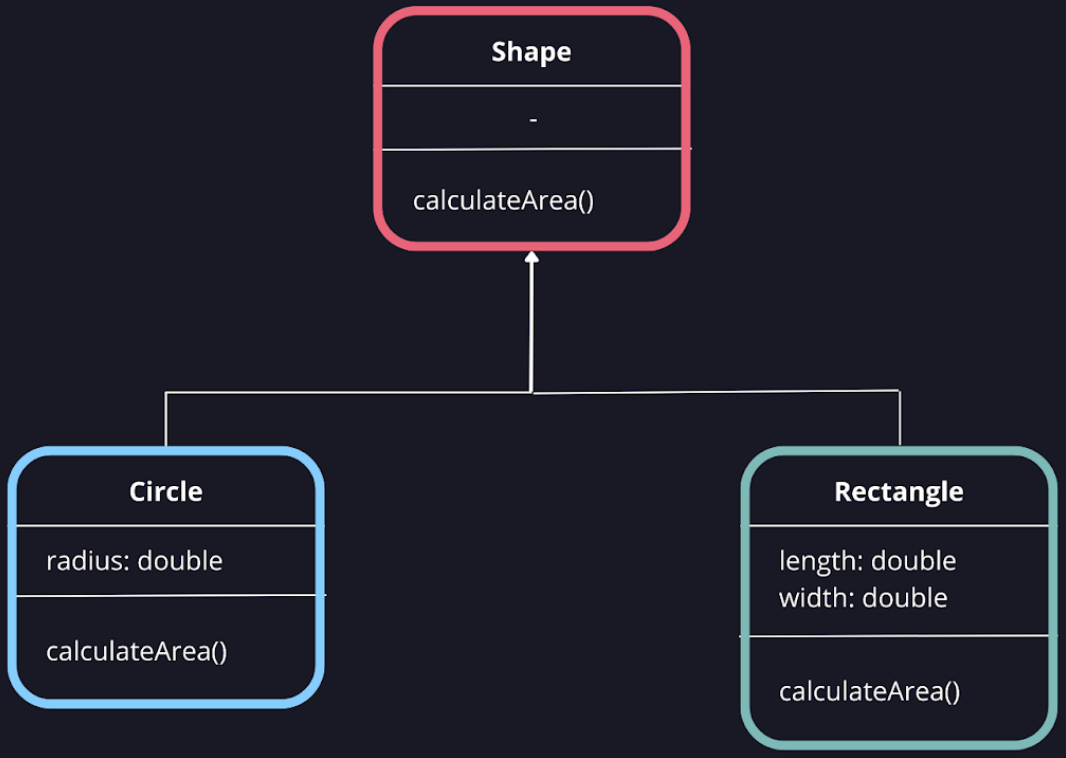
- Polymorphism is considered as one of the important features of Object Oriented Programming.
- **In C++ polymorphism is mainly divided into two types:**
  - Compile time Polymorphism : This type of polymorphism is achieved by:
    - Function overloading or
    - Operator overloading
  - Runtime Polymorphism : This type of polymorphism is achieved by Function Overriding.

For Example

plus '+' is used to make sum of two **number** as well as it is used to combine two **strings**.



Polymorphism



Contd…

- Advantages of OOPs:
  - Code recycle and reuse.
  - Easy to partition the work in a project based on objects. So, Software Complexity can be easily handled and managed.
  - Message passing technique between objects for communication makes interface description with external systems much more straightforward.
  - Possible to map objects in a problem domain within a program.
  - Data hiding is possible.
  - Use of inheritance can eliminate redundant codes in a program.

Contd…

- Disadvantages of OOPs:
  - Compile and runtime overhead.
  - Re-orientation of software developer to object-oriented thinking.
  - Requires the mastery in software engineering and programming methodology.
  - Benefits only in long run while managing large software projects.
  - The message passing between many object in a complex application can be difficult to trace and debug.

Structured Programming	Object Oriented Programming
Structured Programming is designed which focuses on procedure and then data required for that procedure.	Object oriented programming is designed which focuses on data.
Structured programming is also known as modular Programming and a subset of procedural programming language.	Object oriented programming supports inheritance, encapsulation, abstraction, polymorphism, etc.
In structured programming, programs are divided into small self contained functions.	In Object oriented programming, Programs are divided into small entities called objects.
Structured programming is less secure as there is no way of data hiding.	Object oriented programming is more secure as having data hiding feature.
Structured programming can solve moderately complex programs.	Object oriented programming can solve any complex programs.
Structured programming provides less reusability, more function dependency.	Object oriented programming provides more reusability, less function dependency.
Less abstraction and less flexibility.	More abstraction and more flexibility.

Assignment-1

1. Explain characteristics and limitations of procedural programming? How OOP overcomes those limitations.
2. What is objected oriented programming? Explain objects, class, encapsulation, data hiding, inheritance, and polymorphism.
3. Differentiate between structured programming and OOP
4. “Inheritance and polymorphism is important features of OOPs” explain and justify this statement.

End of Unit 1

Thank You !