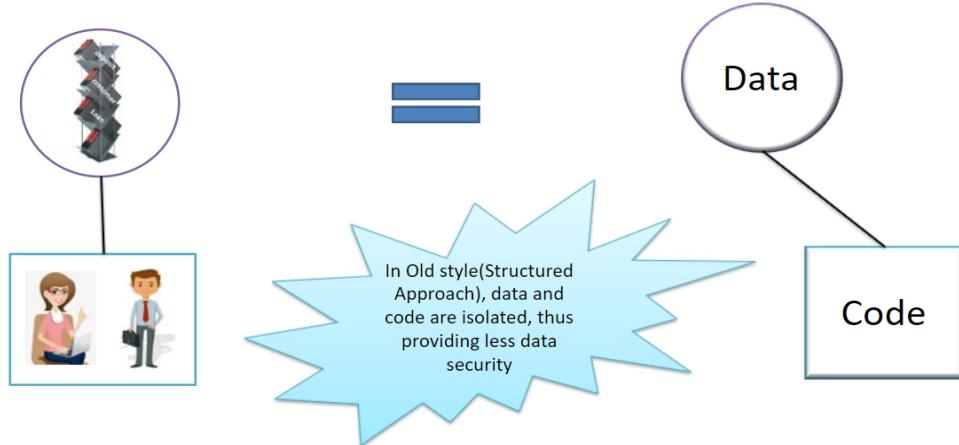
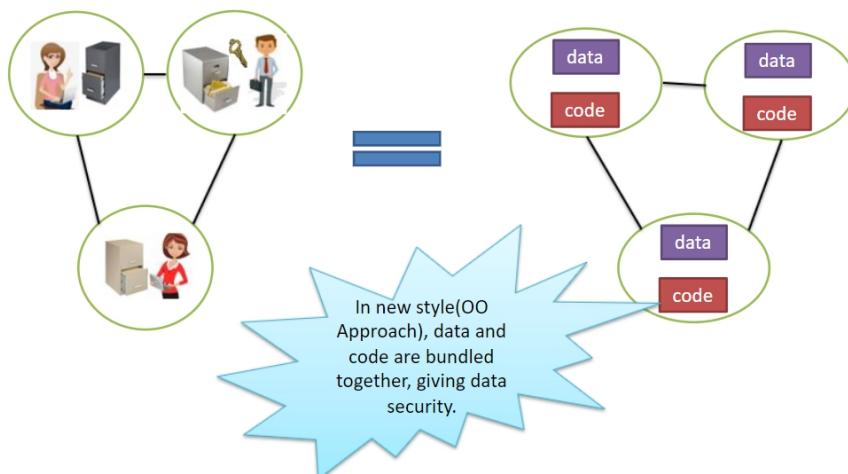


Overview



Overview



Object Oriented Concepts



An Object Oriented program consists of many well encapsulated objects, and interacts with each other by sending messages

The object-oriented languages focus on components that the user perceives, with objects as the basic unit.

Benefits

Ease in software design as you could think in the problem space rather than the machine's bits and bytes.

Ease in software maintenance: object-oriented software are easier to understand, therefore easier to test, debug and maintain.

Reusable software: you don't need to keep re-inventing the wheels and re-write the same functions for different situations.

Object Oriented Concepts



The basic unit of OO Approach is

- Class
- Object
- messages

TeamLead : Richard



Web Programmer : Hawking



Object

Message passing

Object

Class



Classes are the fundamental building blocks of a Java program

A class is a definition of objects of the same kind.

Class can be defined as a blueprint, template or a prototype that defines and describes the *attributes* and *behaviors* common to all objects of the same kind.

Attribute of a class



Attribute is a named property of a class describing a range of values that instances of the class may hold for that property.

An attribute has a type and only the holding object is able to change the values of its own attributes.

The set of attribute values defines the state of the object



27, 5th street,
T. Nagar,
Chennai

My address

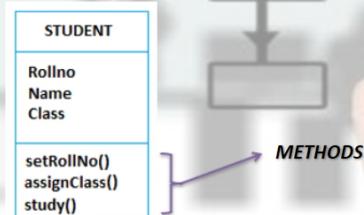
ATTRIBUTES

STUDENT
RollNo Name Class
setRollNo() assignClass() study()

Methods of Class



Methods are members of a class that provide a service for an object or perform some business logic.



Object



An object is an instance of a class.

Objects are the real time entity which are created through their template, their class

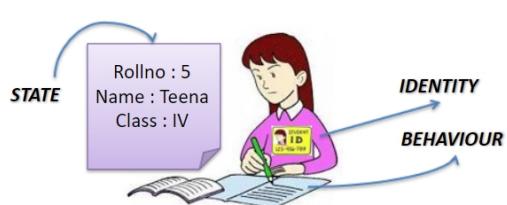
Object is dynamic



Properties of Object



- State : An object's state is defined by the attributes of the object and by the values they have.
- Behavior : Behavior is how an object acts and reacts in terms of its state changes and message passing
- Identity : Identity is that property of an object which distinguishes it from all others



Principles of OOP



Abstraction

Encapsulation

Polymorphism

Hierarchy

Modularity

Typing

Persistence

Principles of OOP



In Real Life...



Abstraction



Login Validation
Process Hidden



Inheritance



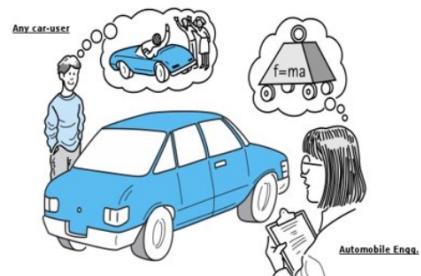
Polymorphism

Abstraction



Abstraction includes the essential details relative to the perspective of the viewer

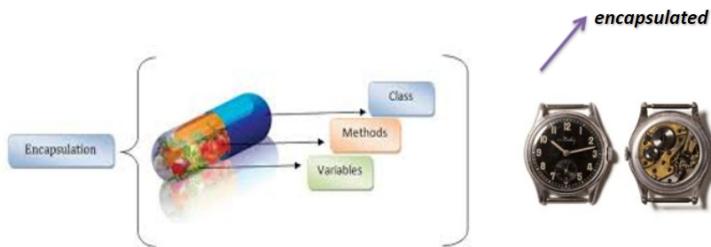
Abstraction allows us to manage complexity by concentrating on the essential aspects making an entity different from others.



Encapsulation

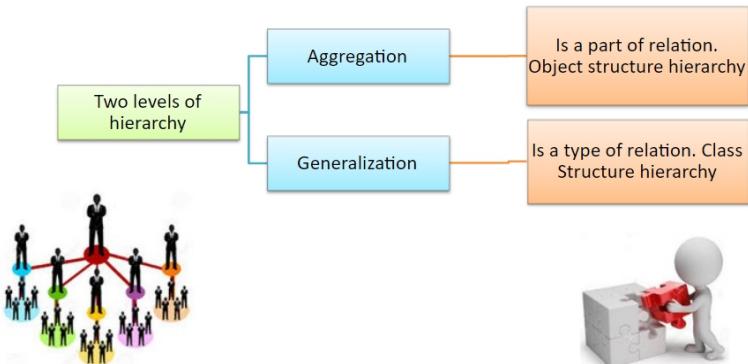
Encapsulation is the process of binding data and functions together into a single functional unit.

Provides data security using information hiding



Hierarchy

Hierarchy is ranking or ordering of abstractions.



Generalization

Objects of the real world often form conceptual hierarchies.

Ordering of abstractions into a tree-like structure.

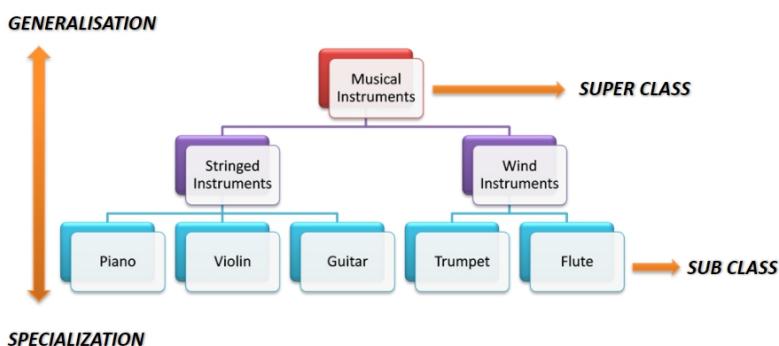
The parent class defines the common properties of one or more child classes.

The child classes derive the properties of the parent and also have specific properties of their own

A parent class itself can have another parent and hence forming a hierarchy.



Generalization



Polymorphism



Polymorphism Shape



Having multiple forms

Refers to a **programming** language's ability to process objects differently depending on the context

Polymorphism could be

Overloading - Static polymorphism in simple words means two methods having the same method name, but taking different input parameters.

Overriding - derived class is implementing a method of its super class.

Circle

Square

Triangle

Aggregation

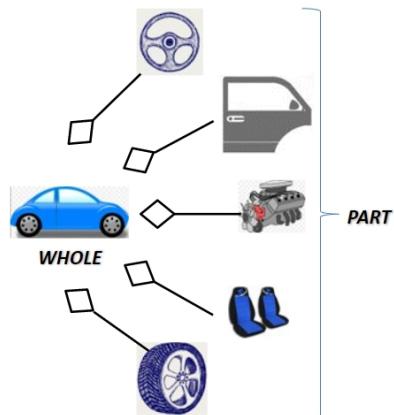


Aggregation represents a “whole-part or a part-of” the relationship.

When an object has another object within it, it is a ‘has-a’ relationship, called aggregation

When the whole Object is destroyed the part object can still be used

Stronger form of aggregation is composition
Eg: House and room



Modularity



- Modularity is a method to divide a program into smaller units
- Modularity is implemented in java as packages



Typing



Identifies a programming language type conversion and characteristics.

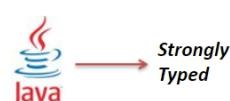
It can be either strongly or weakly typed

Strong typing requires explicit conversions for conversions between different data types in expressions.

Typing is inherently Language Implementation Dependent

`int a=10.5;`

Weakly
Typed



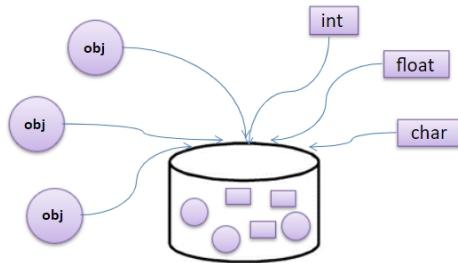
Strongly
Typed

Persistence

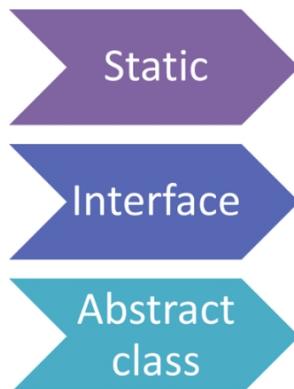


Persistence is the ability of an object or data to outlive the lifetime of the program that creates, accesses or modifies it.

A persistent object is one that continues to exist after a program that creates or uses it, terminates.



Few more concepts of OOP



Static



Exactly! What do you infer from this scenario? Certain things are available for each individual apartment, whereas certain things are common for all apartments.



Lift, Generator, Gym,
Swimming Pool



Static - Overview



Likewise, in an object oriented way, certain attributes will be applicable for each instance of that class and some attributes will be common to all instances in a class. These common attributes are termed static.



CompanyName="Teknoturf"

Abstract class

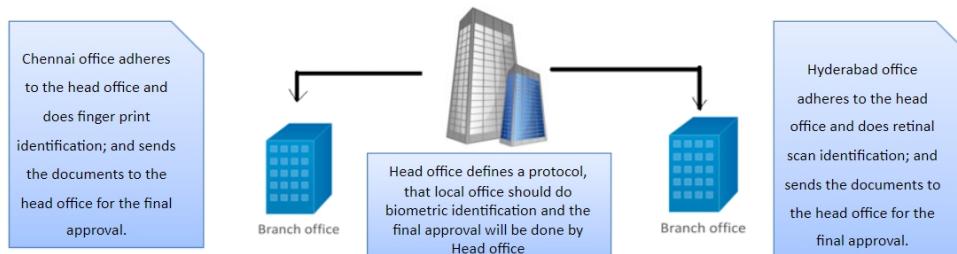
Methods that do not have any implementations are called as abstract methods.

Abstract class may contain zero or more abstract methods

Abstract class cannot be instantiated

Abstraction- Real world scenario

Let us consider the Aadhaar card issuing process. The head office for the final approval is present in Bangalore and we have two local offices one at Chennai and the other at Hyderabad. Aadhaar card issuing involves 2 steps. Biometric identification and the other is final approval and issuing the acknowledgement. Chennai does the biometric identification through the finger print and Hyderabad does it through retinal scan. So the Bangalore head office has decided to delegate the biometric identification to both the local offices and take care of the final approval and issuing the acknowledgement



Interface

Interface is a contract for the classes to implement the behavior in their own way.

All methods inside the interface are public and abstract



Interfaces

RBI fixes the base interest for Home Loan.

All banks calculate the interest for home loan based on some specific factors of theirs.

Here RBI acts as an interface with the method calculateHomeLoanInterest as abstract

All banks implement this interface and give implementation for this abstract method

