# OOPs in Java.

-By Engineering animutyam.

→ Manaki nachina datatype ni create cheyadaniki classes vadalthamu.

→ class names first Cap lettle undali Chaishop.

→ Constructor :- oka class ki object create chesinappudu its method adhi automatatic ga call iyidhi

→ Constructor ki no return type & Name is same as class name.

→ Method Overloading :-

Same method name but it should be different in any one of the below:

→ no. of parameters
→ Datatype of parameters
→ Order of parameters.

→ Method Overriding:

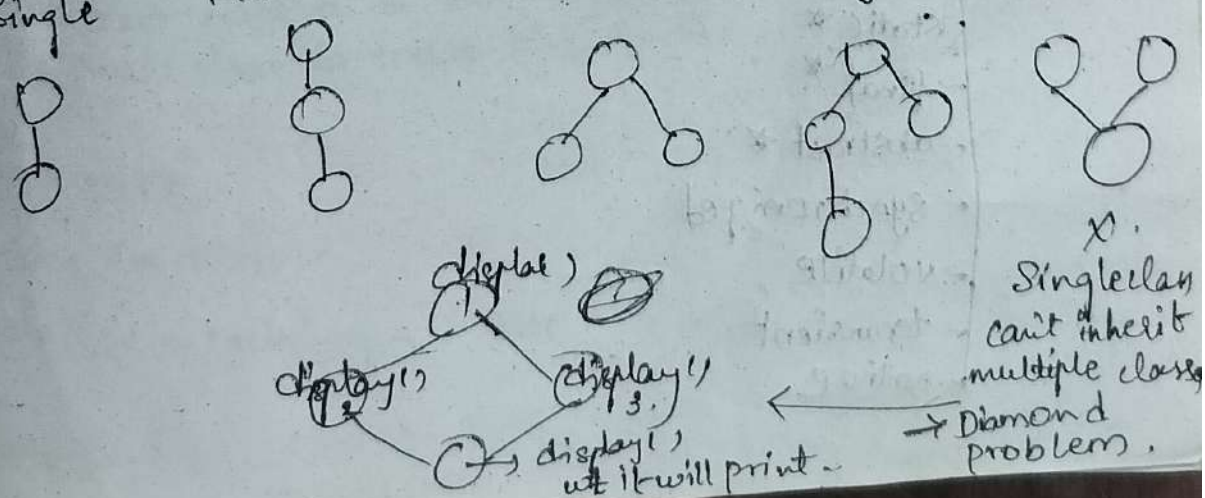
Same method name with same number & type & order of parameters.

@ Override - annotation. chesthadhi

parent class method ni child class method override

Types of Inheritance.

Single Multilevel Hierarchical Hybrid Multiple




X. Single class can't inherit multiple classes

→ Diamond problem.

Interface :- (implements).

Multiple inheritance kosam vachindhi interface.

→ All variables should be initialized in interface

→ method should not have body

→ Every method should be overrided in child class.

Inner classes :-

```
class Lopala class.
class ChaiShop{
    String branchName = "Boom";
    class Tikka {
        String yevariki = "nake";
    }
}
class Main {
    P S V m (S a[ ]){
        ChaiShop b1 = new ChaiShop();
        S.O.ph(b1.branchName);
        ChaiShop.Tikka th = b1.new Tikka();
        S.O.pln(th.yevariki);
    }
}
```

object creation for inner class.

Java Packages :-

Data Modifiers.

- Acces Modifers - public, protected, default, private
- Non-acces Modifiers.
  - static *
  - final *
  - abstract *.
  - synchronized
  - volatile
  - transient
  - native

Static:

→ static variables class ki belong ithe object ki kadhu.
→ static methods static variables ne access cheyagaladhu. so, non-static variables ni access cheyaledhu.
→ static inner classes ki vadochu. Outer main class ki static pettalem.

=0=

→variables. →methods.

Parent pc = new Child()
↓ ↓
reference Object.

→ time to variables tavalante parents ve print ithay
→ methods unte override chesi child method ni print chesthadhi. If in case child lo method lekapothe parent ye print chesthadhi

→ static methods can't be overrided.

=0=

Final:

→ final ni oka variable ki pedithe danni modify cheyalem
final int b=5.
~~Always~~ initialize final variable.
→ final methods can't be overrided
→ final classes cannot be inherited or extended.

=0=

Abstract:

→ class ki abstract ani pedithe daniki object create cheyalem
→ abstract class lo both abstract & concreate(normal) methods untay.
→ abstract methods must be overriden when they are extended by other classes.
→ abstract methods ni only abstract class lo ne create cheyali Normal class lo create cheyakudadhu.

=0=

Synchronized:

→ Race Condition.

Volatile: no caching, take variables from memory.

=

→ transient :- code ni byte streams lo store cheyadanni serialization antaru. & Deserialize - reverse

→ oka variable ki transient ani pedithe, daniki Serialize cheyadhu.

Native: declares a method implemented in language like c or cpp

Features of OOPs:

1) Encapsulation:- oka class ni create chesttam by using variables & methods

→ Private variables ni modify cheyadaniki getters & setters ni use chesttaru.

2) Abstraction:- hides implementation details.

→ achieved using abstract classes & interfaces.

3) Polymorphism:

CTP (Method overloading) RTP (Method overriding)

4) Inheritance:

5) Dynamic Binding - RTP:

Constructors in Inheritance:-

super():-

→ parameterised constructors & non-parameterised constructors.

↳ Default constructors.

Singleton class:

It is a class that allows only one object to be created

→ constructor ki private pedithe, object create cheyalem a class ki bayata. class lopala create cheyali