# JAVA OOPS:

↳ create objects
real world easily depicted

Classes & Objects
↳
property & Behaviour
state

problems divided in chunks.

→ Class [depicts properties & Behaviours of objects]

```
class className{
    //method
    //fields
}
```

→ Object

~~Object~~ ~~className~~
class className = new class();

→ Method

```
class ClassName{
    int name; ——— property
    void methodName(){
        }
    void methodName(int value){
        }
}
```

behaviour

Will work as different parameters but same methods can't be executed
this is overloading

## Constructor

public Animal() {

}

default constructor → used when made object.

multiple constructor can be made.

## STATIC

one properties and methods should be static. fixed

→ if we want to access class members without making objects then we need to use static in that class members.
(property & behaviour)

## THIS

used to refer current object in method or constructor.

→ argument → ex: when we need to access the same object
→ ambiguity in naming
→ constructor overloading → this(); → used to call another constructor

## Inheritance

helps to make new class using already existing class

super
parent class [original class] child class [sub class] → made from original class

class subclassName extends superClassName {

→ one class can extend only once at a time

## Polymorphism

Run time polymorphism : It will find the methods during running

Compile time polymorphism : as soon as we write code it knows which function to be called.

like overloading of methods in same class

↳ same class [compile time]

overridding → when finding in parent class [run time]

## Access Modifiers

private → only that class can access the method contained

public → any class can access that methods

default → one class can use it in other class from same package.

protected → if another class from same package wants to access and are children