What is oop?

Object Oriented Programming

Combo of entities allows u to combo data elements and operational elements into single elements = class

Advantages:

Data and functions combined

Better modularity

Low coupling

Re-use easier

Closely modelling the rl world

4 principles of oop:

Abstraction

Encapsulation

Inheritance

Polymorphism

Abstraction:

e.g. car. Abstract understanding of a car e.g. doors, windows etc. shared understanding of what a car is and features that they have. Can mean many things but common criteria to identify. Also operations for it like turning

this is a class. Defined by 2 main things: data it is going to contain and methods it is going to perform.

Need 3 elements to define class: 1. Specify what the properties are of an instance of this particular type/class. 2. Specify what the operations are. 3. Specify name of class we r creating

When want to use e.g. create an instance, have to provide acc values for the dif characteristics of object

Diagram

Description automatically generated

Coding examples of abstraction:

class Customer: # class always capitalised

# everything else always indented under

pass #keyword in python to indicate we r not doing anything

^ this is enough to satisfy python syntax.

Create new instance of a class:

def \_\_init\_\_(self, fname, lname) #self refers to current object and means the object we r currently creating

self.firstname = fname

self.lastname = lname

def print(self):

print(f’Full name: {self.fullName} {self.lastName}

Now in main .py can:

import Customer

cust = customer.Customer(‘Jyoti’, ‘Suresh’) #creating customer object

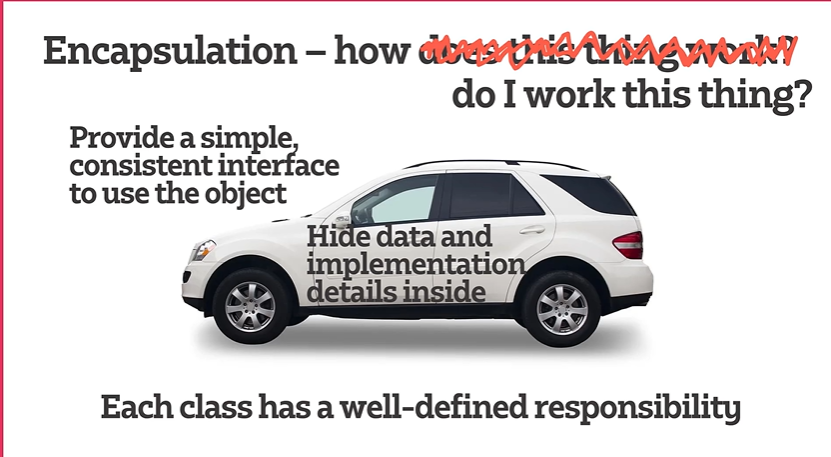
cust.print()

^ there are alt ways of initialising object

Encapsulation:

How does this work?

When u get into car, don’t need to know everything to work it, just need to know how to use:



Coding example:

No control over access to instance variables to class or any objects of that class without this. To stop things from changing, can use decorator e.g. @property

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Inheritance:

Relationship btwn classes

Graphical user interface

Description automatically generated

Aka base class and derived class

Inheritance hierarchy

Coding example:

Can pull up into superclass

Graphical user interface, application

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Polymorphism:

Will have an object and can tell it to do smth by sending it a msg. tell thing to do its job and it will know how to do it.

Graphical user interface, application

Description automatically generated

Coding example:

A screenshot of a computer

Description automatically generated

^ the print functions r polymorphism

Text

Description automatically generated

^another example