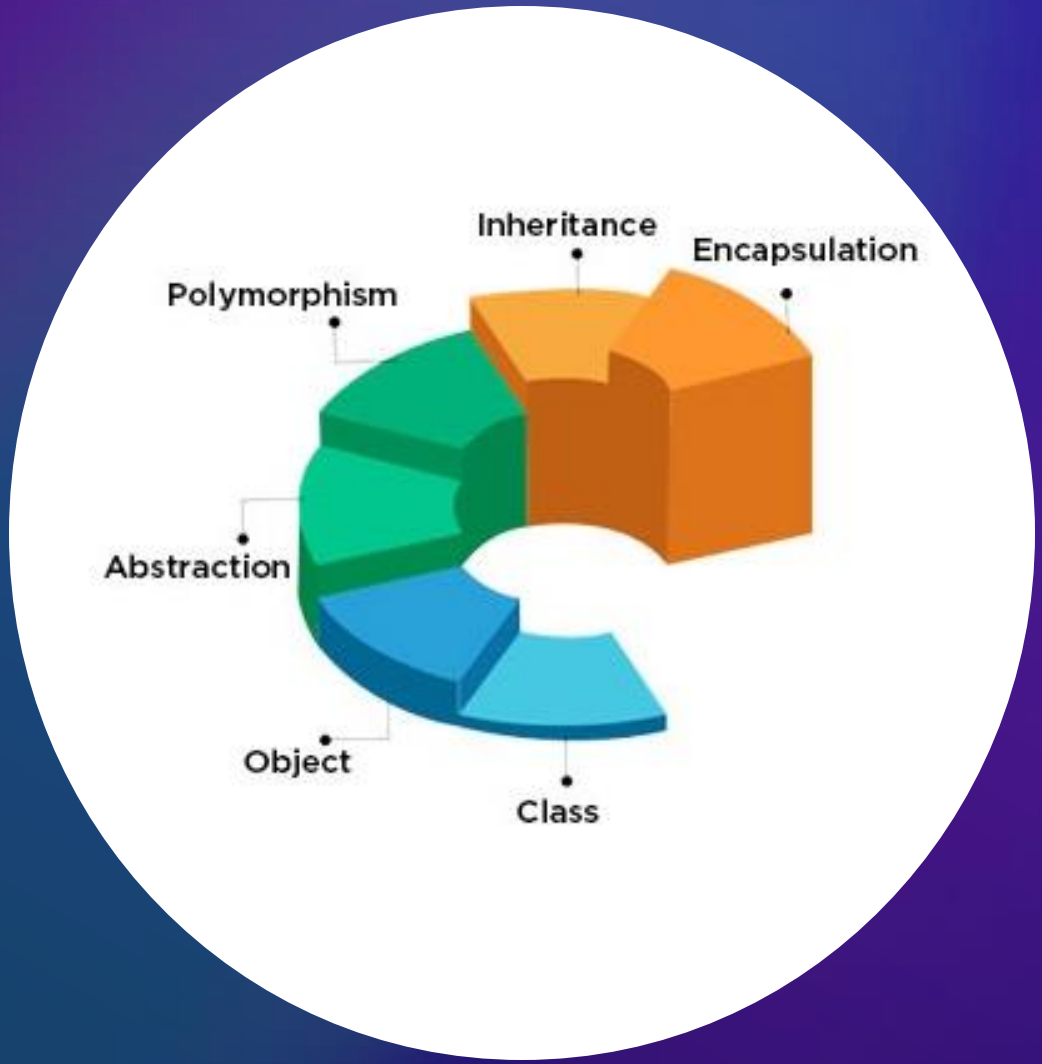
OOPS IN PYTHON

PRESENTATION BY

BHUTHKURI SUCHARITHA



OVERVIEW

- Class
- Object
- Abstraction
- Polymorphism
- Inheritance
- Encapsulation

CLASS

Class Definition Syntax:

```
class ClassName:
   # Statement-1
   .
   .
   .
   # Statement-N
```

1. Classes are created by keyword class.

2. Attributes are the variables that belong to a class

3. Attributes are always public and can be accessed using the dot (.) operator. Eg.: Myclass.Myattribute

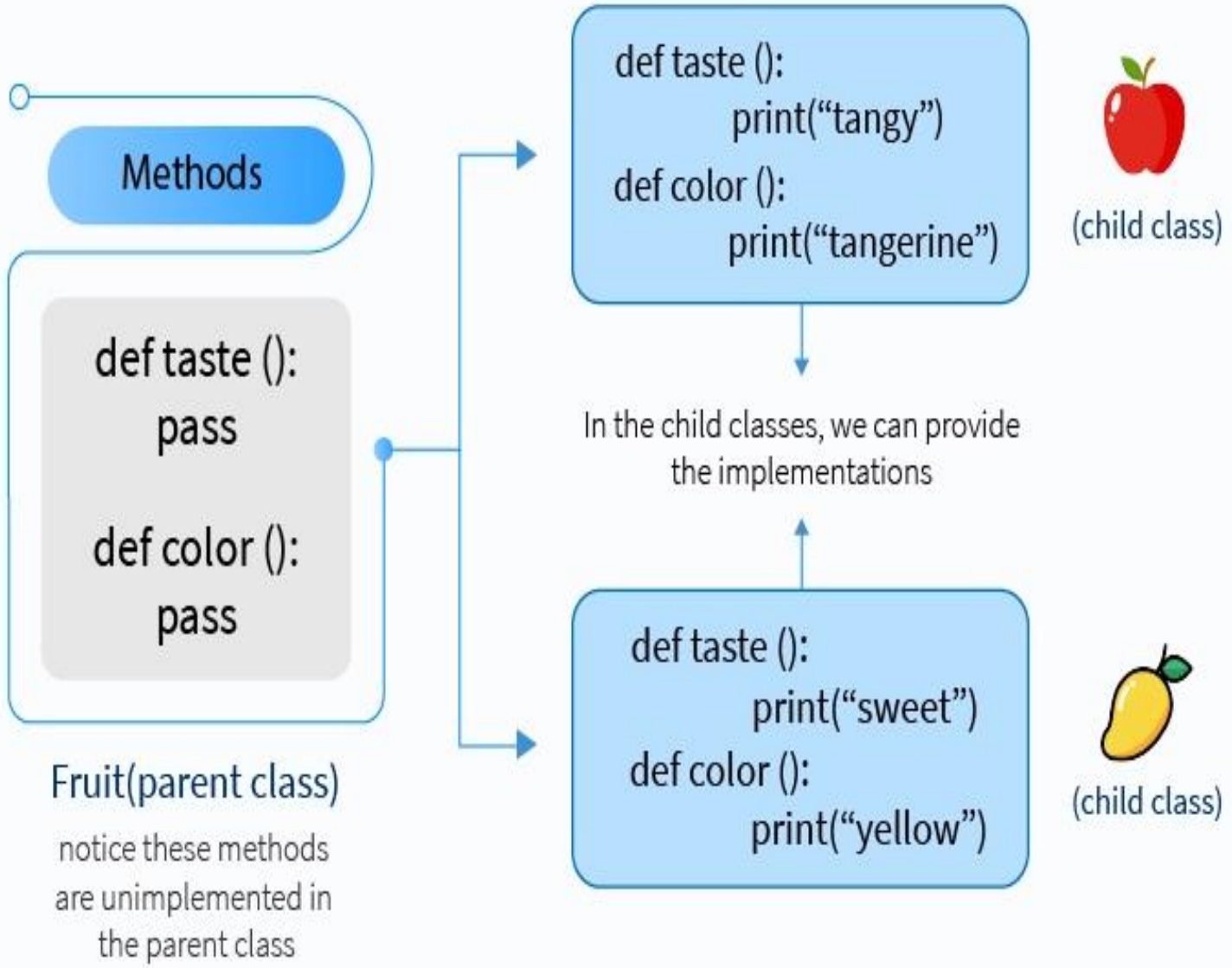
OBJECT

- **An object consists of:**
- **State:** It is represented by the attributes of an object. It also reflects the properties of an object.
- **Behavior:** It is represented by the methods of an object. It also reflects the response of an object to other objects.
- **Identity:** It gives a unique name to an object and enables one object to interact with other objects.

ABSTRACTION

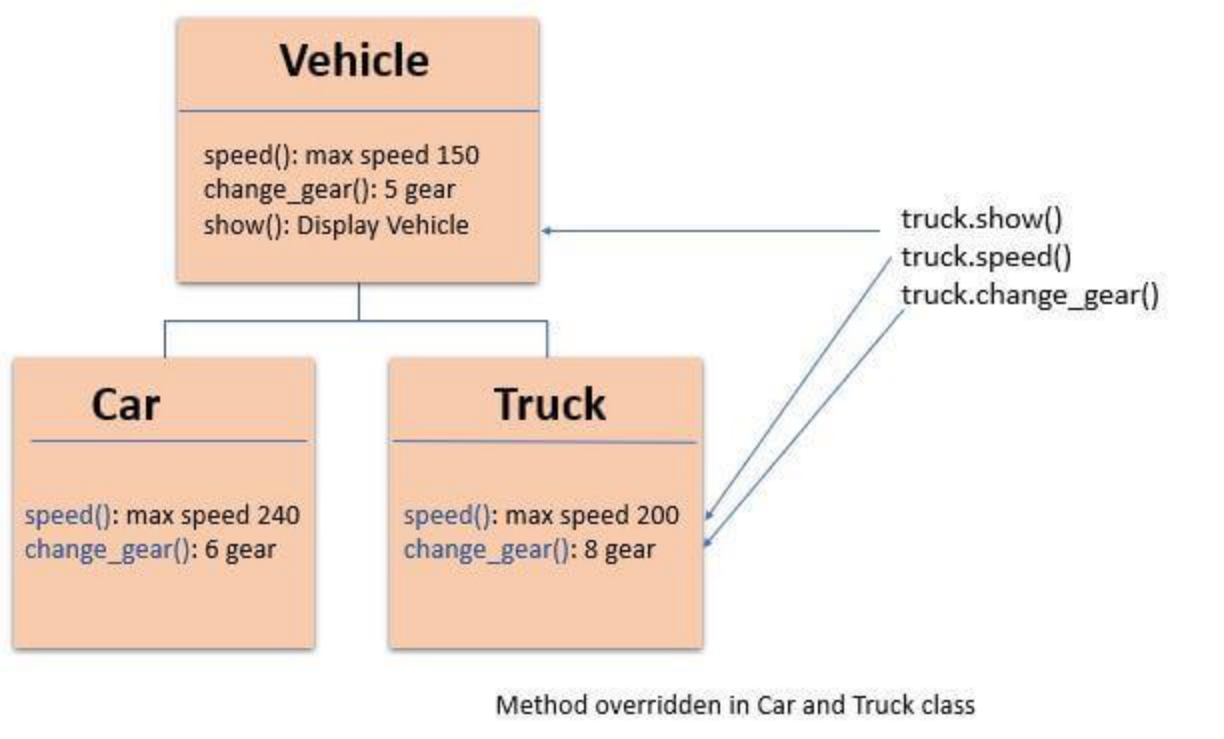
It hide unnecessary code details from the user. Also, when we do not want to give out sensitive parts of our code implementation and this is where data abstraction came.

Data Abstraction in Python can be achieved by creating abstract classes.



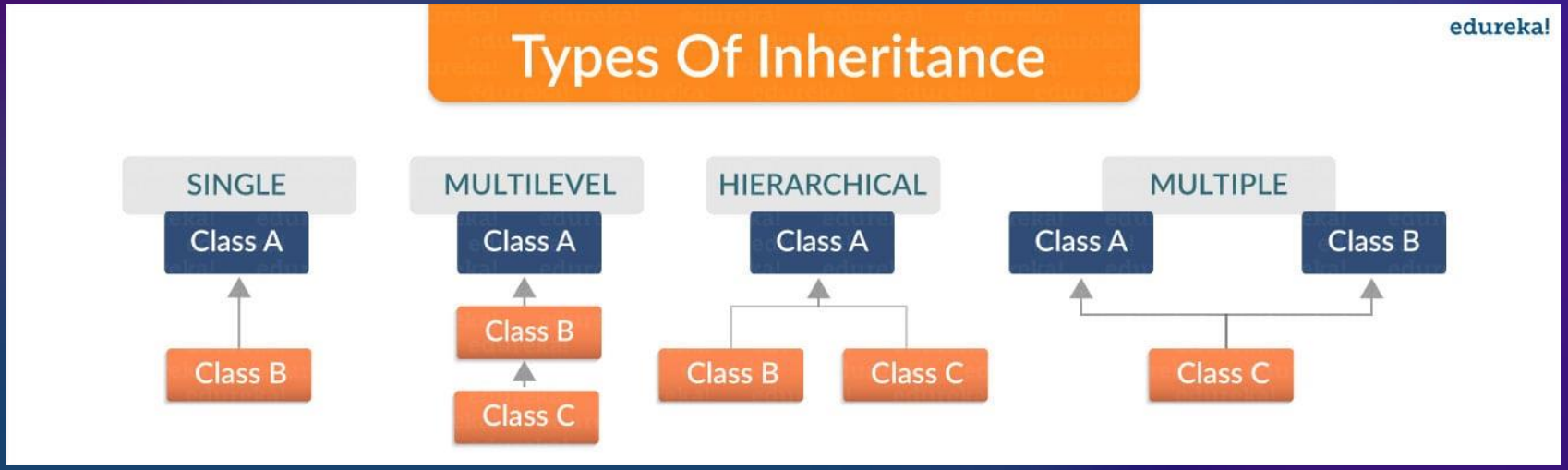
POLYMORPHISM

The concept of inheritance and method overriding in Python classes. It shows how subclasses can override methods defined in their parent class to provide specific behavior while still inheriting other methods from the parent class.

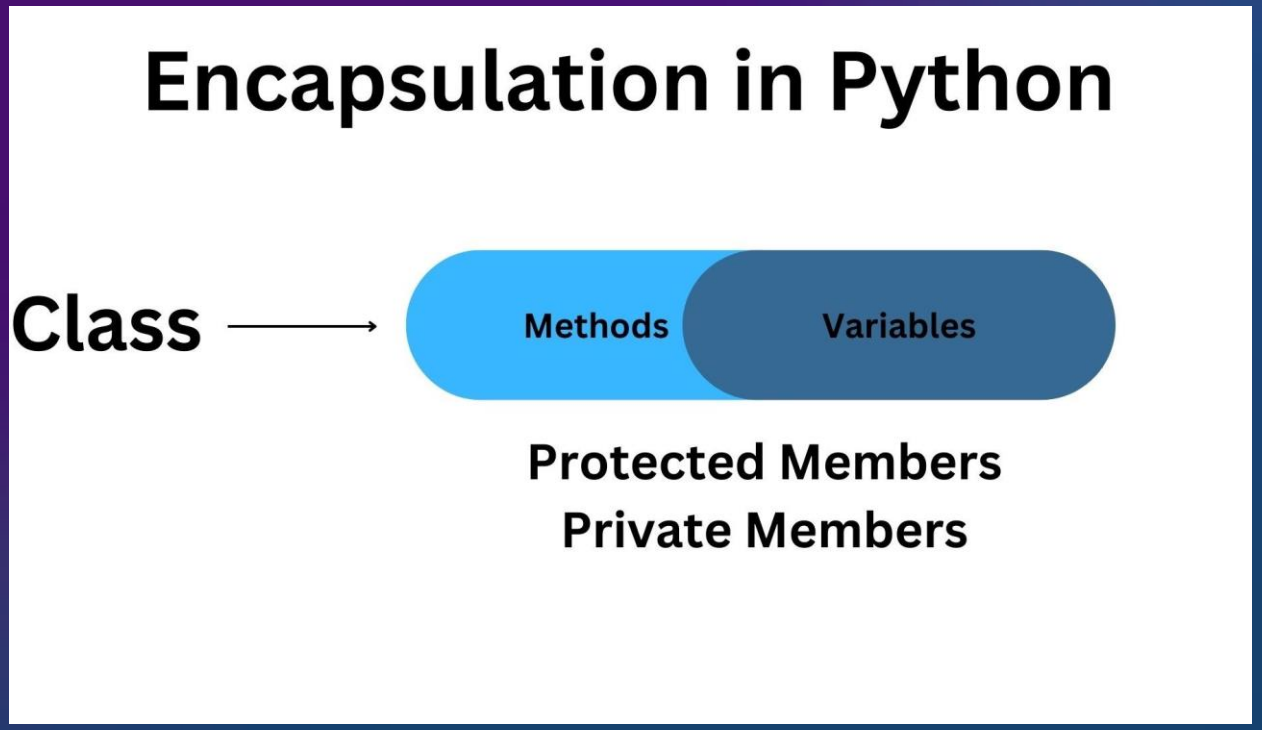


INHERITANCE

- Inheritance is the capability of one class to derive or inherit the properties from another class. The class that derives properties is called the derived class or child class and the class from which the properties are being derived is called the base class or parent class.



ENCAPSULATION



It describes the idea of wrapping data and the methods that work on data within one unit. This puts restrictions on accessing variables and methods directly and can prevent the accidental modification of data.


THANK YOU