BLACK N WHITE

Learn Today Lead Tomorrow

jag....

Polymorphism

In Python

www.blacksnwhite.com

Polymorphism

Polymorphism in Python is the ability of an object to take many forms. It allows us to perform the same action in many different ways.

Built-in polymorphic functions

There are some built-in functions in Python which are compatible to run with multiple data types. Functions such as len(), max() etc. are the polymorphic functions.

The function len() returns the length of an object depending upon its type. If an object is a string, it returns the count of characters. If an object is a list, it returns the count of items in a list. If an object is a dictionary, it returns the count of items (key:value pair) in the dictionary.

Using Polymorphism in len() function

The len() method treats an object as per its class type.

```
students = ['adi', 'tanu',
'jony'] school = 'vnmps'
# calculate count
print(len(students))
print(len(school))
```

It prints 3 for list and 5 for string, thats polymorphism.

Let's take an example

We have a vehicle class as a parent and a 'Car' and 'Truck' as its sub-class.

But each vehicle can have a different seating capacity, speed, etc., so we can have the same instance method name in each class but with a different implementation.

Example

```
class Vehicle:
def __init__(self, name, color, price):
self.name name
self.color color
self.price price
def show(self):
print('Details:', self.name, self.color, self.price)
def max speed(self):
print('Vehicle max speed is 158')
def change_gear(self):
print('Vehicle change 6 gear')
# inherit from vehicle class
class Car(Vehicle):
def max speed(self):
print('Car max speed is 240')
def change gear(self):
print('Car change 7 gear')
#Car Object
 car Car('Car x1', 'Red', 20000)
 car.show()
#calls methods from Cor class
car.max_speed()
car.change gear()
 #Ventele object
vehicle Vehicle('Truck x1', 'white', 75000)
vehicle.show()
#colls method from a Vehtele class
vehicle.max_speed()
vehicle.change_gear()
```

Output:

Details: Car x1 Red 20000
Car max speed is 240
Car change 7 gear
Details: Truck x1 white 75000
Vehicle max speed is 150
Vehicle change 6 gear

Explanation

Due to polymorphism, the Python interpreter recognizes that the max_speed() and change_gear() methods are overridden for the car object. So, it uses the one defined in the child class (Car)

Polymorphism in class method

```
class Ferrari:
def fuel_type(self): print("Petrol")
def max_speed(self):
print("Max speed 350")
class BMW:
def fuel_type (self): print("Diesel")
def max_speed(self):
print("Max speed is 240")
ferrari Ferrari()
bmw BMW()
#iterate objects of same type
for car in (ferrari, bmw):
car.fuel_type()
car.max_speed()
```

We packed two different objects into a tuple and iterate through it using a car variable.

It is possible due to polymorphism because we have added the same method in both classes

```
Petrol
Max speed 350
Diesel
Max speed is 240
```