OOPS

IECT 28 - Part 1

Procedural programming ¶

In [2]:
```
num1 = 40
num2 = 10

sum = num1+num2
diff = num1-num2
print(sum)
print(diff)
```

```
50
30
```

In [3]:
```
num1 = 60
num2 = 50

sum = num1+num2
diff = num1-num2
print(sum)
print(diff)
```

```
110
10
```

functional programming

In [8]:
```
def add_numbers(a,b):
    print(a+b)

def subtract_numbers(a,b):
    print(a-b)
```

In [11]:
```
add_numbers(40,10)
add_numbers(60,5)
```

```
50
65
```

In [10]:
```
subtract_numbers(40,10)
```

```
30
```

Object Oriented Programming

In [21]:
```
class calculator:
    def add(self,a,b):
        print(a+b)
    def diff(self,a,b):
        print(a-b)
    def multi(self,a,b):
        print(a*b)
```

In [22]:
```
calc = calculator()
```

In [23]:
```
calc.add(40,10)
```

```
50
```

In [24]:
```
calc.multi(40,2)
```

```
80
```

In [25]:
```
calc.diff(40,10)
```

```
30
```

class - object - example

In [26]:
```
class Car:

    def start(self):
        print(f"The {self.color} {self.model} car is starting.")

# Object creation aur attributes set karna
car1 = Car()
car1.model = "Toyota"
car1.color = "Red"

car2 = Car()
car2.model = "Honda"
car2.color = "Blue"

# Methods call karna
car1.start()  # Output: The Red Toyota car is starting.
car2.start()  # Output: The Blue Honda car is starting.
```

```
The Red Toyota car is starting.
The Blue Honda car is starting.
```

init function

init_ Function kya hota hai?

init function ek special method hai jo class ke object create hone par automatic call hota hai. Is function ka kaam hai object ke initial properties ko set karna. Is function ko constructor bhi kehte hain.

__init__ Function kyun banate hain?

Initialization: Object ke properties ko initial values dene ke liye.
Automatic Call: Jab bhi object banate hain, **__init__** function automatic call hota hai.
Consistency: Sab objects ko consistent initial state dene ke liye.

In [27]:
```
class Car:
    def __init__(self, model, color):
        self.model = model
        self.color = color

    def start(self):
        print(f"The {self.color} {self.model} car is starting.")
```

In [30]:
```
car1 = Car("toyota", "black")
```

In [31]:
```
car1.start()
```

```
The black toyota car is starting.
```

In [ ]:
```

```