OOP Inheritence

In [ ]:
```
#base class
class Person:
    def __init__(self,name,age):
        self.name = name
        self.age = age
    def who(self):
        print("I am " + self.name + " and I am " + str(self.age))

paul = Person("Paul",48)
paul.who()
```

```
I am Paul and I am 48
```

In [ ]:
```
#inherited class
class Student(Person):
    def __init__(self, name, age, course):
        super().__init__(name, age)
        self.course = course
    def study(self):
        print("I study " + self.course)

bob = Student("bob", 18, "computing")
bob.who()
bob.study()
```

```
I am bob and I am 18
I study computing
```

In [ ]:
```
#inherited

class Staff(Person):
    def __init__(self, name, age, wages):
        super().__init__(name, age)
        self.wages = wages
    def pay(self):
        print("I earn £" + str(self.wages))
```

```
richard = Staff("Richard", 26, 89900)
richard.who()
richard.pay()
```

```
I am Richard and I am 26
I earn 89900
```

In [ ]:

```
richard.study()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_14300/2639445617.py in <module>
----> 1 richard.study()

AttributeError: 'Staff' object has no attribute 'study'
```