Python Classes

https://www.w3schools.com/python/python_classes.asp

```
# Create the class
class MyClass:
  x = 5

# Create the object
p1 = MyClass()
```

init() functions are created automatically when the class is created. It is always called when the class is initialized.

```
# Example __init__ function
class Person:
  def __init__(self, name, age):
    self.name = name
    self.age = age

p1 = Person("John", 36)

print(p1.name)
print(p1.age)
```

```
John
36
```

str() function says what should be returned when the class is called

WITHOUT **str** , the below returns this: `<__main__.Person object at 0x0000023B8573B430>` WITH it, it returns this: `John(36)`

```
class Person:
  def __init__(self, name, age):
    self.name = name
    self.age = age

  def __str__(self):
    return f"{self.name}({self.age})"

p1 = Person("John", 36)

print(p1)
```

```
John(36)
```

Object Methods are functions that belong to the object. They are defined in the class

```
class Person:
  def __init__(self, name, age):
    self.name = name
    self.age = age

    # self refers to current class and refers to vars in that class

  def myfunc(self):
    print("Hello my name is " + self.name)

p1 = Person("John", 36)
p1.myfunc()
```

```
Hello my name is John
```

self refers to the current clas and the vars inside that class. It is the first var listed in the functions within a class

```
class Person:
  def __init__(mysillyobject, name, age):
    mysillyobject.name = name
    mysillyobject.age = age

  def myfunc(abc):
    print("Hello my name is " + abc.name)

p1 = Person("John", 36)
p1.myfunc()
```

```
Hello my name is John
```

```
# items in the class can be modified or deleted

# change the age to 40
p1.age = 40

# delete the age of p1
del p1.age

# delete entire p1 object
del p1

# How to have an empty class
class Person:
  pass
```