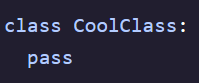
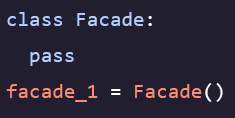
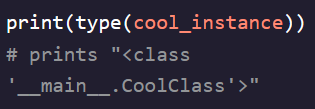
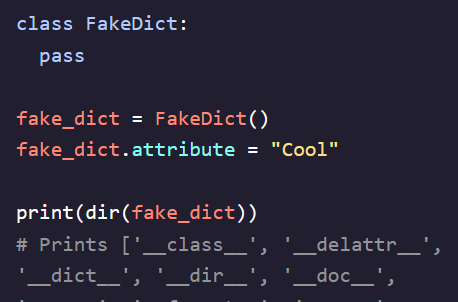
**Class:**

- Different ways to store data in Python (int, float, list, dict, str)  
- Can check what type something is by using the *type()* function  
- A ***class***is a template for a data type – used to define the kinds of information that class will hold and how a programmer will interact with that data  
- Capitalize the names of classes to make them easier to identify  


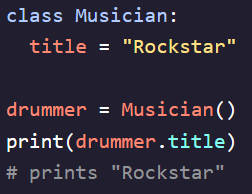
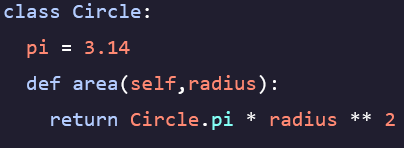
- A class must be *instantiated* before it can be used – this means an instance of the class needs to be created  
- A ***class***is *instantiated* by calling it like a function – can store to a variable to access later in program  
 

**Object Oriented Programming (OOP)**

- The pattern of defining classes and creating objects to represent the responsibilities of a program  
- Encourages sculpting entities with properties and methods names in classes to create applications  
- Instantiation takes a class and turns it into object  
- Classes that have been instantiated in code become objects that can interact with one another to perform the desired functions of the application   
- When calling *type()* on an object it returns the class of that object   
- Attributes can be added to user-defined objects after instantiation – possible to have some attributes that are not explicitly defined in an objects constructor  
- Can use the *dir()* function to investigate an objects attributes at runtime   
- Short for *directory* and offers an organized presentation of object attributes  
 

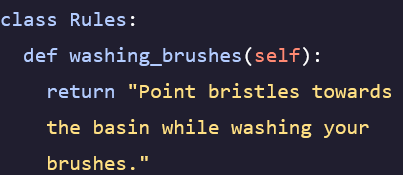
- ***\_main\_*** indicates that this is the current running file  
- Python automatically adds attributes to all created objects

**Class Variables**

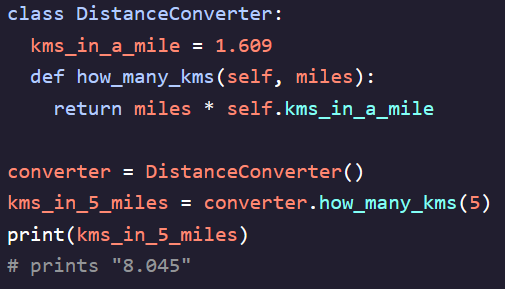
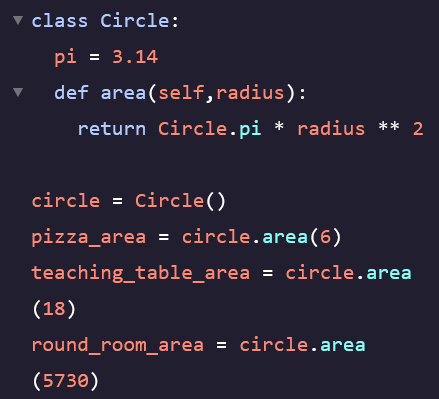
- Employed when we want the same data to be available to every instance of a class  
- Defined by including it in the indented part of your class definition  
- Can access and see all of an object’s class variables by using the *object.variable* syntax  
- Classes are referenced with dot notation (*.title*) to show that they belong to a class  
 

Class variables are the variables stored in the ***class*** template   
- notated as *.pi  
- Can be accessed as an attribute of the class*  
(dog\_time\_dilation, kms\_in\_a\_mile)

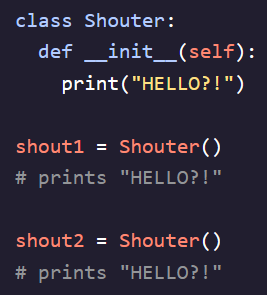
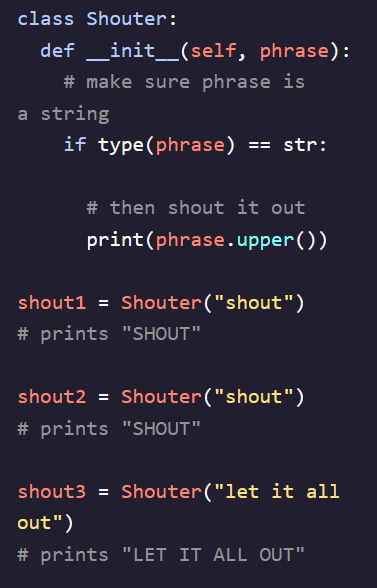
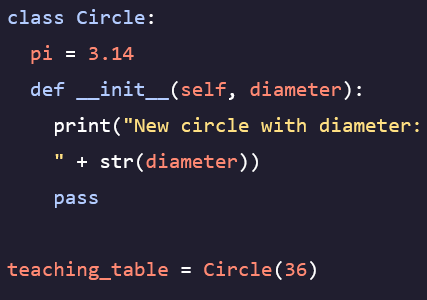
**Methods**

- Functions that are defines as part of a class  
- The first argument in a method is always the object that is calling the method – named *self*- Methods need at least this one function and are defined just like functions except within a ***class***  


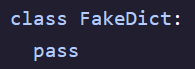
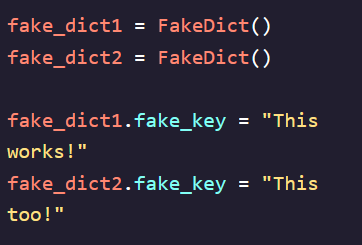
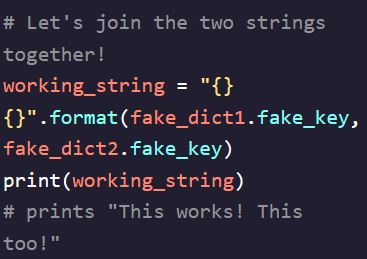
Methods are the functions stored in classes|  
- notated as *.area()* method  
(time\_explanation, how\_many\_kms)

- Methods can take other arguments besides just self  
 

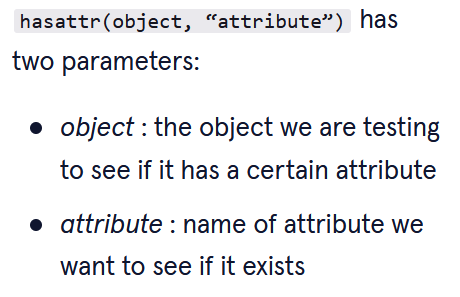
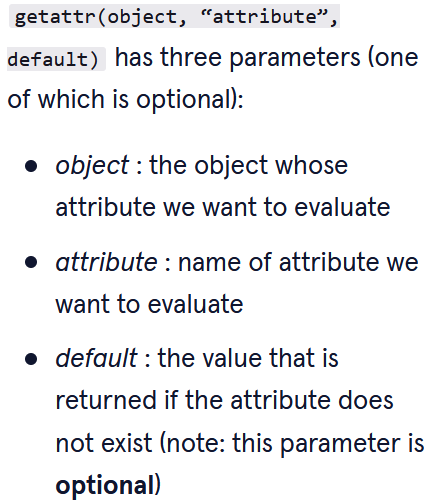
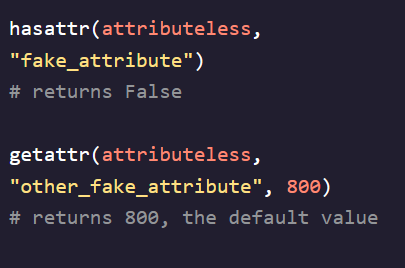
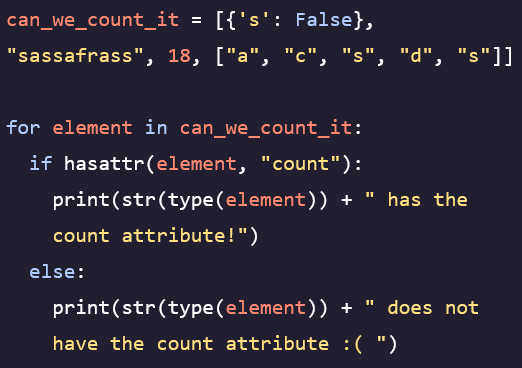
**Constructors**

- ***Dunder Methods*** – Methods with a double underscore that use special syntax to perform class specific operations  
- *\_\_init\_\_()* – This method is used to initialize a newly created object  
- Methods that are used to prepare an object being instantiated are called *constructors*   
  

**Instance Variables**

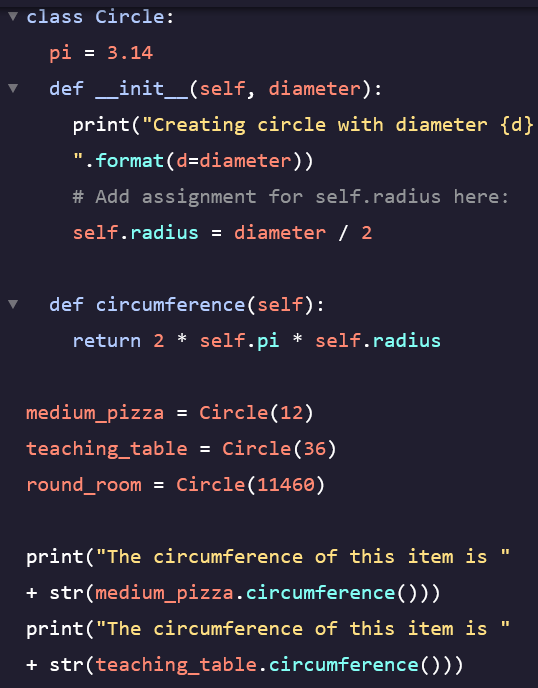
- The data held by an object  
- Not shared by all instances of a class but are specific to the object they are attached to  
- Not instantiated within ***class***- More powerful when you can guarantee a rigidity to the data the object is holding   
  

**Attribute Functions**

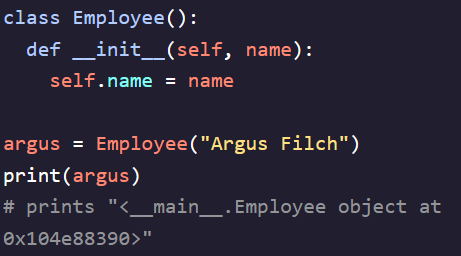
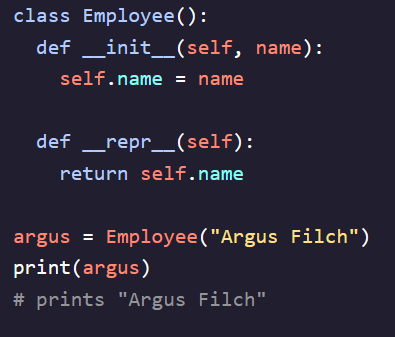
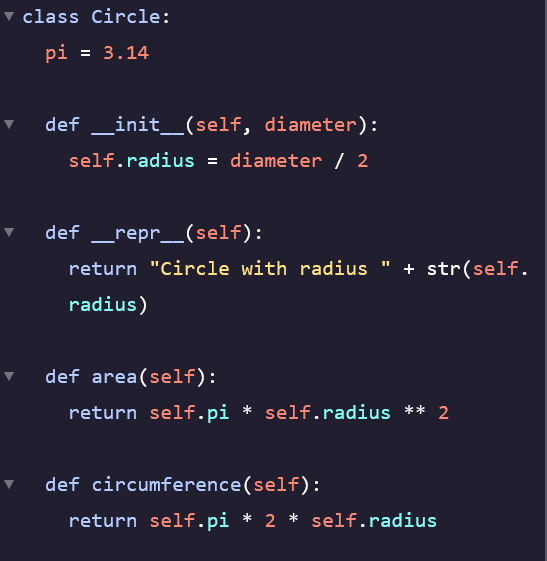
- *hasattr(object, “attribute”)* – returns True if an object has a given attribute and False otherwise  
- *getattr(object, “attribute”, default)* – returns the value of a given object and an attribute – can supply third argument to be the default if object doesn’t have the given attribute  
    


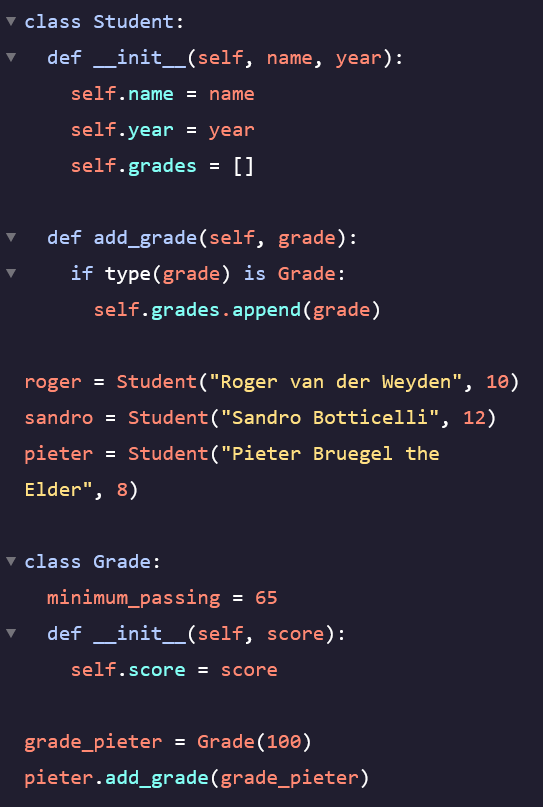
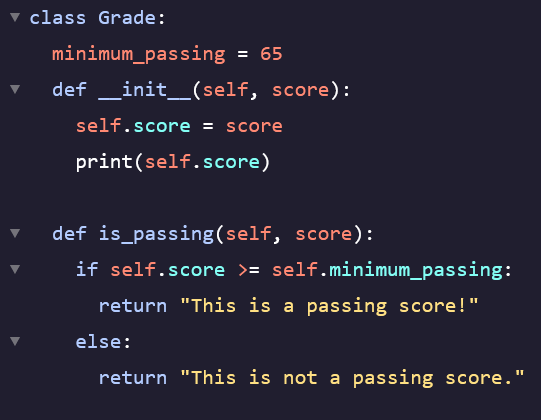
**Self**

- This keyword refers to the object and not the class being called  
- Allows us to write our classes to structure data how we want and write methods that interact with data in a meaningful way

**String Representation**

- Default string representation gives us some information like where the class is defined and our computers memory address  
- *\_\_repr\_\_()* – Used to tell Python what we want the *string representation* of the class to be  
- Can only have one parameter *self* and must return a string  
  

Added a passing grade check method