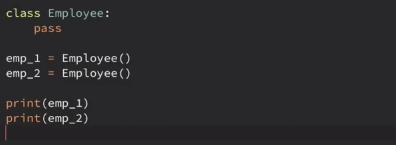
**Classes and Instances**

Classes help group data and functions (attributes and methods respectively) in a way that can be reused and built upon later

Class is basically a blueprint for creating instances and each unique employee created using the class is called an instance of the class.

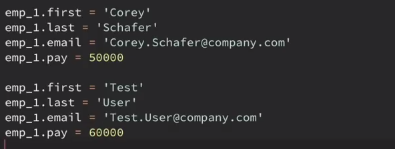
Example:



emp1 and emp2 are instances of class Employee (with no attributes or methods). They are both Employee objects, unique and stored in different locations in memory.

There is huge difference between class variables and instance variables. Instance variables contain data that is particular/unique to that instance.

Example:

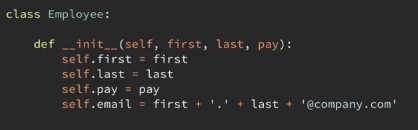


Creating each instance uniquely is difficult and prone to errors. Hence we will use the \_\_init\_\_() method in the class.

Now when creating methods within a class they receive the instance as the first argument automatically. Hence by convention it is called self.

After creating self we can create as many arguments as needed.

Example:



Now we can create as many instances as required:



The variables first, last and pay are attributes of the class Employee.

Now in order to perform some action, a method has to be created.

Example: creating a method to print full name within the class:





Now let us see a case when the self variable is left out:



Throws a typeerror.

Check out the following case where the full name is printed in two different ways but returns the same value:



* When calling the method with respect to a particular instance of the class, we do not need to pass any argument
* But in the second case, when the method is called using the class Employee, it does not know which instance it is referring to. Hence it requires an argument.