**https://github.com/999Path/Coffee-Maker**

## DESCRIPTION OF THE CLASS

---

The coffee machine class purpose is to take in inputs that the user wants to make coffee and print out the recipe before it continues with the production of the coffee. There would be a default value for each input, but the user is allowed to manually edit the recipe to their liking. The __init__ constructor allows the class to initialize the attributes of the class, which are self.shots,self.types,self.__milk_foam, and self.__steam_milk.

## DESCRIPTION OF EACH OF THE CLASS AND DATA VARIABLES

---

self.shots = an int/float variable: the number of shots the user wants. Not including the default shots.

self.type = a string variable: the type of coffee the user wants. The default would be set according to the type input.

self.__steam_milk = an int/float variable : the amount of steam milk the user wants. Not including the default.

self.__milk_foam = Boolean variable: True or False to milk foam existing in the drink.

hot_drink = True, class variable since all drink will come out hot.

## DESCRIPTION OF EACH OF THE METHODS

---

**__int__** method take in the identifier and store it. However, the input for shots should be and int/float. If the input is a string, the program will set the shots input to default. The input for types should be a string with the option of “Latte”, “Espresso”, “Espresso Macchiato”, “Latte” if something else, there would be no default input.

**venti_size** method double the int/float of self.shots and self.__steam_milk. There’s no input needed to call this method.

**grande_size** method divide the self.shots and self.__steam_milk in half. There’s no input needed to call this method.

**set_foam** method, sets the foam amount to the input of the user. The input can only be true or false, if anything else the program will prompt the user to reenter the input.

**set_steam_milk** method sets the steam amount to the steam input of the user. If steam is less than zero, program will prompt the user to change the input. If the steam is more than or equal to zero, it will change self.__steam_milk to steam. If the input is something else, it will prompt the user to change the input to an int or a float.

**get_steam_milk** method returns the amount of steam milk

**get_foam** method returns the amount of milk foam

**__str__** is a magic method, that should return a string representation of the coffee that's making.

**DESCRIPTION OF DEMO PROGRAM**

---

Latte is set as coffee_machine with the input "Latte" and "Lol". The output is "Shots cannot be a string, or else it will be set to default" since "Lol" isn't an integer. Print(latte) would give out: Making: Type - Latte (Ingredients: shots - 1, Milk foam- True, Steam milk - 2). Printing for the user to see what type of coffee the machine is making. On line 83, none is set as coffee_machine("None",2). Meaning that the type is none of the default type, meaning that there's no default input for the shot amounts and steam milk. Line 83 wants to change the steamed_milk, by calling the set_steamed_milk method to -1, however since the program doesn't take a negative input it will prompt the user to change the input. Line 84 calls the method set_foam and change it to True. Line 85 print the what the coffee_machine is making. Line 86 calls the venti_size to double the none shots and steamed milk. Line 87 print what the program is making.

**HOW TO RUN THE DEMO PROGRAM**

---

The user don't have to do anything to run the demo program since it's already running in the main() function and it's already tested to run automatically. However, the user can modify the input that's inside the main function. Like latte = coffee_machine("Latte", 6) or latte = coffee_machine("Latte", LMAO).