Object Oriented Python

In project2 you have to write an expense tracker program.

There is a file Budget.py that is given to you that reads a file budget.txt in the following form.

```
"BudgetType","Max Amount"
"Meals","$300.00"
"Car","$200.00"
```

lines = [["Meals","300"]
[ ]
]

list = lines[i].split(",")

→ list → ["Meals","$300.00"]

list[0]

exptype = list[0].strip('"\n')

→ Meals

removes quotes, spaces...

maxAmount = list[1].strip('$\n') # "$300.0"

def printBudget(budget):
    #print header

print("Type".ljust(12), "Max Amount".ljust(12)

"Type" - 12 spaces - "Max Amount" - 12

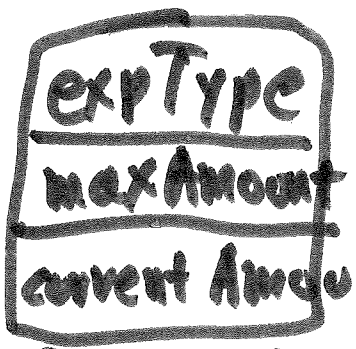
This budget.py uses a list called "budget" of dictionaries of the form

{'exptype': exptype, 'maxamnt': maxamount}

We will re-write "budget.py" now using objects instead of dictionaries.

We we call this budgetObj.py.

Each object will have three attributes



- expType → expense Type
- maxAmount → max amount for this type
- current Amount → current amount initialized to 0

The budget list will be an array of objects of type BudgetObj

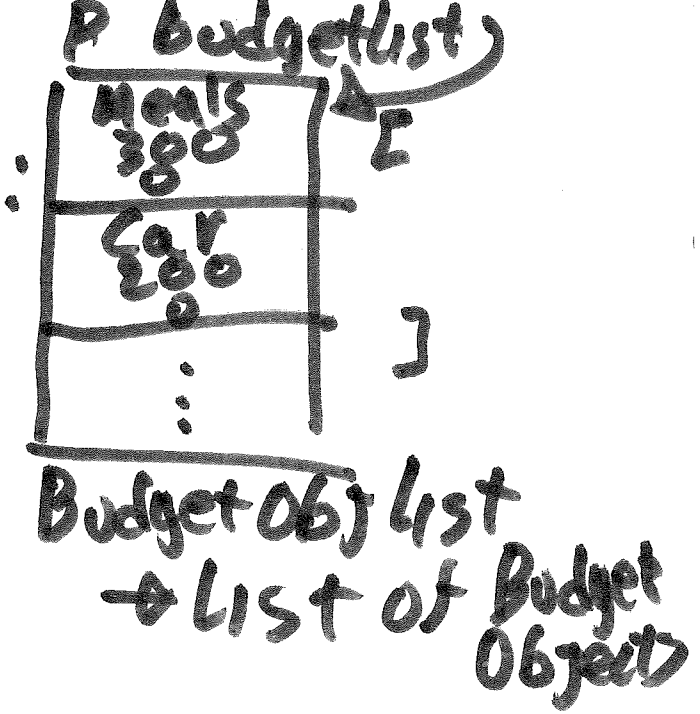
BudgetObj.py


```
class BudgetObj:
    def __init__(self, expenseType, maxAmount):
        self.exptype = expenseType
        self.maxAmount = maxAmount
        self.currentAmount = 0

    def getExpenseType(self):
        return self.exptype
    def getMaxAmount(self):
        return self.maxAmount
    def getCurrentAmount(self):
        return self.currentAmount
```

```
    def setCurrentAmount(self,currentAmt):
        self.currentAmount = currentAmount
    def print(self):
        print(self.expType.ljust(12),
              str("$%.0.2f" % self.maxAmount))

class BudgetObjList:
    def __init__(self,budgetFile):
        #read budgetfile and
        # add entries into BudgetObj
        # in the list.
        #read file
        f = open(budgetFile)
        lines = f.readlines()
        f.close()
```



```
# Initialize budgetlist with empty list
self.budgetlist = []

# Parse the lines
for i in range(len(lines)):
    # Split fields in line
    list = lines[i].split(",")

    # Read exptype
    exptype = list[0].strip('"\n')
    if exptype == "Type":
        # Skip header of budget.txt
        continue

    # read maxamount
    maxamount = list[1].strip('$"\n')
    # Create Budget obj
    b = BudgetObj(exptype, maxamount)
```

budgetType, Maxamount
"$300.0"

BudgetObj
budgetlist

obj1
obj2
.
.

```
        self.budgetlist.append(b)

    #end of constructor

def print(self): #print BudgetObjList:
    print()
    print("---- Budget ----")
    print("Type".ljust(12), "Max Amount".ljust(12))
    #print all entries in budgetlist
    total=0
    for b in self.budgetlist:
        #print this budget entry.
        b.print()
        total=total+b.getMaxAmount
    print("Total=", "$%0.2f" % total)
def Main():
    budgetlist = BudgetObjList("budget.txt")
    budgetlist.print()
```

```
# call main only if we run budget.py directly
if __name__ == __main__:
    main()
```