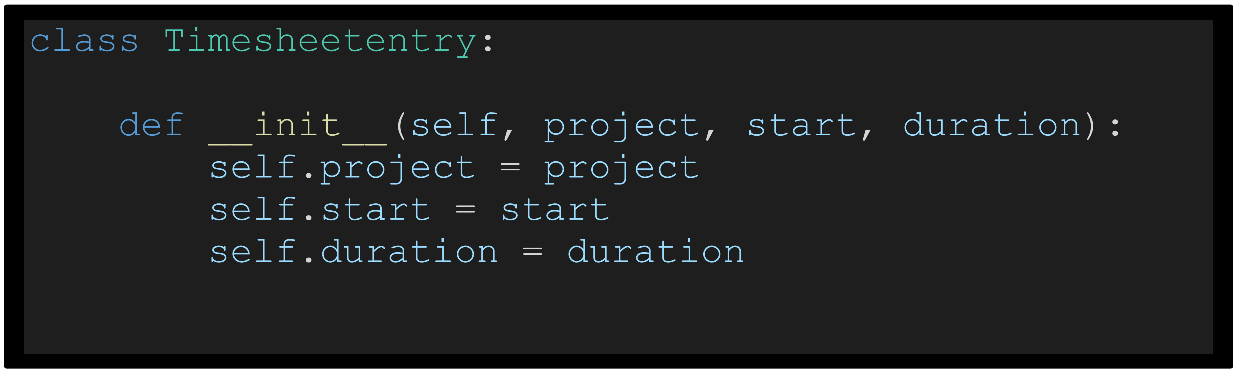
Programming and Scripting
Lab 10- Objects

Introduction.

I would suggest that you create another folder in labs called **Topic10-Objects**

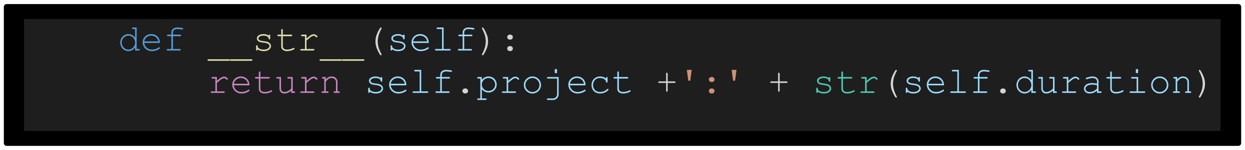
Create a class

1. In a file called timesheetentry.py, create a class called Timesheetentry that has three attributes that are all set by an __init__ function
    a. Project
    b. startTime (a datetime object)
    c. duration (in minutes)



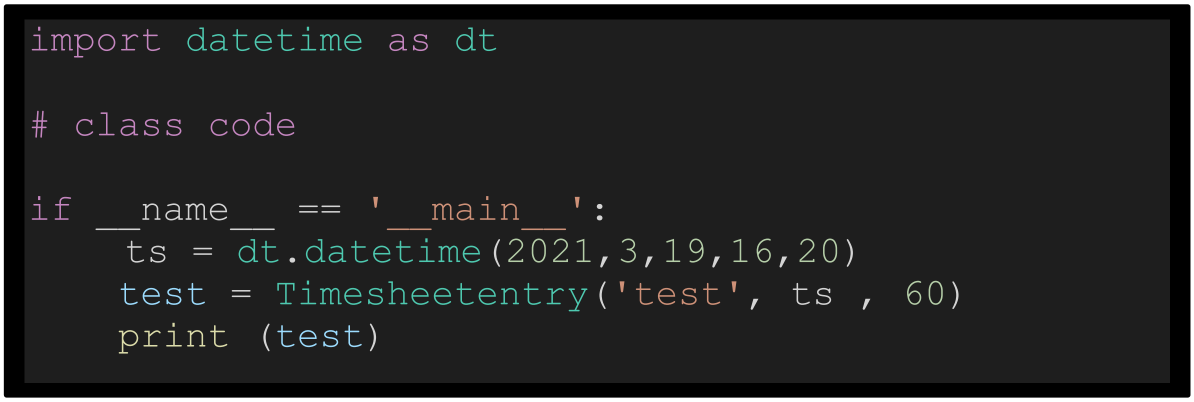
```
class Timesheetentry:

    def __init__(self, project, start, duration):
        self.project = project
        self.start = start
        self.duration = duration
```

2. Write a __str__ function for this class that returns the project and the duration.



```
    def __str__(self):
        return self.project +':' + str(self.duration)
```

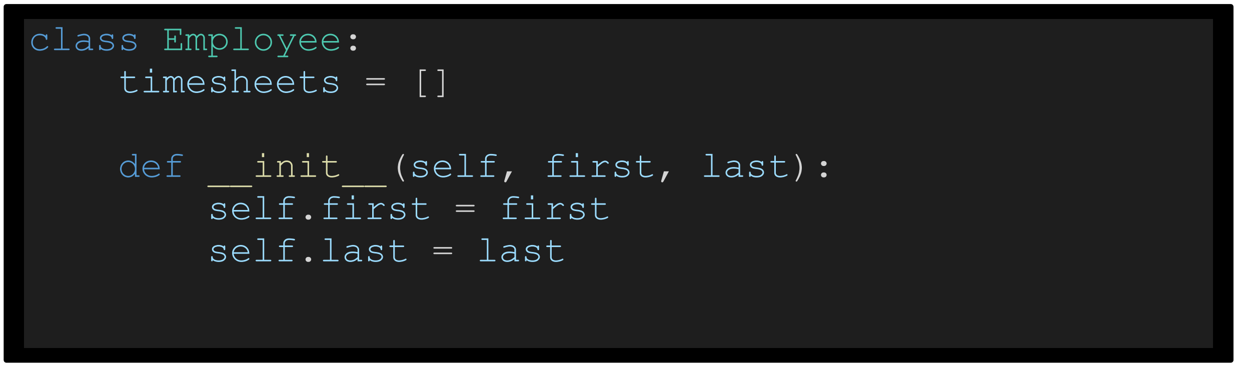
3. Write a test case for this class that creates an instance and prints it out



```
import datetime as dt

# class code

if __name__ == '__main__':
    ts = dt.datetime(2021,3,19,16,20)
    test = Timesheetentry('test', ts , 60)
    print (test)
```

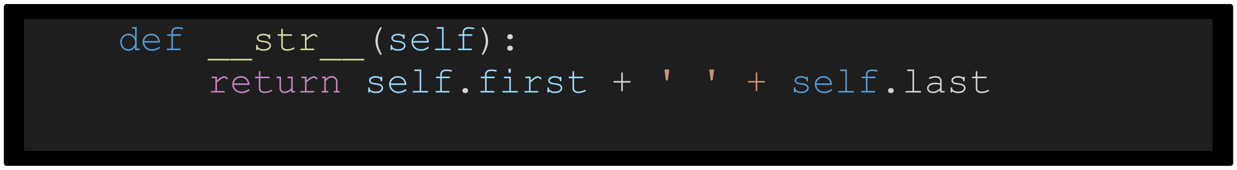
Employee

4. Write a class called Employee (in employee.py), that has one attribute called timesheets (set to an empty list) and an __init__ function that takes in a first and last name.



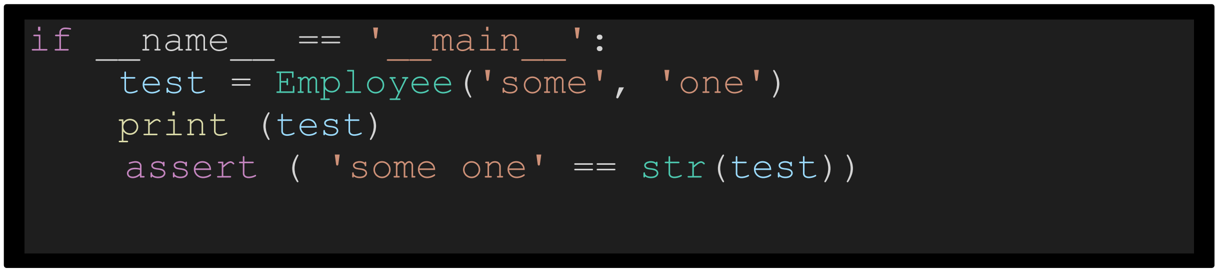
```
class Employee:
    timesheets = []

    def __init__(self, first, last):
        self.first = first
        self.last = last
```

5. And write a __str__ function that returns the full name,



```
    def __str__(self):
        return self.first + ' ' + self.last
```

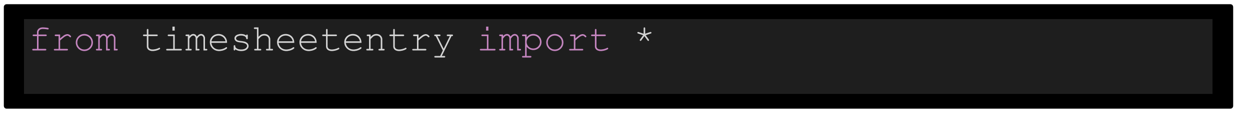
6. Write a testcase.



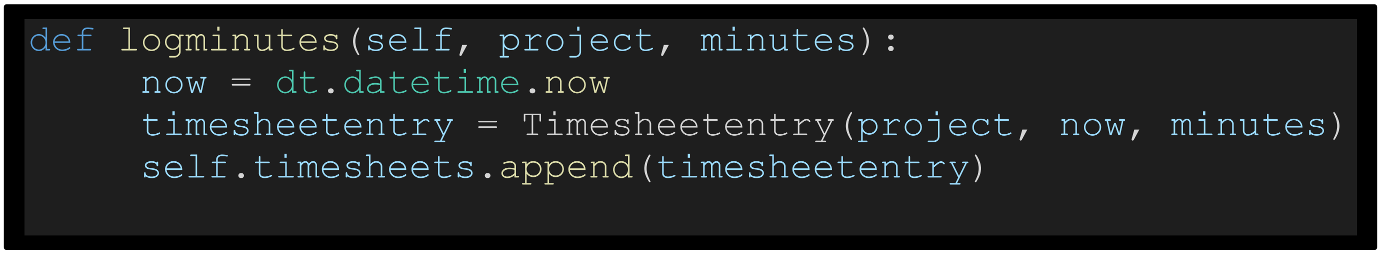
```
if __name__ == '__main__':
    test = Employee('some', 'one')
    print (test)
    assert ( 'some one' == str(test))
```

Creating object in code

7. Write a function in the class Employee called logminutes(project, minutes), the function will create a Timesheetentry with the project and minutes and the time being now (it probably should be now minus minutes), and append it to the list

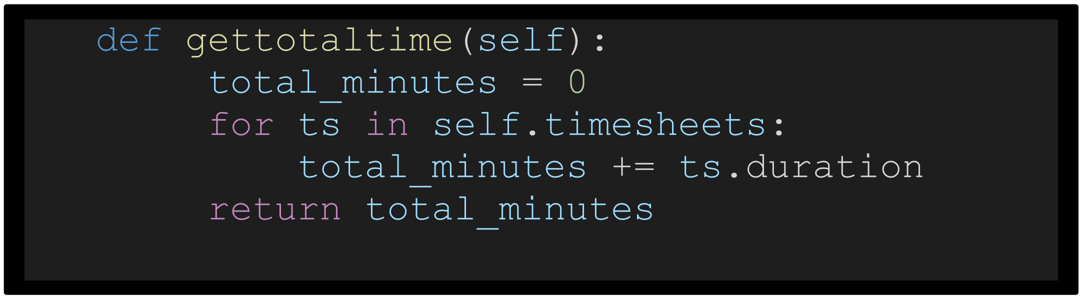


```
from timesheetentry import *
```



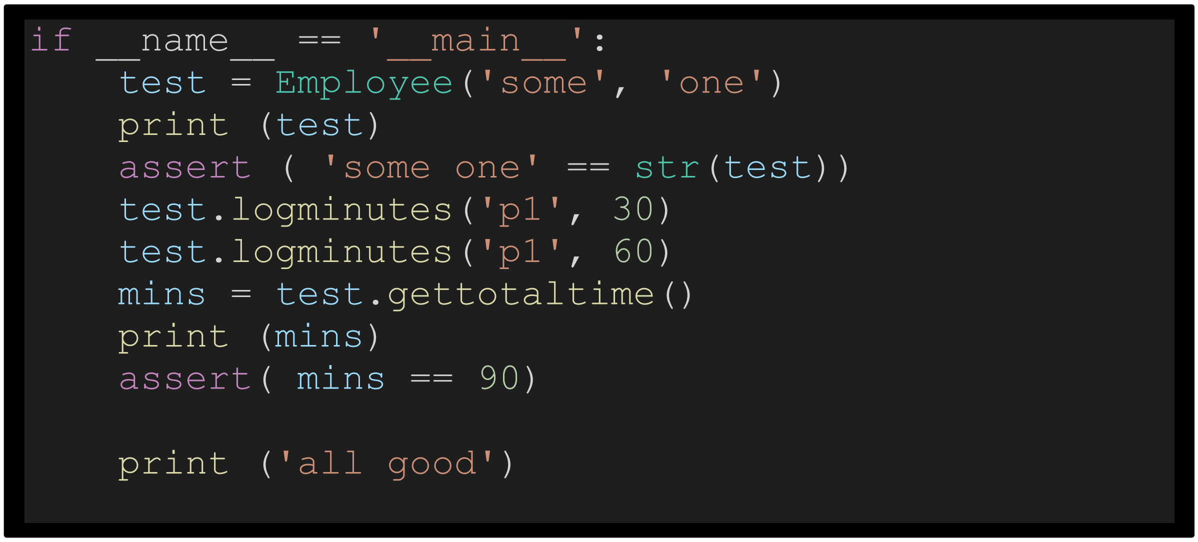
```
def logminutes(self, project, minutes):
     now = dt.datetime.now
     timesheetentry = Timesheetentry(project, now, minutes)
     self.timesheets.append(timesheetentry)
```

8. Write a function called gettotaltime() that will return the total minutes of all the timesheetentrys



```
   def gettotaltime(self):
        total_minutes = 0
        for ts in self.timesheets:
            total_minutes += ts.duration
        return total_minutes
```

9. Write a test case for this.



```
if __name__ == '__main__':
    test = Employee('some', 'one')
    print (test)
    assert ( 'some one' == str(test))
    test.logminutes('p1', 30)
    test.logminutes('p1', 60)
    mins = test.gettotaltime()
    print (mins)
    assert( mins == 90)

    print ('all good')
```

C’est FIN