

# Assignment 3

See D2L for Due Date

-20 points taken off if who you worked with or “worked alone” not specified. Last warning.

## Reading

Review Chapter 8 in **Introduction to Computing using Python: An Application Development Focus, Second Edition** by Ljubomir Perković.

## Logistics

In this class programming assignments may be completed in consultation with up to two other classmates. You must identify the classmates with whom you collaborate in a comment at the top of the assignment, and the number of collaborators on any assignment **may not exceed two other people**. You must also submit a comment in your submission for each assignment that describes in detail how each collaborator contributed to the assignment. If you did not collaborate with anyone on the assignment, you must include a comment that says that. You may not under any circumstances discuss the assignments with classmates other than your identified collaborators. Working so closely with anyone other than your identified collaborators, Mr. Zoko, or lab assistant, so as to produce identical or near identical code is a violation of the Academic Integrity policy. This policy will be strictly enforced.

Please include the following with your assignment submission:

1. A comment at the top of your Python file identifying any classmates with whom you discussed or in any other way collaborated on the assignment. You may work (directly or indirectly) with **no more than two** other people.
2. Add a comment at the top of your Python file that describes for each person what they contributed to the assignment. This must be at least 2-3 sentences and be **very specific and detailed**.

A submission that does not include a list of collaborators and a comment indicating how you collaborated with classmates will earn a 0. If you worked alone, you must put a comment at the top of your file that indicates that. There will be no exceptions to this rule.

# Assignment

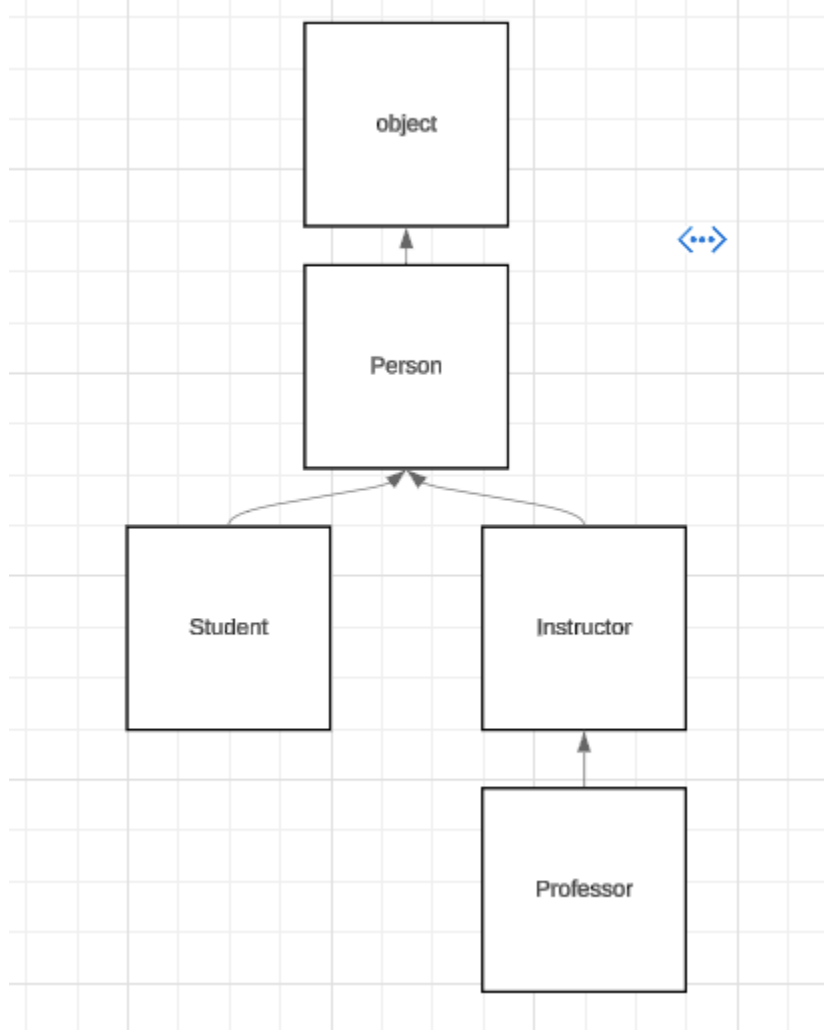
Begin the assignment by downloading the template **csc242hw3.py** found on the D2L site. Each problem is worth 25 points. **No template is being provided. You will write this code from scratch. Look at the screenshots carefully! Points will be taken off if you override methods that don't need to be overridden.**

## Objectives

- 1) Reuse as much code as possible with Inheritance. Points will be taken off for unnecessary duplication
- 2) Use of assert and Class Invariants to maintain consistent state of an object when its being created and modified
- 3) Repr must behave consistently with type and be able to create copies using eval.
- 4) Ensure you understand all the new syntax related to Classes covered in weeks 2 and 3

## Inheritance Tree

**Your code must be implemented using the following inheritance tree:**



## Problem 1: Person Class

### General Information

Write a Person class has the following characteristic and behaviors:

- Name: a name is required and must be a string with length greater than 0.
- Age: is an integer and must be greater than 0
- Email: is optional but must be a string. In other words, can be an empty string

---

```
p=Person()
Traceback (most recent call last):
  File "<pyshell#1>", line 1, in <module>
    p=Person()
TypeError: Person.__init__() missing 2 required positional arguments: 'name' and 'age'
p=Person('',-1)
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    p=Person('',-1)
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 7, in __init__
    assert type(name)==str and len(name)>0, 'Name must be a string with length greater than 0'
AssertionError: Name must be a string with length greater than 0
p=Person('Bob',-1)
Traceback (most recent call last):
  File "<pyshell#3>", line 1, in <module>
    p=Person('Bob',-1)
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 9, in __init__
    assert type(age)==int and age>0 , 'Age must be an integer and greater than 0'
AssertionError: Age must be an integer and greater than 0
p=Person('Bob',1,1)
Traceback (most recent call last):
  File "<pyshell#4>", line 1, in <module>
    p=Person('Bob',1,1)
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 8, in __init__
    assert type(email)==str , 'Email must be a string'
AssertionError: Email must be a string
```

```

p=Person('Bob',1)
str(p)
'Person Bob is 1 years old.  Their email address is '
p
Person('Bob',1, '')
id(p)
1926092248352
p2=eval(repr(p))
id(p2)
1926092249936
p2
Person('Bob',1, '')

```

```

p.setName('')
Traceback (most recent call last):
  File "<pyshell#16>", line 1, in <module>
    p.setName('')
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 20, in setName
    assert type(name)==str and len(name)>0, 'Name must be a string with length greater than 0'
AssertionError: Name must be a string with length greater than 0
p.setAge(-1)
Traceback (most recent call last):
  File "<pyshell#17>", line 1, in <module>
    p.setAge(-1)
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 29, in setAge
    assert type(age)==int and age>0 , 'Age must be an integer and greater than 0'
AssertionError: Age must be an integer and greater than 0
p.setEmail(1)
Traceback (most recent call last):
  File "<pyshell#18>", line 1, in <module>
    p.setEmail(1)
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 38, in setEmail
    assert type(email)==str , 'Email must be a string'
AssertionError: Email must be a string

```

```
p.getName()  
'Bob'  
p.getAge()  
1  
p.getEmail()  
' '  
p.setEmail('bob@bob.com')  
p.getEmail()  
'bob@bob.com'  
|
```

## Problem 2: Student Class

### General Information

Write a Student class has the following characteristic and behaviors:

- Name: a name is required and must be a string with length greater than 0.
- Age: is an integer and must be greater than 0
- Email: is not optional and must be a string greater than 5 characters
- yearInProgram is a string and not optional. It must be a value of Freshman, Sophomore, Junior or Senior. The value is case sensitive.

```
s=Student()
Traceback (most recent call last):
  File "<pyshell#30>", line 1, in <module>
    s=Student()
TypeError: Student.__init__() missing 4 required positional argument
s: 'name', 'age', 'email', and 'yearInProgram'
s=Student('Jill',5,'','Test')
Traceback (most recent call last):
  File "<pyshell#31>", line 1, in <module>
    s=Student('Jill',5,'','Test')
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 57, in __init__
    assert type(email)==str and len(email)>5, 'Email must be a string and have more than 5 characters'
AssertionError: Email must be a string and have more than 5 characters
s=Student('Jill',5,'jill@jill.com','Test')
Traceback (most recent call last):
  File "<pyshell#32>", line 1, in <module>
    s=Student('Jill',5,'jill@jill.com','Test')
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 59, in __init__
    assert type(yearInProgram)==str and yearInProgram in ('Freshman','Sophomore','Junior','Senior'), 'yearInProgram must be a string and value of Freshman,Sophomore,Junior or Senior'
AssertionError: yearInProgram must be a string and value of Freshman,Sophomore,Junior or Senior
s=Student('Jill',5,'jill@jill.com','Junior')
repr(s)
"Student('Jill',5,'jill@jill.com','Junior')"
```

```
s=Student('Jill',5,'jill@jill.com','Junior')
repr(s)
"Student('Jill',5,'jill@jill.com','Junior')"
s2=eval(repr(s))
s2
Student('Jill',5,'jill@jill.com','Junior')
```

```
s.getYearInProgram()
'Junior'
s.setYearInProgram('junior')
Traceback (most recent call last):
  File "<pyshell#39>", line 1, in <module>
    s.setYearInProgram('junior')
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 65, in setYearInProgram
    assert type(yearInProgram)==str and yearInProgram in ('Freshman',
'Sophmore','Junior','Senior'), 'yearInProgram must be a string and
value of Freshman,Sophmore,Junior or Senior'
AssertionError: yearInProgram must be a string and value of Freshman
,Sophmore,Junior or Senior
s.setYearInProgram('Senior')
s.getName()
'Jill'
s.getAge()
5
s.getEmail()
'jill@jill.com'
```



## Problem 3: Instructor Class

### General Information

Write an Instructor class has the following characteristic and behaviors:

- Name: a name is required and must be a string with length greater than 0.
- Age: is an integer and must be greater than 0
- Email: is not optional and must be a string greater than 5 characters
- Office: is not optional and must be a string greater than 0 characters

```
i=Instructor()
Traceback (most recent call last):
  File "<pyshell#4>", line 1, in <module>
    i=Instructor()
TypeError: Instructor.__init__() missing 4 required positional arguments: 'name'
, 'age', 'email', and 'office'
i=Instructor('Ahmed',55,'ahmed@ahmed.com',1)
Traceback (most recent call last):
  File "<pyshell#5>", line 1, in <module>
    i=Instructor('Ahmed',55,'ahmed@ahmed.com',1)
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/
CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py",
line 94, in __init__
    assert type(office)==str and len(office)>0, 'Office must be a string with le
ngth greater than 0'
AssertionError: Office must be a string with length greater than 0
i=Instructor('Ahmed',55,'ahmed@ahmed.com','CDM 200')
i
Instructor('Ahmed',55,'ahmed@ahmed.com','CDM 200')
```

```
i.setEmail('')
Traceback (most recent call last):
  File "<pyshell#10>", line 1, in <module>
    i.setEmail('')
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 108, in setEmail
    assert type(email)==str and len(email)>5, 'Email must be a string and have more than 5 characters'
AssertionError: Email must be a string and have more than 5 characters
i.setOffice(1)
Traceback (most recent call last):
  File "<pyshell#11>", line 1, in <module>
    i.setOffice(1)
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 103, in setOffice
    assert type(office)==str and len(office)>0, 'Office must be a string with length greater than 0'
AssertionError: Office must be a string with length greater than 0
```

```
str(i)
'Instructor Ahmed email address is ahmed@ahmed.com. They are located in office CDM 200.'
id(i)
1981020593456
i2=eval(repr(i))
i2
Instructor('Ahmed', 55, 'ahmed@ahmed.com', 'CDM 200')
id(i2)
1981020591440
```

## Problem 4: Professor Class

### General Information

Write a professor class that has the following characteristics and behaviors:

- Name: a name is required and must be a string with length greater than 0.
- Age: is an integer and must be greater than 0
- Email: is not optional and must be a string greater than 5 characters
- Office: is not optional and must be a string greater than 0 characters
- ResearchArea: is not optional and must be a string greater than 0 characters

```

p=Professor()
Traceback (most recent call last):
  File "<pyshell#38>", line 1, in <module>
    p=Professor()
TypeError: Professor.__init__() missing 5 required positional arguments: 'name', 'age', 'email', 'office', and 'researchArea'
p=Professor('Isabel',60,'isabel@isabel.com','CDM 201',0)
SyntaxError: invalid syntax. Perhaps you forgot a comma?
p=Professor('Isabel',60,'isabel@isabel.com','CDM 201',0)
Traceback (most recent call last):
  File "<pyshell#40>", line 1, in <module>
    p=Professor('Isabel',60,'isabel@isabel.com','CDM 201',0)
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 126, in __init__
    assert type(researchArea)==str and len(researchArea)>0, 'researchArea must be a string with length greater than 0'
AssertionError: researchArea must be a string with length greater than 0
p=Professor('Isabel',60,'isabel@isabel.com','CDM 201','Computer Science')
str(p)
'Professor Isabel email address is isabel@isabel.com. They are located in office CDM 201. Their research area is Computer Science.'

```

```
p
Professor('Isabel',60,'isabel@isabel.com
','CDM 201','Computer Science')
p.getName()
'Isabel'
p.setName('')
Traceback (most recent call last):
  File "<pyshell#48>", line 1, in <modul
e>
    p.setName('')
  File "C:/Users/azoko.BASEMENT/OneDrive
- DePaul University/Documents/Teaching/C
SC 242/Evening Class - Section 901/Week
3/Assignment 3/csc242hw3-solution.py", l
ine 20, in setName
    assert type(name)==str and len(name)
>0, 'Name must be a string with length g
reater than 0'
AssertionError: Name must be a string wi
th length greater than 0
p.getAge()
60
p.getEmail()
'isabel@isabel.com'
p.getOffice()
'CDM 201'
```

```
p.getResearchArea()
'Computer Science'
p.setResearchArea('')
Traceback (most recent call last):
  File "<pyshell#54>", line 1, in <module>
    p.setResearchArea('')
  File "C:/Users/azoko.BASEMENT/OneDrive - DePaul University/Documents/Teaching/CSC 242/Evening Class - Section 901/Week 3/Assignment 3/csc242hw3-solution.py", line 135, in setResearchArea
    assert type(researchArea)==str and len(researchArea)>0, 'researchArea must be a string with length greater than 0'
AssertionError: researchArea must be a string with length greater than 0
```

```
id(p)
2052272382528
p2=eval(repr(p))
p2
Professor('Isabel',60,'isabel@isabel.com', 'CDM 201', 'Computer Science')
```

## Submitting the assignment

You must submit the assignment using the assignment 3 dropbox on [the D2L site](#). Submit a Python file (csc242hw3.py) with your implementation in it and comments describing your collaboration status. Submissions after the deadline listed above will be automatically rejected by the system. See the syllabus for the grading policy.

## Grading

The assignment is worth 100 points. Any student who does not submit comments in the Python file describing the contributions of each team member or indicating that they worked alone will earn a 0 on the assignment.