Lab 10

Lab 10 - Code

people.py

people.py   people_demo.py

◀ ▶ | Instructor ∨ | __repr__ ∨

```python
1  class Person:
2      def __init__(self, name, year):
3          self._name = name
4          self._year = year
5
6      def __repr__(self):
7          return "%s, %s" % (self._name, self._year)
8
9
10 class Student(Person):
11     def __init__(self, name, year, major):
12         super().__init__(name, year)
13         self._major = major
14
15     def __repr__(self):
16         return "%s, born %d is a %s major" \
17                 % (self._name, self._year, self._major)
18
19
20 class Instructor(Person):
21     def __init__(self, name, year, salary):
22         super().__init__(name, year)
23         self._salary = salary
24
25     def __repr__(self):
26         return "%s, born in %d has a $%.2f salary" \
27                 % (self._name, self._year, self._salary)
28
```

people_demo.py

people.py   people_demo.py

◀ ▶ | (bottom) ∨

```python
1  from people import *
2
3
4  def main():
5      studentdemo = Student("John Smith", 1987, "CS")
6      instructordemo = Instructor("Dave White", 1980, 80000)
7
8      print(studentdemo)
9      print(instructordemo)
10
11
12 main()
13
```

Lab 10 - Output

Python Shell: Wing

```
Python Shell

 Commands execute without debug.  Use arrow keys for history.

    Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)]
    Type "help", "copyright", "credits" or "license" for more information.
>>> [evaluate people_demo.py]
    John Smith, born 1987 is a CS major
    Dave White, born in 1980 has a $80000.00 salary
>>>
```

Lab 10 – Written Code
people.py

```python
class Person:
    def __init__(self, name, year):
        self._name = name
        self._year = year

    def __repr__(self):
        return "%s, %s" % (self._name, self._year)


class Student(Person):
    def __init__(self, name, year, major):
        super().__init__(name, year)
        self._major = major

    def __repr__(self):
        return "%s, born %d is a %s major" % (self._name, self._year, self._ma-
jor)


class Instructor(Person):
    def __init__(self, name, year, salary):
        super().__init__(name, year)
        self._salary = salary

    def __repr__(self):
        return "%s, born in %d has a $%.2f salary" % (self._name, self._year,
self._salary)
```

people_demo.py

```python
from people import *


def main():
    studentdemo = Student("John Smith", 1987, "CS")
    instructordemo = Instructor("Dave White", 1980, 80000)

    print(studentdemo)
    print(instructordemo)


main()
```