

Quiz: OOP II (Practice Problems)

1 Inheritance

Note 1. Whenever an attribute is not found in a subclass, Python will check the superclass for the attribute before raising the `AttributeError` exception.

Problem 1. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     pass
5 class Bar(Foo):
6     pass
7 a = Foo()
8 Foo.message = 'hello world'
9 b = Bar()
10 Bar.message = 'hola mundo'
11 try:
12     print('b.message=', b.message)
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

Problem 2. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     pass
5 class Bar(Foo):
6     pass
7 a = Foo()
8 Foo.message = 'hello world'
9 b = Bar()
10 a.message = 'hola mundo'
11 try:
12     print('b.message=', b.message)
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

Problem 3. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     pass
5 class Bar(Foo):
6     pass
7 a = Foo()
8 Foo.message = 'hello world'
9 b = Bar()
10 a.message = 'hola mundo'
11 try:
12     print('b.message=', b.message)
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

Problem 4. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     pass
5 class Bar(Foo):
6     pass
7 a = Foo()
8 a.message = 'hello world'
9 b = Bar()
10 try:
11     print('b.message=', b.message)
12 except AttributeError:
13     print('AttributeError')
14 EOF
15 $ python3 foo.py
```

Problem 5. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     pass
5 class Bar(Foo):
6     pass
7 a = Foo()
8 Foo.message = 'hello world'
9 b = Bar()
10 Bar.message = 'hola mundo'
11 try:
12     print('a.message=', a.message)
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

1.1 With Constructors

Note 2. The constructor of a superclass will only be called if it is explicitly called in the constructor of the subclass. You should use the `super` function to get the superclass.

Problem 6. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self):
5         self.message = 'hello world'
6 class Bar(Foo):
7     def __init__(self):
8         self.message = 'hola mundo'
9 a = Foo()
10 b = Bar()
11 try:
12     print('b.message=', b.message)
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

Problem 7. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self):
5         self.message = 'hello world'
6 class Bar(Foo):
7     def __init__(self):
8         self.message = 'hola mundo'
9         super().__init__()
10 a = Foo()
11 b = Bar()
12 try:
13     print('b.message=', b.message)
14 except AttributeError:
15     print('AttributeError')
16 EOF
17 $ python3 foo.py
```

Problem 8. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self):
5         self.message = 'hello world'
6 class Bar(Foo):
7     def __init__(self):
8         super().__init__()
9         self.message = 'hola mundo'
10 a = Foo()
11 b = Bar()
12 try:
13     print('b.message=', b.message)
14 except AttributeError:
15     print('AttributeError')
16 EOF
17 $ python3 foo.py
```

Problem 9. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6 class Bar(Foo):
7     def __init__(self, message=None):
8         self.message = message
9         super().__init__(message)
10 a = Foo('hello world')
11 b = Bar('hola mundo')
12 try:
13     print('b.message=', b.message)
14 except AttributeError:
15     print('AttributeError')
16 EOF
17 $ python3 foo.py
```

Problem 10. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         Foo.message = message
6 class Bar(Foo):
7     def __init__(self, message=None):
8         super().__init__(message)
9 a = Foo('hello world')
10 b = Bar('hola mundo')
11 try:
12     print('b.message=', b.message)
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

Problem 11. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         Foo.message = message
6 class Bar(Foo):
7     def __init__(self, message=None):
8         super().__init__(message)
9 b = Bar('hola mundo')
10 a = Foo('hello world')
11 try:
12     print('b.message=', b.message)
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

Problem 12. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     pass
5 class Bar(Foo):
6     def __init__(self, message=None):
7         super().__init__()
8         Foo.message = message
9 a = Foo()
10 b = Bar('hola mundo')
11 c = Bar()
12 try:
13     print('b.message=', b.message)
14 except AttributeError:
15     print('AttributeError')
16 EOF
17 $ python3 foo.py
```

1.2 With built-in Classes

Note 3. Recall that when you subclass an existing class, you get all the superclass's functionality "for free". It is very common in python to subclass the built-in python classes in order to extend their functionality. Again, these problems focus on the "what/how" instead of the "why".

Problem 13. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo(list):
4     def __init__(self, xs=[]):
5         super().__init__(xs + [0])
6 xs = Foo([1])
7 xs = Foo([1, 2])
8 xs = Foo([1, 2, 3])
9 try:
10     print('xs=', xs)
11 except AttributeError:
12     print('AttributeError')
13 EOF
14 $ python3 foo.py
```

Problem 14. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo(list):
4     def __init__(self, xs=[]):
5         xs.append(len(xs))
6         super().__init__(xs)
7 xs = Foo([1, 2, 3])
8 xs = Foo(xs)
9 xs = Foo(xs)
10 try:
11     print('xs=', xs)
12 except AttributeError:
13     print('AttributeError')
14 EOF
15 $ python3 foo.py
```

Problem 15. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo(list):
4     def __init__(self, xs=[]):
5         super().__init__(xs)
6         xs.append(len(xs))
7 xs = Foo()
8 xs = Foo()
9 xs = Foo()
10 try:
11     print('xs=', xs)
12 except AttributeError:
13     print('AttributeError')
14 EOF
15 $ python3 foo.py
```

Problem 16. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo(list):
4     def __init__(self, xs=[]):
5         xs.append(len(xs))
6         super().__init__(xs)
7 xs = Foo()
8 xs = Foo([1, 2, 3])
9 xs = Foo()
10 try:
11     print('xs=', xs)
12 except AttributeError:
13     print('AttributeError')
14 EOF
15 $ python3 foo.py
```

2 Static Methods

2.1 Without Inheritance

Problem 17. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6     def foo(self):
7         return self.message
8 a = Foo('hello world')
9 try:
10     print('a.foo()=', a.foo())
11 except AttributeError:
12     print('AttributeError')
13 EOF
14 $ python3 foo.py
```

Problem 18. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     message = 'salve munde'
5     def __init__(self, message=None):
6         self.message = message
7     @staticmethod
8     def foo():
9         return Foo.message
10 a = Foo('hello world')
11 try:
12     print('a.foo()=', a.foo())
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

Problem 19. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     message = 'salve munde'
5     def __init__(self, message=None):
6         self.message = message
7     @staticmethod
8     def foo():
9         return Foo.message
10 a = Foo('hello world')
11 try:
12     print('Foo.foo()=', Foo.foo())
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

Problem 20. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     message = 'salve munde'
5     def __init__(self, message=None):
6         Foo.message = message
7     @staticmethod
8     def foo():
9         return Foo.message
10 a = Foo('hello world')
11 try:
12     print('Foo.foo()=', Foo.foo())
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```

2.2 With Inheritance

Problem 21. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6     def foo(self):
7         return self.message
8 class Bar(Foo):
9     def __init__(self, message=None):
10         super().__init__(message)
11     def bar(self):
12         return self.message
13 a = Foo('hello world')
14 b = Bar('hola mundo')
15 try:
16     print('b.foo()=', b.foo())
17 except AttributeError:
18     print('AttributeError')
19 EOF
20 $ python3 foo.py
```

Problem 22. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6     def foo(self):
7         return self.message
8 class Bar(Foo):
9     def __init__(self, message=None):
10         super().__init__(message)
11     def bar(self):
12         return self.message
13 a = Foo('hello world')
14 b = Bar('hola mundo')
15 try:
16     print('a.bar()=', a.bar())
17 except AttributeError:
18     print('AttributeError')
19 EOF
20 $ python3 foo.py
```

Problem 23. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6     def foo(self):
7         return self.message
8 class Bar(Foo):
9     message = 'salve munde'
10    def __init__(self, message=None):
11        super().__init__(message)
12    @staticmethod
13    def bar():
14        return Bar.message
15 a = Foo('hello world')
16 b = Bar('hola mundo')
17 try:
18     print('b.bar()=', b.bar())
19 except AttributeError:
20     print('AttributeError')
21 EOF
22 $ python3 foo.py
```

Problem 24. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6     def foo(self):
7         return self.message
8 class Bar(Foo):
9     message = 'salve munde'
10    def __init__(self, message=None):
11        super().__init__(message)
12    @staticmethod
13    def bar():
14        return Bar.message
15 a = Foo('hello world')
16 b = Bar('hola mundo')
17 try:
18     print('Bar.bar()=', Bar.bar())
19 except AttributeError:
20     print('AttributeError')
21 EOF
22 $ python3 foo.py
```

Problem 25. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     message = 'salve munde'
5     def __init__(self, message=None):
6         Bar.message = message
7     @staticmethod
8     def foo():
9         return Bar.message
10 class Bar(Foo):
11     def __init__(self, message=None):
12         super().__init__(message)
13     @staticmethod
14     def bar():
15         return Foo.message
16 a = Foo('hello world')
17 b = Bar('hola mundo')
18 try:
19     print('Bar.foo()=', Bar.foo())
20 except AttributeError:
21     print('AttributeError')
22 EOF
23 $ python3 foo.py
```

Problem 26. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     message = 'salve munde'
5     def __init__(self, message=None):
6         Bar.message = message
7     @staticmethod
8     def foo():
9         return Bar.message
10 class Bar(Foo):
11     def __init__(self, message=None):
12         super().__init__(message)
13     @staticmethod
14     def bar():
15         return Foo.message
16 a = Foo('hello world')
17 b = Bar('hola mundo')
18 try:
19     print('Bar.bar()=', Bar.bar())
20 except AttributeError:
21     print('AttributeError')
22 EOF
23 $ python3 foo.py
```

3 Recursive Classes

Problem 27. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6 foo = Foo('hello world')
7 foo.child = Foo('hola mundo')
8 foo.child.child = Foo('salve munde')
9 try:
10     print(' foo.child.message=', foo.child.message)
11 except AttributeError:
12     print('AttributeError')
13 EOF
14 $ python3 foo.py
```

Problem 28. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6 foo = Foo('hello world')
7 foo.child = Foo('hola mundo')
8 foo.child.child = Foo('salve munde')
9 try:
10     print(' foo.child.child.child.message=', foo.child.child.child.message)
11 except AttributeError:
12     print('AttributeError')
13 EOF
14 $ python3 foo.py
```

Problem 29. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6 foo = Foo('hello world')
7 foo.child = foo
8 try:
9     print(' foo.child.child.child.message=', foo.child.child.child.message)
10 except AttributeError:
11     print('AttributeError')
12 EOF
13 $ python3 foo.py
```

Problem 30. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6 foo = Foo('hello world')
7 foo.child = Foo('hola mundo')
8 foo.child.child = foo
9 try:
10     print(' foo.child.child.child.message=', foo.child.child.child.message)
11 except AttributeError:
12     print('AttributeError')
13 EOF
14 $ python3 foo.py
```

Problem 31. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo:
4     def __init__(self, message=None):
5         self.message = message
6 foo = Foo('hello world')
7 foo.child = Foo('hola mundo')
8 foo.child.child = foo.child
9 try:
10     print(' foo.child.child.child.message=', foo.child.child.child.message)
11 except AttributeError:
12     print('AttributeError')
13 EOF
14 $ python3 foo.py
```

3.1 With built-in classes

Problem 32. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 xs = [1, 2, 3]
4 xs.append(xs)
5 xs[0] = xs
6 try:
7     print('xs[-1][0][2]=' ,xs[-1][0][2])
8 except TypeError:
9     print('TypeError')
10 EOF
11 $ python3 foo.py
```

Problem 33. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 xs = [1, 2, 3]
4 xs.append(xs)
5 xs[0] = xs
6 try:
7     print('xs[1][0][2]=' ,xs[1][0][2])
8 except TypeError:
9     print('TypeError')
10 EOF
11 $ python3 foo.py
```

Problem 34. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
1 $ cd; rm -rf quiz; mkdir quiz; cd quiz
2 $ cat > foo.py <<EOF
3 class Foo(list):
4     def __init__(self, xs=[]):
5         xs.append(len(xs))
6         super().__init__(xs)
7 xs = Foo()
8 xs.append(Foo())
9 xs[0] = 1
10 xs.append(Foo())
11 try:
12     print('xs[0]=' , xs[0])
13 except AttributeError:
14     print('AttributeError')
15 EOF
16 $ python3 foo.py
```