

[15-112] Lecture 19

Lecture 19: Poll 1 (SOLO)

What does the following code print?

```
class A:
    d = dict()
    def __init__(self, x, y):
        self.x = x
        self.y = y
        A.d[x] = A.d.get(x,y) + 1

# What does this print?
a = A(1,1)
print(A.d)
```

- A. {}
- B. {1:2}
- C. {2:1}
- D. It will crash with syntax error
- E. It will crash with a runtime error
- F. I don't know

Lecture 19: Poll 1 (GROUP)

What does the following code print?

```
class A:  
    d = dict()  
    def __init__(self, x, y):  
        self.x = x  
        self.y = y  
        A.d[x] = A.d.get(x,y) + 1  
  
# What does this print?  
a = A(1,1)  
print(A.d)
```

- A. {}
- B. {1:2}
- C. {2:1}
- D. It will crash with syntax error
- E. It will crash with a runtime error
- F. I don't know

Lecture 19: Poll 1 (SOLUTION)

What does the following code print?

```
class A:
    d = dict()
    def __init__(self, x, y):
        self.x = x
        self.y = y
        A.d[x] = A.d.get(x,y) + 1

# What does this print?
a = A(1,1)
print(A.d)
```

- A. {}
- B. {1:2}**
- C. {2:1}
- D. It will crash with syntax error
- E. It will crash with a runtime error
- F. I don't know

```
class A:
    d = dict()
    def __init__(self, x, y):
        self.x = x
        self.y = y
        A.d[x] = A.d.get(x,y) + 1
```

[PUT METHOD HERE]

```
A. @staticmethod
    def updateDv1(x, y):
        A.d[x] = y
        print(A.d)
a = A(1,1)
A.updateDv1(1,3)
```

```
B. @staticmethod
    def updateDv2(self,x, y):
        A.d[x] = y
        print(A.d)
a = A(1,1)
A.updateDv2(1,3)
```

Lecture 19: Poll 2 (SOLO)

Which of the following methods, prints {1:3}?

```
C. @staticmethod
    def updateDv3(x, y):
        self.d[x] = y
        print(self.d)
a = A(1,1)
a.updateDv3(1,3)
```

```
D. @staticmethod
    def updateDv4(self, x, y):
        self.d[x] = y
        print(self.d)
a = A(1,1)
a.updateDv4(1,3)
```

E. None of the above
F. I don't know

```
class A:
    d = dict()
    def __init__(self, x, y):
        self.x = x
        self.y = y
        A.d[x] = A.d.get(x,y) + 1
```

[PUT METHOD HERE]

```
A. @staticmethod
    def updateDv1(x, y):
        A.d[x] = y
        print(A.d)
a = (1,1)
A.updateDv1(1,3)
```

```
B. @staticmethod
    def updateDv2(self,x, y):
        A.d[x] = y
        print(A.d)
a = (1,1)
A.updateDv2(1,3)
```

Lecture 19: Poll 2 (GROUP)

Which of the following methods, prints {1:3}?

```
C. @staticmethod
    def updateDv3(x, y):
        self.d[x] = y
        print(self.d)
a = (1,1)
a.updateDv3(1,3)
```

```
D. @staticmethod
    def updateDv4(self, x, y):
        self.d[x] = y
        print(self.d)
a = (1,1)
a.updateDv4(1,3)
```

E. None of the above
F. I don't know

```
class A:
    d = dict()
    def __init__(self, x, y):
        self.x = x
        self.y = y
        A.d[x] = A.d.get(x,y) + 1
```

[PUT METHOD HERE]

```
A. @staticmethod
    def updateDv1(x, y):
        A.d[x] = y
        print(A.d)
a = (1,1)
A.updateDv1(1,3)
```

```
B. @staticmethod
    def updateDv2(self,x, y):
        A.d[x] = y
        print(A.d)
a = (1,1)
A.updateDv2(1,3)
```

Lecture 19: Poll 2 (SOLUTION)

Which of the following methods, prints {1:3}?

```
C. @staticmethod
    def updateDv3(x, y):
        self.d[x] = y
        print(self.d)
a = (1,1)
a.updateDv3(1,3)
```

```
D. @staticmethod
    def updateDv4(self, x, y):
        self.d[x] = y
        print(self.d)
a = (1,1)
a.updateDv4(1,3)
```

E. None of the above
F. I don't know

Lecture 19: Poll 3 (SOLO)

What is the type of self in sayHello?

```
class Pet:
    def __init__(self, name):
        self.name = name

    def sayHello(self):
        # WHAT IS SELF HERE?
        print(f"Hi my name is {self.name}" +
              f"and I am a {(type(self))} object")

class Dino(Pet):
    def __init__(self, name, species):
        super().__init__(name)
        self.species = species
        self.title = f"{name} the {species}"

d = Dino("Trixie", "Triceratops")
d.sayHello()
```

- A. Object
- B. Class
- C. Pet
- D. Dino
- E. String
- F. Not enough information
- G. I don't know

Lecture 19: Poll 3 (GROUP)

What is the type of self in sayHello?

```
class Pet:
    def __init__(self, name):
        self.name = name

    def sayHello(self):
        # WHAT IS SELF HERE?
        print(f"Hi my name is {self.name}" +
              f"and I am a {(type(self))} object")

class Dino(Pet):
    def __init__(self, name, species):
        super().__init__(name)
        self.species = species
        self.title = f"{name} the {species}"

d = Dino("Trixie", "Triceratops")
d.sayHello()
```

- A. Object
- B. Class
- C. Pet
- D. Dino
- E. String
- F. Not enough information
- G. I don't know

Lecture 19: Poll 3 (SOLUTION)

What is the type of self in sayHello?

```
class Pet:
    def __init__(self, name):
        self.name = name

    def sayHello(self):
        # WHAT IS SELF HERE?
        print(f"Hi my name is {self.name}" +
              f"and I am a {(type(self))} object")

class Dino(Pet):
    def __init__(self, name, species):
        super().__init__(name)
        self.species = species
        self.title = f"{name} the {species}"

d = Dino("Trixie", "Triceratops")
d.sayHello()
```

- A. Object
- B. Class
- C. Pet
- D. **Dino**
- E. String
- F. Not enough information
- G. I don't know

Lecture 17 Poll 3 (SOLO)

What is the efficiency of the following is True?

- A. $O(1)$
- B. $O(N)$
- C. $O(N^2)$
- D. $O(\log N)$
- E. $O(N \log N)$
- F. I don't know