

Q1: Name Resolution and Inheritance

```
class A:
```

```
    def f(self):  
        return self.g()
```

```
    def g(self):  
        return 10
```

```
class B(A):
```

```
    def g(self):  
        return 14
```

```
    def h(self):  
        return 18
```

- Execute the following:
 >>> **a** = A()
 >>> **b** = B()
- What is value of **a.f()**?

A: 10

B: 14

C: 5

D: ERROR

E: I don't know

A1: Name Resolution and Inheritance

```
class A:
```

```
    def f(self):  
        return self.g()
```

```
    def g(self):  
        return 10
```

```
class B(A):
```

```
    def g(self):  
        return 14
```

```
    def h(self):  
        return 18
```

- Execute the following:
 >>> **a** = A()
 >>> **b** = B()

• What is value of **a.f()**?

A: 10 **CORRECT**

B: 14

C: 5

D: ERROR

E: I don't know

Q2: Name Resolution and Inheritance

```
class A:
```

```
    def f(self):  
        return self.g()
```

```
    def g(self):  
        return 10
```

```
class B(A):
```

```
    def g(self):  
        return 14
```

```
    def h(self):  
        return 18
```

- Execute the following:
 >>> **a** = A()
 >>> **b** = B()
- What is value of **b.f()**?

A: 10

B: 14

C: 5

D: ERROR

E: I don't know

A2: Name Resolution and Inheritance

```
class A:
```

```
    def f(self):  
        return self.g()
```

```
    def g(self):  
        return 10
```

```
class B(A):
```

```
    def g(self):  
        return 14
```

```
    def h(self):  
        return 18
```

- Execute the following:
 >>> **a** = A()
 >>> **b** = B()
- What is value of **b.f()**?

A: 10

B: 14 CORRECT

C: 5

D: ERROR

E: I don't know

Q3: Name Resolution and Inheritance

```
class A:
    x = 3 # Class Variable
    y = 5 # Class Variable

    def f(self):
        return self.g()

    def g(self):
        return 10

class B(A):
    y = 4 # Class Variable
    z = 42 # Class Variable

    def g(self):
        return 14

    def h(self):
        return 18
```

- Execute the following:
 >>> **a** = A()
 >>> **b** = B()
- What is value of **b.x**?

A: 4
B: 3
C: 42
D: ERROR
E: I don't know

A3: Name Resolution and Inheritance

```
class A:
    x = 3 # Class Variable
    y = 5 # Class Variable

    def f(self):
        return self.g()

    def g(self):
        return 10

class B(A):
    y = 4 # Class Variable
    z = 42 # Class Variable

    def g(self):
        return 14

    def h(self):
        return 18
```

- Execute the following:
 >>> **a** = A()
 >>> **b** = B()
- What is value of **b.x**?

A: 4
B: 3 CORRECT
C: 42
D: ERROR
E: I don't know

Q4: Name Resolution and Inheritance

```
class A:
    x = 3 # Class Variable
    y = 5 # Class Variable

    def f(self):
        return self.g()

    def g(self):
        return 10

class B(A):
    y = 4 # Class Variable
    z = 42 # Class Variable

    def g(self):
        return 14

    def h(self):
        return 18
```

- Execute the following:
 >>> **a** = A()
 >>> **b** = B()
- What is value of **a.z**?

A: 4
B: 3
C: 42
D: ERROR
E: I don't know

A4: Name Resolution and Inheritance

```
class A:
    x = 3 # Class Variable
    y = 5 # Class Variable

    def f(self):
        return self.g()

    def g(self):
        return 10

class B(A):
    y = 4 # Class Variable
    z = 42 # Class Variable

    def g(self):
        return 14

    def h(self):
        return 18
```

- Execute the following:
 >>> **a** = A()
 >>> **b** = B()
- What is value of **a.z**?

A: 4
B: 3
C: 42
D: ERROR **CORRECT**
E: I don't know

eq vs. is

`==` compares equality

`is` compares identity

```
c1 = Circle(1, 1, 25)
```

```
c2 = Circle(1, 1, 25)
```

```
c3 = c2
```

```
c1 == c2 → ?
```

```
c1 is c2 → ?
```

```
c2 == c3 → ?
```

```
c2 is c3 → ?
```

c1 id4

c2 id5

c3 id5

id4

Circle

x 1

y 1

radius 25

id5

Circle

x 1

y 1

radius 25

eq vs. is

`==` compares equality

`is` compares identity

```
c1 = Circle(1, 1, 25)
```

```
c2 = Circle(1, 1, 25)
```

```
c3 = c2
```

```
c1 == c2 → ? True
```

```
c1 is c2 → ? False
```

```
c2 == c3 → ? True
```

```
c2 is c3 → ? True
```

c1 id4

c2 id5

c3 id5

id4

Circle

x 1

y 1

radius 25

id5

Circle

x 1

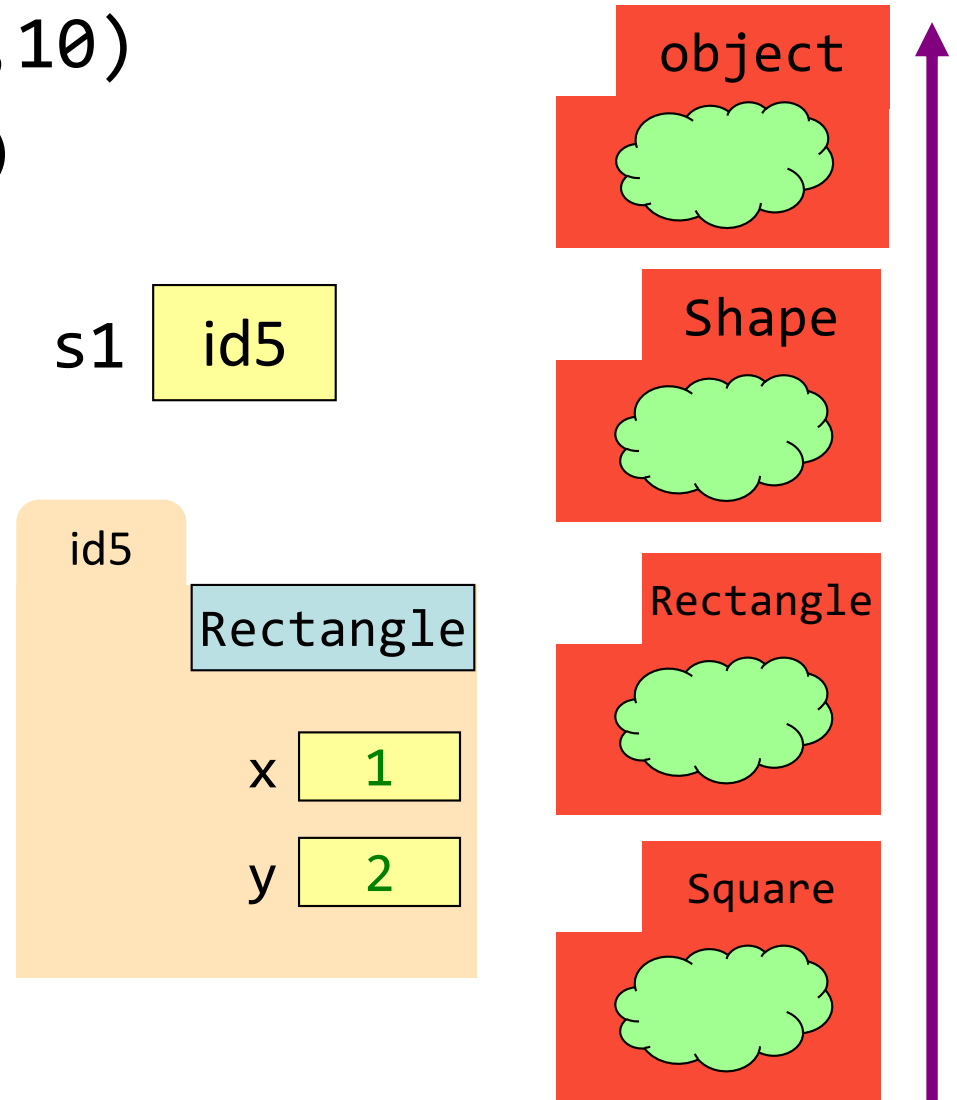
y 1

radius 25

Q5: `isinstance` and Subclasses

```
>>> s1 = Rectangle(0,0,10,10)
>>> isinstance(s1, Square)
???
```

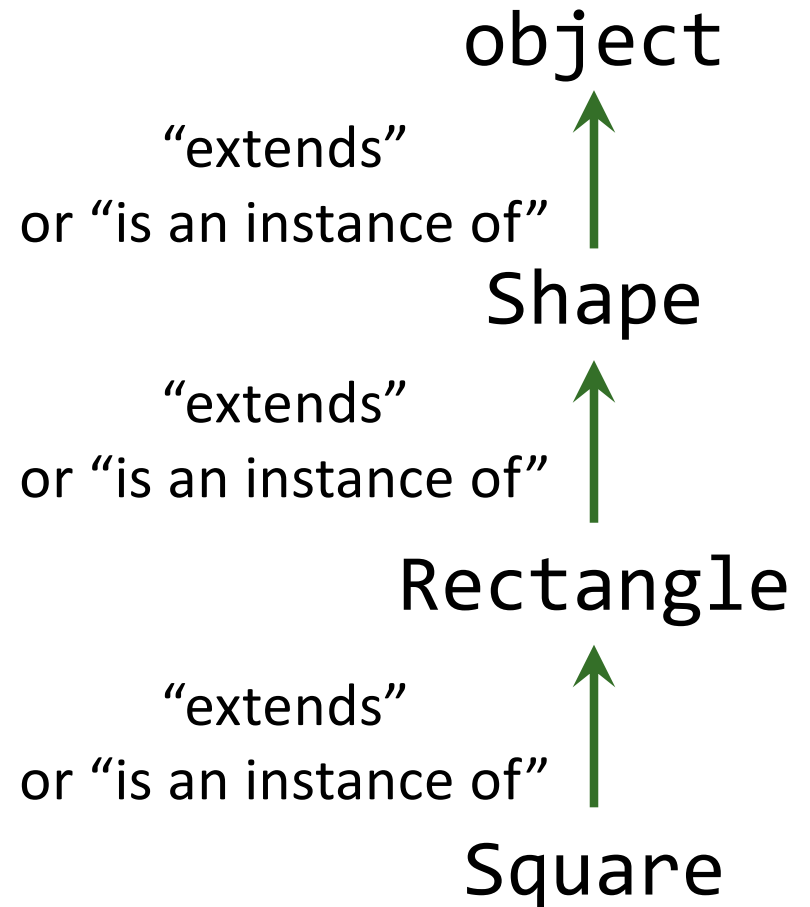
- A: True
- B: False
- C: Error
- D: I don't know



A5: `isinstance` and Subclasses

```
>>> s1 = Rectangle(0,0,10,10)
>>> isinstance(s1, Square)
???
```

- A: True
- B: False
- C: Error
- D: I don't know



A5: `isinstance` and Subclasses

```
>>> s1 = Rectangle(0,0,10,10)
>>> isinstance(s1, Square)
???
```

A: True
B: False **CORRECT**
C: Error
D: I don't know

