

A1: Name Resolution and Inheritance

```
class A0:
    def f(self):
        | return self.g0

    def g(self):
        | return 10

class B(A):
    def g(self):
        | return 14

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

- Execute the following:


```
>>> a = A0
>>> b = B0
```
- What is value of `a.f()`?

A: 10 **CORRECT**
 B: 14
 C: 5
 D: *ERROR*
 E: *I don't know*

13

A2: Name Resolution and Inheritance

```
class A0:
    def f(self):
        | return self.g0

    def g(self):
        | return 10

class B(A):
    def g(self):
        | return 14

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

- Execute the following:


```
>>> a = A0
>>> b = B0
```
- What is value of `b.f()`?

A: 10
 B: 14 **CORRECT**
 C: 5
 D: *ERROR*
 E: *I don't know*

15

A3: Name Resolution and Inheritance

```
class A0:
    x = 3 # Class Variable
    y = 5 # Class Variable

    def f(self):
        | return self.g0

    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 = A0
>>> b = B0
```
- What is value of `b.x`?

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

22

A4: Name Resolution and Inheritance

```
class A0:
    x = 3 # Class Variable
    y = 5 # Class Variable

    def f(self):
        | return self.g0

    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 = A0
>>> b = B0
```
- What is value of `a.z`?

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

24

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
```

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*

```

graph BT
    object -- "extends" --> Shape
    Shape -- "extends" --> Rectangle
    Rectangle -- "extends" --> Square
    
```

31