

# ***KillTest***

高品質 より良いサービス



## 練習問題

<https://www.killtest.jp>

1年間無料アップデート

**Exam : PCPP-32-101**

**Title : PCPP1-Certified  
Professional in Python  
Programming 1**

**Version : DEMO**

1. Select the true statement about composition.

- A. Composition extends a class's capabilities by adding new components and modifying the existing ones
- B. Composition allows a class to be projected as a container of different classes
- C. Composition is a concept that promotes code reusability, while Inheritance promotes encapsulation
- D. Composition is based on the has a relation, so it cannot be used together with inheritance

**Answer: B**

2. Analyze the following snippet and select the statement that best describes it.

```
class OwnMath:
    pass

def calculate_value(numerator, denominator):
    try:
        value = numerator / denominator
    except ZeroDivisionError as e:
        raise OwnMath from e
    return value
```

```
calculate_value(4, 0)
```

- A. The code is an example of implicitly chained exceptions.
- B. The code is erroneous as the OwnMath class does not inherit from any Exception type class
- C. The code is fine and the script execution is not interrupted by any exception.
- D. The code is an example of explicitly chained exceptions

**Answer: B**

3. Analyze the following snippet and select the statement that best describes it.

```
class Sword:
    var1 = 'weapon'

    def __init__(self):
        self.name = 'Excalibur'
```

- A. self.name is the name of a class variable
- B. var1 is the name of a global variable
- C. Excalibur is the value passed to an instance variable
- D. weapon is the value passed to an instance variable

**Answer: C**

4.The following snippet represents one of the OOP pillars.

```
class A:
    def run(self):
        print("A is running")

class B:
    def fly(self):
        print("B is flying")

class C:
    def run(self):
        print("C is running")

for element in A(), B(), C():
    element.run()
```

Which one is that?

- A. Serialization
- B. Inheritance
- C. Encapsulation
- D. Polymorphism

**Answer: D**

5.Analyze the following function and choose the statement that best describes it.

```
def my_decorator(coating):  
    def level1_wrapper(my_function):  
        def level2_wrapper(*args)  
            our_function(*args)  
        return level2_wrapper  
    return level1_wrapper
```

- A. It is an example of a decorator that accepts its own arguments.
- B. It is an example of decorator stacking.
- C. It is an example of a decorator that can trigger an infinite recursion
- D. The function is erroneous.

**Answer:** D