

✓ Congratulations! You passed!

[Go to next item](#)Grade received **100%** To pass 80% or higher

1. Which of these is an example of instantiating a Python object?

1 / 1 point

- ☐ class Fighter():
- ☒ fighter = Fighter()
- ☐ def punch():
- ☐ print(fighter)

✓ **Correct**

Correct. This creates a Fighter object by calling the Fighter class.

2. What allows a Python class to inherit attributes and behaviors?

1 / 1 point

- ☐ The self parameter
- ☒ The parentheses after the class name
- ☐ The init method
- ☐ The super function

✓ **Correct**

Yes, by putting the parent class name in parentheses a child class can inherit from it.

3. Which decorator allows creating read-only attributes?

1 / 1 point

- ☐ @method
- ☐ @inherit
- ☒ @property
- ☐ @attribute

✓ **Correct**

Correct. @property creates a read-only attribute.

4. What allows a child class to call a parent method directly?

1 / 1 point

- ☒ The super function
- ☐ The self parameter
- ☐ The init method
- ☐ The inherit function

✓ **Correct**

Correct. Super explicitly calls the parent class method.

5. Why might name mangling with double underscores be used?

1 / 1 point

- ☒ To avoid name clashes with subclasses
- ☐ To make methods private
- ☐ To call superclass constructors
- ☐ To inherit built-in Python methods

✓ **Correct**

Correct. Name mangling avoids conflicts between class variable names.

