

## Zadanie J

### Singleton i Final

Twoim zadaniem jest zaimplementowanie metaklas **Singleton** i **Final**. Klasa będąca obiektem typu **Singleton** może wyprodukować tylko jeden obiekt swojego typu, zaś klasa będąca obiektem typu **Final** ma symulować klasę finalną z Javy. Dokładne zachowanie się klas typu **Singleton** i **Final** proszę wywnioskować z testów przykładowych.

## Wysyłanie rozwiązania

Masz wysłać jeden plik o nazwie **meta.py**, w którym będzie znajdowała się definicja metaklas **Singleton** i **Final**. Testami są pliki **\*.py**, które importują metaklasy **Singleton** oraz **Final** i sprawdzają ich poprawność.

Dostępna pamięć: 512MB

## Przykład

Dla testu:

```
from meta import Singleton, Final
```

```
class A(metaclass = Singleton):
```

```
    def __init__(self, *args, **kwargs):
        self._args = args
        self._kwargs = kwargs
```

```
    def show(self):
```

```
        print("{} object, args:{}, kwargs: {}".
              format(self.__class__.__name__, self._args, self._kwargs ))
```

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

```
    pass
```

```
class RegularClass:
```

```
    pass
```

```
x = A("A1", val = 2)
```

```
y = A("A2", val = 3)
```

```
x.show()
```

```
y.show()
```

```
print(x is y)
```

```
x = B("B1", val = 2)
```

```
y = B("B2", val = 3)
```

```
x.show()
```

```
y.show()
```

```
print(x is y)
```

```
x = A()
```

```
y = B()
```

```
x.show()
```

```
y.show()
```

```
print(x is y)
```

```
x = RegularClass()
```

```
y = RegularClass()
```

```
print(x is y)
```

Poprawną odpowiedzią jest:

```
A object, args:('A1',), kwargs: {'val': 2}
```

```
A object, args:('A1',), kwargs: {'val': 2}
```

```
True
```

```
B object, args:('B1',), kwargs: {'val': 2}
```

```
B object, args:('B1',), kwargs: {'val': 2}
```

```
True
```

```
A object, args:('A1',), kwargs: {'val': 2}
```

```
B object, args:('B1',), kwargs: {'val': 2}
```

```
False
```

```
False
```

Dla testu:

```
from meta import Final

class A(object):
    pass

class B(A, metaclass=Final):
    pass

print(isinstance(B, Final))

try:
    class C(B):
        pass
except TypeError:
    print("Cannot inherit from final class")
```

Poprawną odpowiedzią jest:

```
True
Cannot inherit from final class
```