

Finals Task 2. Inheritance

performer.py

```
class Performer:
    def __init__(self, name: str, age: int):
        self.name = name
        self.age = age

    def get_name(self) -> str:
        return self.name

    def get_age(self) -> int:
        return self.age
```

singer.py

```
from performer import Performer

class Singer(Performer):
    def __init__(self, name: str, age: int, vocal_range: str):
        super().__init__(name, age)
        self.vocal_range = vocal_range

    def get_vocal_range(self) -> str:
        return self.vocal_range

    def sing(self) -> None:
        print(f"{self.name} is singing with a {self.vocal_range} range.")
```

dancer.py

```
from performer import Performer

class Dancer(Performer):
    def __init__(self, name: str, age: int, dance_style: str):
        super().__init__(name, age)
        self.dance_style = dance_style

    def get_dance_style(self) -> str:
        return self.dance_style

    def dance(self) -> None:
        print(f"{self.name} is performing {self.dance_style} dance.")
```

test_class.py

```
from performer import Performer
from singer import Singer
from dancer import Dancer

# test case 1
p = Performer("John", 25)
print([p.get_name(), p.get_age()]) # ['John', 25]

# test case 2
d = Dancer("Emily", 28, "Ballet")
print([d.get_name(), d.get_age(), d.get_dance_style()]) # ['Emily', 28, 'Ballet']

# test case 3
d.dance() # Emily is performing Ballet dance.

# test case 4
print(issubclass(Dancer, Performer)) # True

# test case 5
s = Singer("Linda", 35, "Soprano")
print([s.get_name(), s.get_age(), s.get_vocal_range()]) # ['Linda', 35, 'Soprano']

# test case 6
s.sing() # Linda is singing with a Soprano range.
```

Sample output for the Test Class

```
['John', 25]
['Emily', 28, 'Ballet']
Emily is performing Ballet dance.
True
['Linda', 35, 'Soprano']
Linda is singing with a Soprano range.
```