Name: **Torres, Justine Kurt Q.**
7OOP
Course & Section: **BSCS – C204** **Finals**
Lab Task 3

Simple Polymorphism

Problem: Chirp and Tweet
Create a simple program to demonstrate basic polymorphism with bird sounds.

Class – Bird:
- Methods:
  - def make_sound(self) -> None: An abstract method that represents making a sound. It doesn't have a specific implementation in the base class Bird.

Class – Sparrow (extends Bird):
- Methods:
  - def make_sound(self) -> None: Overrides the make_sound method from the base class Bird. It prints the sound "Chirp Chirp" when called.

Class – Parrot (extends Bird):
- Methods:
  - def make_sound(self) -> None: Overrides the make_sound method from the base class Bird. It prints the sound "Tweet Tweet" when called.

Class – BirdCage:
- Methods:
  - def make_sound(self, birds: List) -> None: Accepts a list of Bird object as input. Iterates through the list of birds and calls the make_sound method on each bird to make it's sound.

Note:
- The test cases are not outputs of your main file but of a hidden test file. Create and implement the classes instructed to test your code.
- Each class should be defined in its own file, with file name following camelCase conventions (e.g., bankAccount.py)

Source Codes:
Class Bird:

main.py | bird.py | sparrow.py | parrot.py | birdCage.py

```
1 class Bird:
2     def make_sound(self) -> None:
3         pass
```

Class Sparrow:

main.py | bird.py | sparrow.py | parrot.py | birdCage.py

```python
from bird import Bird

class Sparrow(Bird):
    def make_sound(self) -> None:
        print("Chirp chirp")
```

Class Parrot:

main.py | bird.py | sparrow.py | parrot.py | birdCage.py

```python
from bird import Bird

class Parrot(Bird):
    def make_sound(self) -> None:
        print("Tweet tweet")
```

Class birdcage:

main.py | bird.py | sparrow.py | parrot.py | birdCage.py

```python
from typing import List
from bird import Bird

class BirdCage:
    def make_bird_sounds(self, birds: List[Bird]) -> None:
        for bird in birds:
            bird.make_sound()

```

Main file:

main.py | bird.py | sparrow.py | parrot.py | birdCage.py

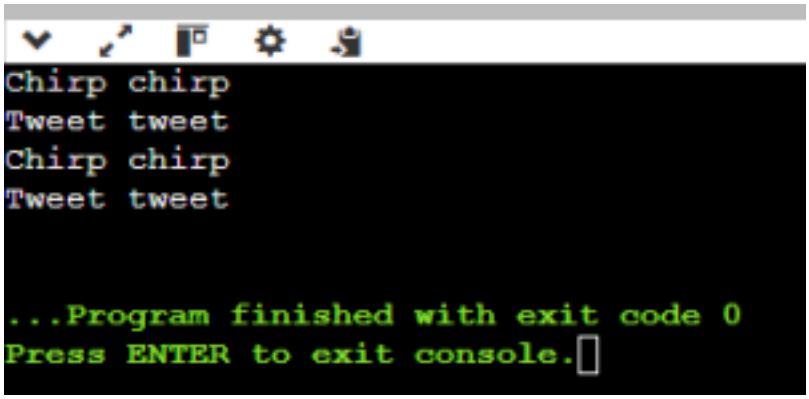
```python
from sparrow import Sparrow
from parrot import Parrot
from birdCage import BirdCage

b1 = Sparrow()
b2 = Parrot()
cage = BirdCage()

birds = [b1, b2, Sparrow(), Parrot()]
cage.make_bird_sounds(birds)

```

Output:


Chirp chirp
Tweet tweet
Chirp chirp
Tweet tweet
...Program finished with exit code 0
Press ENTER to exit console.