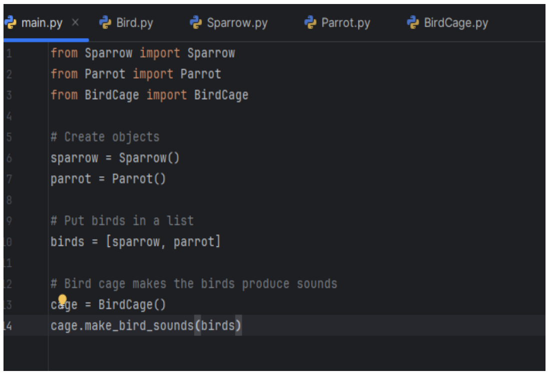
Hannizel,Dumangas

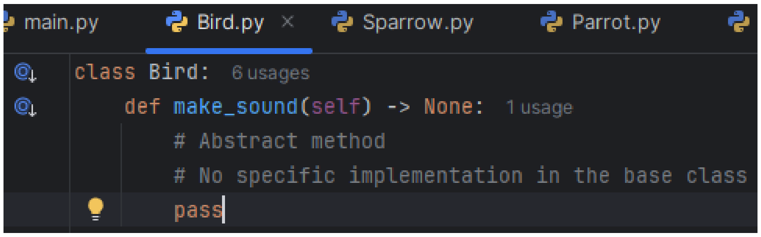
BSCS – C204

Simple Polymorphism:

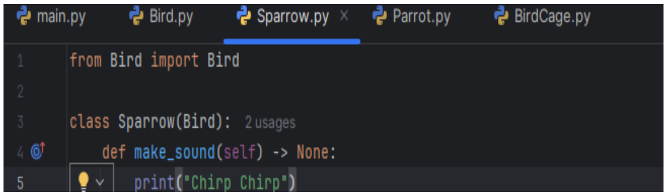
Source Code:



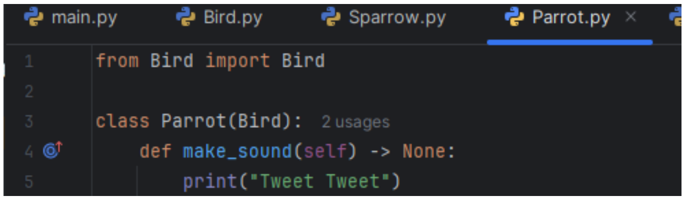
```python
from Sparrow import Sparrow
from Parrot import Parrot
from BirdCage import BirdCage

# Create objects
sparrow = Sparrow()
parrot = Parrot()

# Put birds in a list
birds = [sparrow, parrot]

# Bird cage makes the birds produce sounds
cage = BirdCage()
cage.make_bird_sounds(birds)
```



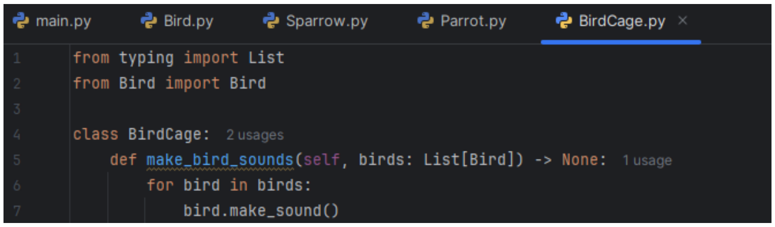
```python
class Bird:
    def make_sound(self) -> None:
        # Abstract method
        # No specific implementation in the base class
        pass
```



```python
from Bird import Bird

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

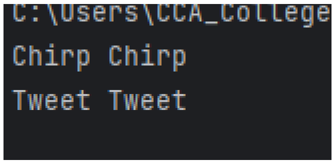


```python
from Bird import Bird

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



```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()
```

Sample output:



```
C:\Users\CCA_College
Chirp Chirp
Tweet Tweet
```