

Nima Memarzadeh
Module 7.2 Assignment

city_functions_v1.py > ...

```
1 def city_country(city, country):
2     """Return a formatted string of city and country."""
3     return f"{city.title()}, {country.title()}"
4
5 # Call the function at least three times
6 print("Initial Function Calls:")
7 print(city_country('santiago', 'chile'))
8 print(city_country('tokyo', 'japan'))
9 print(city_country('paris', 'france'))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

- nimamemarzadeh@Nimas-MacBook-Pro module-7 % python3 city_functions_v1.py
Initial Function Calls:
Santiago, Chile
Tokyo, Japan
Paris, France
- nimamemarzadeh@Nimas-MacBook-Pro module-7 %

test_cities_v1.py > ...

```
1  import unittest
2  from city_functions_v1 import city_country
3
4  class CityTestCase(unittest.TestCase):
5      def test_city_country(self):
6          """Test that city_country returns the correct string."""
7          formatted_city = city_country('santiago', 'chile')
8          self.assertEqual(formatted_city, 'Santiago, Chile')
9
10 if __name__ == '__main__':
11     unittest.main()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
● nimamemarzadeh@Nimas-MacBook-Pro module-7 % python3 test_cities_v1.py
Initial Function Calls:
Santiago, Chile
Tokyo, Japan
Paris, France
.
-----
Ran 1 test in 0.000s

OK
○ nimamemarzadeh@Nimas-MacBook-Pro module-7 %
```

city_functions_v2.py > ...

```
1 def city_country(city, country, population=None):
2     """Return a formatted string of city and country with optional population."""
3     if population:
4         return f"{city.title()}, {country.title()} - population {population}"
5     return f"{city.title()}, {country.title()}"
6
7 # Call the function with different parameter combinations
8 print("Population Optional Function Calls:")
9 print(city_country('santiago', 'chile'))
10 print(city_country('santiago', 'chile', 5000000))
11 print(city_country('tokyo', 'japan', 13929286))
12 print(city_country('paris', 'france', 2140526))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

zsh + ▢

```
● nimamemarzadeh@Nimas-MacBook-Pro module-7 % python3 city_functions_v2.py
Population Optional Function Calls:
Santiago, Chile
Santiago, Chile - population 5000000
Tokyo, Japan - population 13929286
Paris, France - population 2140526
○ nimamemarzadeh@Nimas-MacBook-Pro module-7 %
```

test_cities_v2.py > ...

```
1 import unittest
2 from city_functions_v2 import city_country
3
4 class CityTestCase(unittest.TestCase):
5     def test_city_country(self):
6         """Test that city_country returns the correct string."""
7         formatted_city = city_country('santiago', 'chile')
8         self.assertEqual(formatted_city, 'Santiago, Chile')
9
10    def test_city_country_population(self):
11        """Test city_country with population."""
12        formatted_city = city_country('santiago', 'chile', 5000000)
13        self.assertEqual(formatted_city, 'Santiago, Chile - population 5000000')
14
15 if __name__ == '__main__':
16     unittest.main()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

zsh + -

```
● nimamemarzadeh@Nimas-MacBook-Pro module-7 % python3 test_cities_v2.py
Population Optional Function Calls:
Santiago, Chile
Santiago, Chile - population 5000000
Tokyo, Japan - population 13929286
Paris, France - population 2140526
..
```

Ran 2 tests in 0.000s

OK

○ nimamemarzadeh@Nimas-MacBook-Pro module-7 %

city_functions_v3.py > ...

```
1 def city_country(city, country, population=None, language=None):
2     """Return a formatted string of city and country with optional population and language."""
3     if population and language:
4         return f"{city.title()}, {country.title()} - population {population}, {language}"
5     elif population:
6         return f"{city.title()}, {country.title()} - population {population}"
7     return f"{city.title()}, {country.title()}"
8
9 # Final function call demonstration
10 print("Final Function Calls Demonstration:")
11 print("No optional parameters:", city_country('santiago', 'chile'))
12 print("With population:", city_country('santiago', 'chile', 5000000))
13 print("With population and language:", city_country('santiago', 'chile', 5000000, 'Spanish'))
14 print("With different cities:")
15 print(city_country('tokyo', 'japan', 13929286, 'Japanese'))
16 print(city_country('paris', 'france', 2140526, 'French'))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
● nimamemarzadeh@Nimas-MacBook-Pro module-7 % python3 city_functions_v3.py
Final Function Calls Demonstration:
No optional parameters: Santiago, Chile
With population: Santiago, Chile - population 5000000
With population and language: Santiago, Chile - population 5000000, Spanish
With different cities:
Tokyo, Japan - population 13929286, Japanese
Paris, France - population 2140526, French
○ nimamemarzadeh@Nimas-MacBook-Pro module-7 %
```

test_cities_v3.py > ...

```
1  import unittest
2  from city_functions_v3 import city_country
3
4  class CityTestCase(unittest.TestCase):
5      def test_city_country(self):
6          """Test that city_country returns the correct string."""
7          formatted_city = city_country('santiago', 'chile')
8          self.assertEqual(formatted_city, 'Santiago, Chile')
9
10     def test_city_country_population(self):
11         """Test city_country with population."""
12         formatted_city = city_country('santiago', 'chile', 5000000)
13         self.assertEqual(formatted_city, 'Santiago, Chile - population 5000000')
14
15     def test_city_country_population_language(self):
16         """Test city_country with population and language."""
17         formatted_city = city_country('santiago', 'chile', 5000000, 'Spanish')
18         self.assertEqual(formatted_city, 'Santiago, Chile - population 5000000, Spanish')
19
20 if __name__ == '__main__':
21     unittest.main()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

zsh + -

```
● nimamemarzadeh@Nimas-MacBook-Pro module-7 % python3 test_cities_v3.py
Final Function Calls Demonstration:
No optional parameters: Santiago, Chile
With population: Santiago, Chile - population 5000000
With population and language: Santiago, Chile - population 5000000, Spanish
With different cities:
Tokyo, Japan - population 13929286, Japanese
Paris, France - population 2140526, French
...
-----
Ran 3 tests in 0.000s

OK
○ nimamemarzadeh@Nimas-MacBook-Pro module-7 %
```