

# Unit-Testing 2: Mocking and Stubbing with Dependency Injection

1.

```
def test_get_country_code():
    test_countries = [
        {"name": "United Kingdom", "alpha3Code": "UK"},
        {"name": "United States", "alpha3Code": "USA"}
    ]

    test_name = "United Kingdom"
    expected = "UK"
    actual = get_country_code(test_countries, test_name)

    assert expected == actual
    print("test_get_country_code: Passed")
```

2.

```
def test_get_country_currency():
    test_countries = [
        {"name": "United Kingdom", "currencies": [{"code": "GBP"}]},
        {"name": "United States", "currencies": [{"code": "USD"}]}
    ]

    test_name = "United Kingdom"
    expected = "GBP"
    actual = get_country_currency(test_countries, test_name)

    assert expected == actual
    print("test_get_country_currency: Passed")
```

3.

```
def test_transform():
    def mock_get_countries():
        return [{"name": "United Kingdom", "alpha3Code": "UK", "currencies": [{"code": "GBP"}]}]

    test_name = "United Kingdom"
    expected = {"name": "United Kingdom", "country_code": "UK", "currency_code": "GBP"}
    actual = transform(test_name, mock_get_countries)

    assert actual == expected
    print("test_transform: Passed")
```

4.

```
def test_show_country_info():
    def mock_get_countries():
        return [{"name": "United Kingdom", "alpha3Code": "UK", "currencies": [{"code": "GBP"}]}]

    def mock_print(*args):
        pass

    def mock_input(msg):
        return 0

    expected = {"name": "United Kingdom", "country_code": "UK", "currency_code": "GBP"}
    actual = show_country_info(mock_get_countries, transform, mock_print, mock_input)

    assert expected == actual
    print("test_show_country_info: Passed")
```