

### Caso de Prueba Unitaria 1 (Caso de éxito):

```
1 import unittest
2 from Metodos_distancias import DistanceCalculatorFactory
3
4 #CASO DE ÉXITO
5 class TestDistanceCalculator1(unittest.TestCase):
6
7     def test_CSV_method(self):
8         distance = DistanceCalculatorFactory.create_distance_calculator("CSV", "Kolkata", "India", "Lagos", "Nigeria")
9         self.assertAlmostEqual(distance, 9219.74141992384, places=2)
10
11     def test_API_method(self):
12         distance = DistanceCalculatorFactory.create_distance_calculator("API", "Kolkata", "India", "Lagos", "Nigeria")
13         self.assertAlmostEqual(distance, 9218.040829550899, places=2)
14
15     def test_MOOK_method(self):
16         distance = DistanceCalculatorFactory.create_distance_calculator("MOOK", "Kolkata", "India", "Lagos", "Nigeria")
17         self.assertAlmostEqual(distance, 9219.74141992384, places=2)
18
```

Se observa el script para el código de test 1. Aquí se considera el caso de éxito para calcular distancias con la forma CSV, API Y MOCK entre las mismas ciudades. Como ciudad 1 se ingresa Kolkata, como país 1 está India, y como ciudad 2, Lagos en el país Nigeria. Los datos de ciudades y países son iguales para todos los formatos.

### Caso de Prueba Unitaria 2 (Caso extremo - una de las ciudades no existe) :

```
19 #CASO EXTREMO 1:
20 class TestDistanceCalculator2(unittest.TestCase):
21
22     def test_nonexistent_city_csv(self):
23         # Test when a city does not exist in the CSV data
24         result = DistanceCalculatorFactory.create_distance_calculator("CSV", "Lima123", "Peru", "Quito", "Ecuador")
25         self.assertEqual(result, "No se encuentra la ciudad en la base de datos")
26
27     def test_nonexistent_city_api(self):
28         # Test when a city does not exist in the Nominatim API
29         result = DistanceCalculatorFactory.create_distance_calculator("API", "Lima123", "Peru", "Quito", "Ecuador")
30         self.assertEqual(result, "No se encuentra la ciudad en la base de datos")
31
```

Se puede observar el script para el código de test 2 con CSV y la API. Donde el input de Ciudad 1 es "Lima123" y como se sabe, no existe esa ciudad en el CSV ni en la API

### Caso de Prueba Unitaria 3 (Caso extremo - entregar la misma ciudad dos veces) :

A continuación se inserta el script para el código de test. Aquí se considera el caso de extremo 2.

```
31
32 #CASO EXTREMO 2:
33
34 class TestDistanceCalculator3(unittest.TestCase):
35
36     def test_repeated_city_csv(self):
37         # Test when both country and citys repeated
38
39         result = DistanceCalculatorFactory.create_distance_calculator("CSV", "Kolkata", "India", "Kolkata", "India")
40         self.assertEqual(result, "Las ciudades son iguales")
41
42     def test_repeated_city_api(self):
43         # Test when both country and citys repeated
44
45         result = DistanceCalculatorFactory.create_distance_calculator("API", "Kolkata", "India", "Kolkata", "India")
46         self.assertEqual(result, "Las ciudades son iguales")
47
48
49
50 if __name__ == '__main__':
51     unittest.main()
```

En la imagen se puede observar el script para el código de test con CSV y la API. Donde el input de Ciudad 1 es "Kolkata" y el input del país 1 es "India", de igual forma para la Ciudad 2 y país 2 respectivamente.

### Resultados de los test unitarios:

Todos nuestros tests están correctos. Por lo tanto, retorna los siguientes mensajes:

```
● → Ing_EC2 git:(main) python3 -u "/Users/leoisidro/Library/CloudStorage/OneDrive-UNIVERSIDADDEINGENIERIAYTECNOLOGIA/UTEC_MAC/ING.SOFTWARE/EC2/Ing_EC2/pruebas.py"
.....
Ran 7 tests in 2.792s

OK
○ → Ing_EC2 git:(main) █
```

### Casos de Prueba Manual:

Test Case	Precondition	Test Steps	Test Data	Expected Result
1.Verificar que las ciudades ingresadas pertenezcan a sus países respectivos y que la distancia entre ellos sea la misma con la impuesta en el código en cualquiera de los tests para el cálculo de distancia	Que el usuario tenga una conexión de red y que haya clonado nuestro repositorio	<ol style="list-style-type: none"> <li>1. Ejecutar la aplicación</li> <li>2. Ingresa "Kolkata" en la primera casilla de ciudad.</li> <li>3. Ingresa "India" en la primera casilla del país.</li> <li>4. Ingresa "Lagos" en la segunda casilla de ciudad.</li> <li>5. Ingresa "Nigeria" en la segunda casilla de país.</li> <li>6. Haz clic en el botón "CSV", "MOCK" o "API".</li> </ol>	<p>Ciudad 1: Kolkata País 1: India</p> <p>Ciudad 2: Lagos País 2: Nigeria</p> <p>Opción para calcular distancia: CSV, MOCK o API</p>	<p>El usuario deberá observar la distancia en km entre estas ciudades que se encuentran en la data.</p> <p>- Si escoge la opción "CSV", la distancia debe ser: 9219.74141992384 km</p> <p>- Si escoge la opción "MOCK" la distancia debe ser: 9219.74141992384 km</p> <p>- Si escoge la opción "API" la distancia debe ser: 9218.040829550899 km</p> <p>Caso de éxito.</p>
2. Verificar que ambas ciudades con sus respectivos países existan en los distintos tests para el cálculo de distancia	Que el usuario tenga una conexión de red y que haya clonado nuestro repositorio	<ol style="list-style-type: none"> <li>1. Ejecutar la aplicación</li> <li>2. Ingresa "Lima123" en la primera casilla de ciudad.</li> <li>3. Ingresa "Peru" en la primera casilla del país.</li> <li>4. Ingresa "Quito" en la segunda casilla de ciudad.</li> <li>5. Ingresa "Ecuador" en la segunda casilla de país.</li> <li>6. Haz clic en el botón "CSV" o</li> </ol>	<p>Ciudad 1: Lima123 País 1: Peru</p> <p>Ciudad 2: Quito País 2: Ecuador</p> <p>Opción para calcular distancia: CSV o API</p>	<p>El usuario deberá recibir un mensaje de error para que complete nuevamente los inputs respectivos porque no existen en los distintos tests para el cálculo de distancia.</p> <p>Caso extremo.</p>

		"API".		
3.Verificar que ambas ciudades con sus respectivos países no se repitan.	Que el usuario tenga una conexión de red y que haya clonado nuestro repositorio	<ol style="list-style-type: none"> <li>1. Ejecutar la aplicación</li> <li>2. Ingresa "Kolkata" en la primera casilla de ciudad.</li> <li>3. Ingresa "India" en la primera casilla del país.</li> <li>4. Ingresa "Kolkata" en la segunda casilla de ciudad.</li> <li>5. Ingresa "India" en la segunda casilla de país.</li> <li>6. Haz clic en el botón "CSV" o "API".</li> </ol>	<p>Ciudad 1: Kolkata País 1: India</p> <p>Ciudad 2: Kolkata País 2: India</p> <p>Opción para calcular distancia: CSV o API</p>	<p>El usuario deberá recibir un mensaje de error para que complete nuevamente los inputs respectivos ya que no deberían repetirse.</p> <p>Caso extremo.</p>

Testing para calcular distancias entre Kolkata y Lagos con Mock:

Calculadora de Distancia entre Ciudades

Ciudad 1:  
Kolkata

País 1:  
India

Ciudad 2:  
Lagos

País 2:  
Nigeria

Método de Cálculo:

**Resultado del cálculo de distancia**

Distancia entre Kolkata, India y Lagos, Nigeria utilizando MOOK: 9219.74141992384 km

Testing para calcular distancias entre Istanbul y Karachi con Mock:

The screenshot shows a web browser at the address 127.0.0.1:5000/distance. The page has a green header with the title "Calculadora de Distancia entre Ciudades". Below the header, there are four input fields: "Ciudad 1:" with "Istanbul", "País 1:" with "Turkey", "Ciudad 2:" with "Karachi", and "País 2:" with "Pakistan". Below these fields are three buttons: "CSV" (green), "API" (green), and "Mock" (blue). The "Mock" button is selected. At the bottom, a green section titled "Resultado del cálculo de distancia" displays the text: "Distancia entre Istanbul, Turkey y Karachi, Pakistan utilizando MOOK: 3937.9403519773987 km".

Testing para calcular distancias entre Istanbul y Karachi con API:

The screenshot shows the same web browser and application as above, but with the "API" button selected. The input fields remain the same: "Ciudad 1:" (Istanbul), "País 1:" (Turkey), "Ciudad 2:" (Karachi), and "País 2:" (Pakistan). The "API" button is now highlighted in blue. The result section at the bottom shows: "Resultado del cálculo de distancia" and "Distancia entre Istanbul, Turkey y Karachi, Pakistan utilizando API: 3937.2595452559617 km".

Testing para calcular distancias entre Kolkata y Lagos con API:

The screenshot shows a web browser at the URL 127.0.0.1:5000/distance. The page has a green header with the title "Calculadora de Distancia entre Ciudades". The main content area has a black background with white text and input fields. It contains four input fields for "Ciudad 1:" (Kolkata), "País 1:" (India), "Ciudad 2:" (Lagos), and "País 2:" (Nigeria). Below these is a "Método de Cálculo:" section with three buttons: "CSV" (green), "API" (blue), and "Mock" (green). The "API" button is selected. At the bottom, a light gray footer displays the "Resultado del cálculo de distancia" as "Distancia entre Kolkata, India y Lagos, Nigeria utilizando API: 9218.040829550899 km".

Testing para calcular distancias entre Kolkata y Lagos con CSV:

This screenshot is identical to the one above, but the "CSV" button in the "Método de Cálculo:" section is now selected (highlighted in blue). Consequently, the result in the footer is "Distancia entre Kolkata, India y Lagos, Nigeria utilizando CSV: 9219.74141992384 km".

Testing para calcular distancias entre Istanbul y Karachi con CSV:

The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/distance". The browser's tab bar includes several open tabs: "Gmail", "Documentos de Go...", "Traductor de Google", "Documentation", "UTEC-CanvasTablero", "UTEC - Platform", "UTEC 2023-2", "Universidad de Ing...", "UTEC-SIS", "CS <3", and "Todos los marcadores".

The web application has a green header with the title "Calculadora de Distancia entre Ciudades". Below the header, on a black background, are four input fields for city and country information:

- Ciudad 1:** Istanbul
- País 1:** Turkey
- Ciudad 2:** Karachi
- País 2:** Nigeria

Below these fields is a section labeled "Método de Cálculo:" with three buttons: "CSV" (highlighted in blue), "API" (green), and "Mock" (green).

The bottom section of the application, on a light gray background, displays the result:

**Resultado del cálculo de distancia**  
Distancia entre Istanbul, Turkey y Karachi, Pakistan utilizando CSV: 3937.9403519773987 km