



Tecnológico
de Monterrey



MAESTRIA EN INTELIGENCIA
ARTIFICIAL APLICADA

Tarea 6.2 Ejercicio de programación 3

BERNAGA TORRES, ASTRID R. A01793080

PRUEBAS DE SOFTWARE Y
ASEGURAMIENTO DE LA CALIDAD

PROFR. DR GERARDO PADILLA ZÁRATE

PROFR. TUTOR. MTRO. YAMIL ADRIÁN ELÍAS SOTO



hotel.py

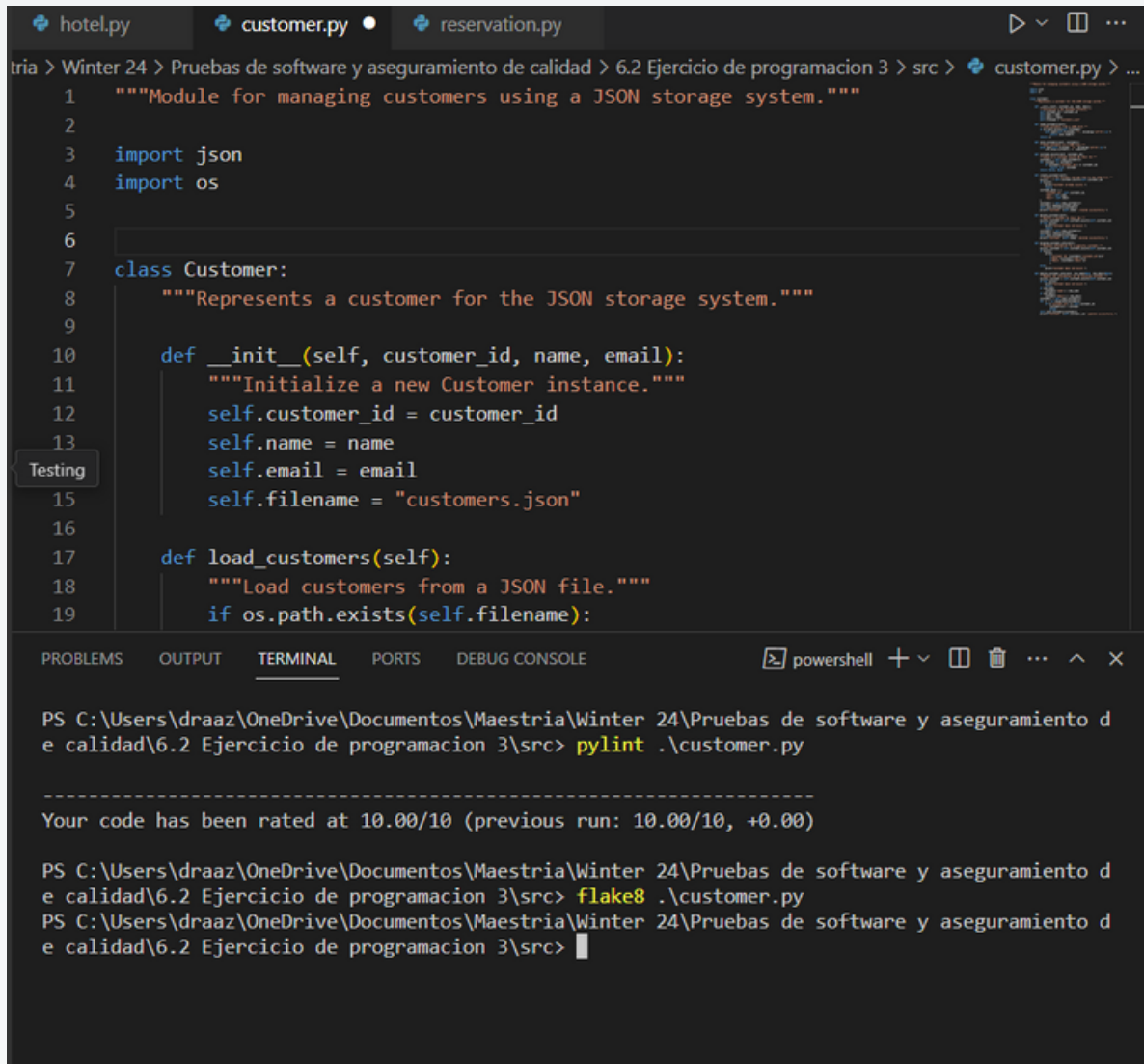
```
Maestria > Winter 24 > Pruebas de software y aseguramiento de calidad > 6.2 Ejercicio de programacion 3 > src > hotel.py
8 class Hotel:
80     def reserve_room(self, room_id, customer_id):
89         room['reserved'] = True
90         hotel_data['reservations'].append(
91             {'room_id': room_id, 'customer_id': customer_id})
92         self.save_to_file(hotel_data)
93         print(f"Room {room_id} reserved successfully.")
94         return
95     print("Room not available or already reserved.")
96
97     def cancel_reservation(self, room_id, customer_id):
98         """Cancel a room reservation for a customer."""
99         if not self.hotel_exists():
100             print("Hotel does not exist.")
101             return
102         hotel_data = self.load_from_file()
103         for reservation in hotel_data['reservations']:
104             if (reservation['room_id'] == room_id and
105                 reservation['customer_id'] == customer_id):
```

El código ha sido calificado con una puntuación perfecta de 10.00/10.

Esto significa que pylint no ha encontrado ningún problema en el código en términos de convenciones de estilo, errores o posibles problemas.

También se ejecutó flake8 y no se generaron comentarios, lo que indica que el código cumple con las convenciones de estilo definidas.

customer.py



```
1 """Module for managing customers using a JSON storage system."""
2
3 import json
4 import os
5
6
7 class Customer:
8     """Represents a customer for the JSON storage system."""
9
10    def __init__(self, customer_id, name, email):
11        """Initialize a new Customer instance."""
12        self.customer_id = customer_id
13        self.name = name
14        self.email = email
15        self.filename = "customers.json"
16
17    def load_customers(self):
18        """Load customers from a JSON file."""
19        if os.path.exists(self.filename):
```

PROBLEMS OUTPUT **TERMINAL** PORTS DEBUG CONSOLE

PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento de calidad\6.2 Ejercicio de programacion 3\src> pylint .\customer.py

Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)

PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento de calidad\6.2 Ejercicio de programacion 3\src> flake8 .\customer.py

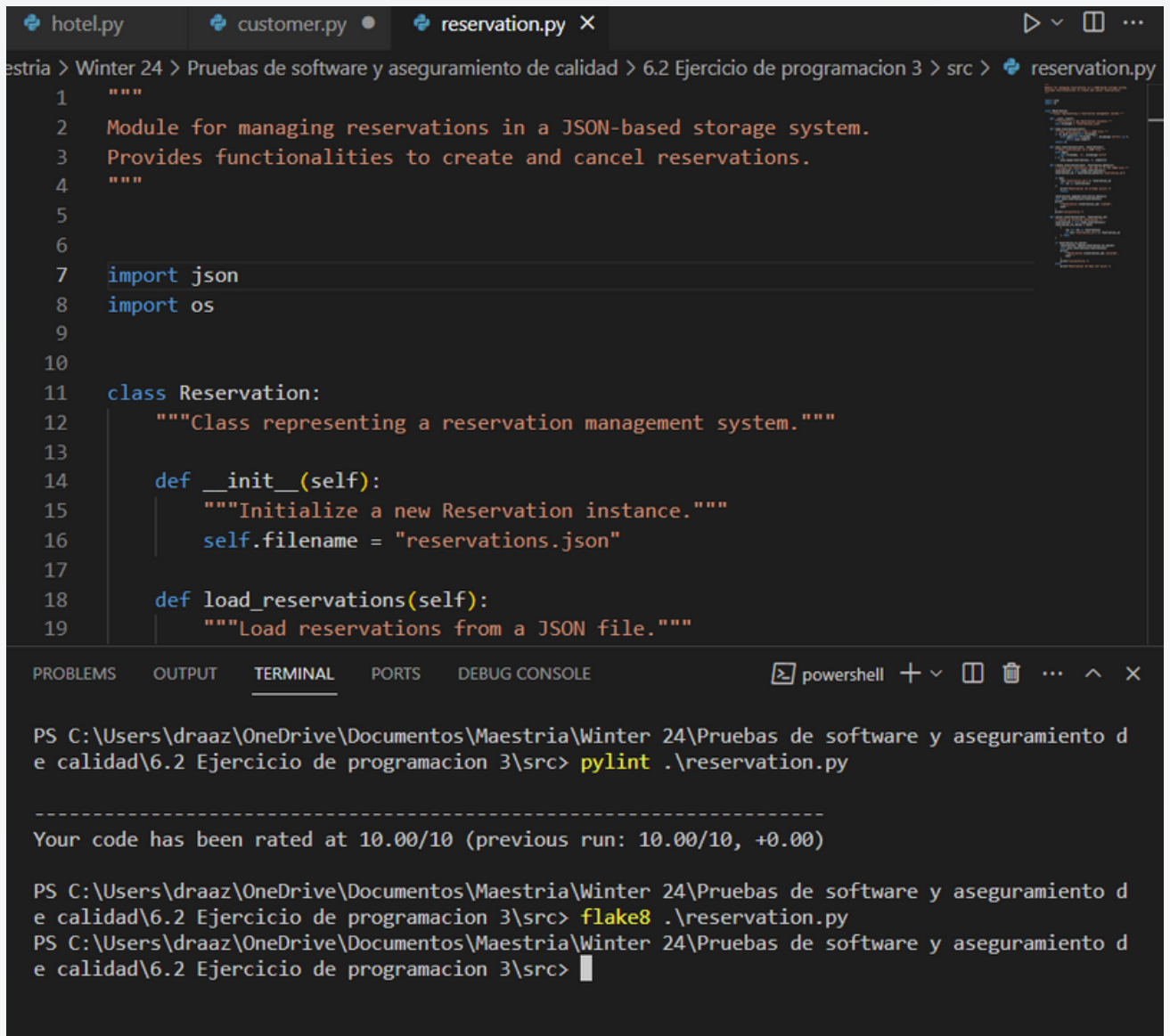
PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento de calidad\6.2 Ejercicio de programacion 3\src> █

El código ha sido calificado con una puntuación perfecta de 10.00/10.

Esto significa que pylint no ha encontrado ningún problema en el código en términos de convenciones de estilo, errores o posibles problemas.

También se ejecutó flake8 y no se generaron comentarios, lo que indica que el código cumple con las convenciones de estilo definidas.

reservation.py



```
1  """
2  Module for managing reservations in a JSON-based storage system.
3  Provides functionalities to create and cancel reservations.
4  """
5
6
7  import json
8  import os
9
10
11 class Reservation:
12     """Class representing a reservation management system."""
13
14     def __init__(self):
15         """Initialize a new Reservation instance."""
16         self.filename = "reservations.json"
17
18     def load_reservations(self):
19         """Load reservations from a JSON file."""
```

PROBLEMS OUTPUT **TERMINAL** PORTS DEBUG CONSOLE

```
PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento d
e calidad\6.2 Ejercicio de programacion 3\src> pylint .\reservation.py

-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)

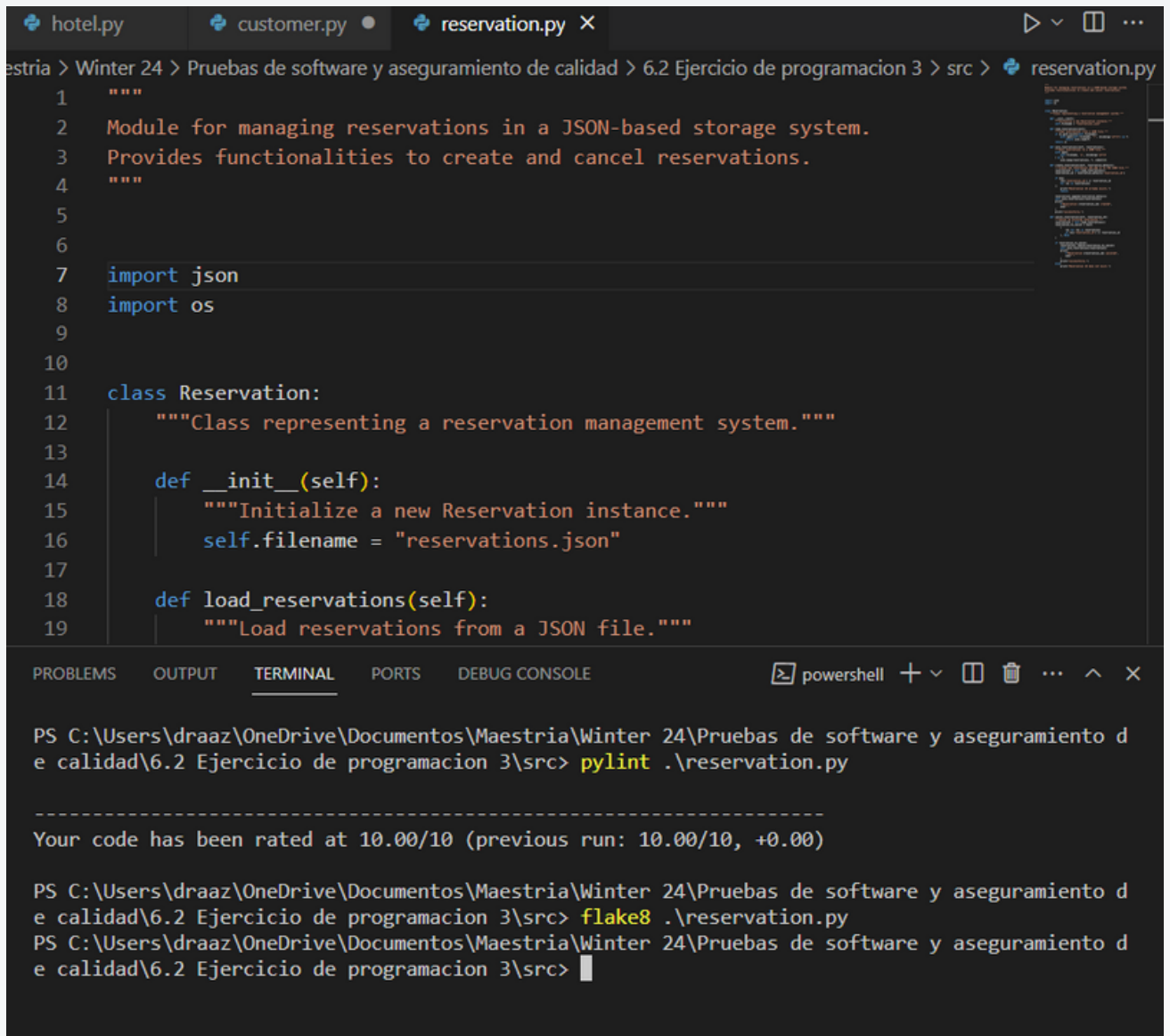
PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento d
e calidad\6.2 Ejercicio de programacion 3\src> flake8 .\reservation.py
PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento d
e calidad\6.2 Ejercicio de programacion 3\src>
```

El código ha sido calificado con una puntuación perfecta de 10.00/10.

Esto significa que pylint no ha encontrado ningún problema en el código en términos de convenciones de estilo, errores o posibles problemas.

También se ejecutó flake8 y no se generaron comentarios, lo que indica que el código cumple con las convenciones de estilo definidas.

test_hotel.py



```
Maestria > Winter 24 > Pruebas de software y aseguramiento de calidad > 6.2 Ejercicio de programacion 3 > src > reservation.py
1  """
2  Module for managing reservations in a JSON-based storage system.
3  Provides functionalities to create and cancel reservations.
4  """
5
6
7  import json
8  import os
9
10
11  class Reservation:
12      """Class representing a reservation management system."""
13
14      def __init__(self):
15          """Initialize a new Reservation instance."""
16          self.filename = "reservations.json"
17
18      def load_reservations(self):
19          """Load reservations from a JSON file."""
```

```
PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento de calidad\6.2 Ejercicio de programacion 3\src> pylint .\reservation.py
-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)

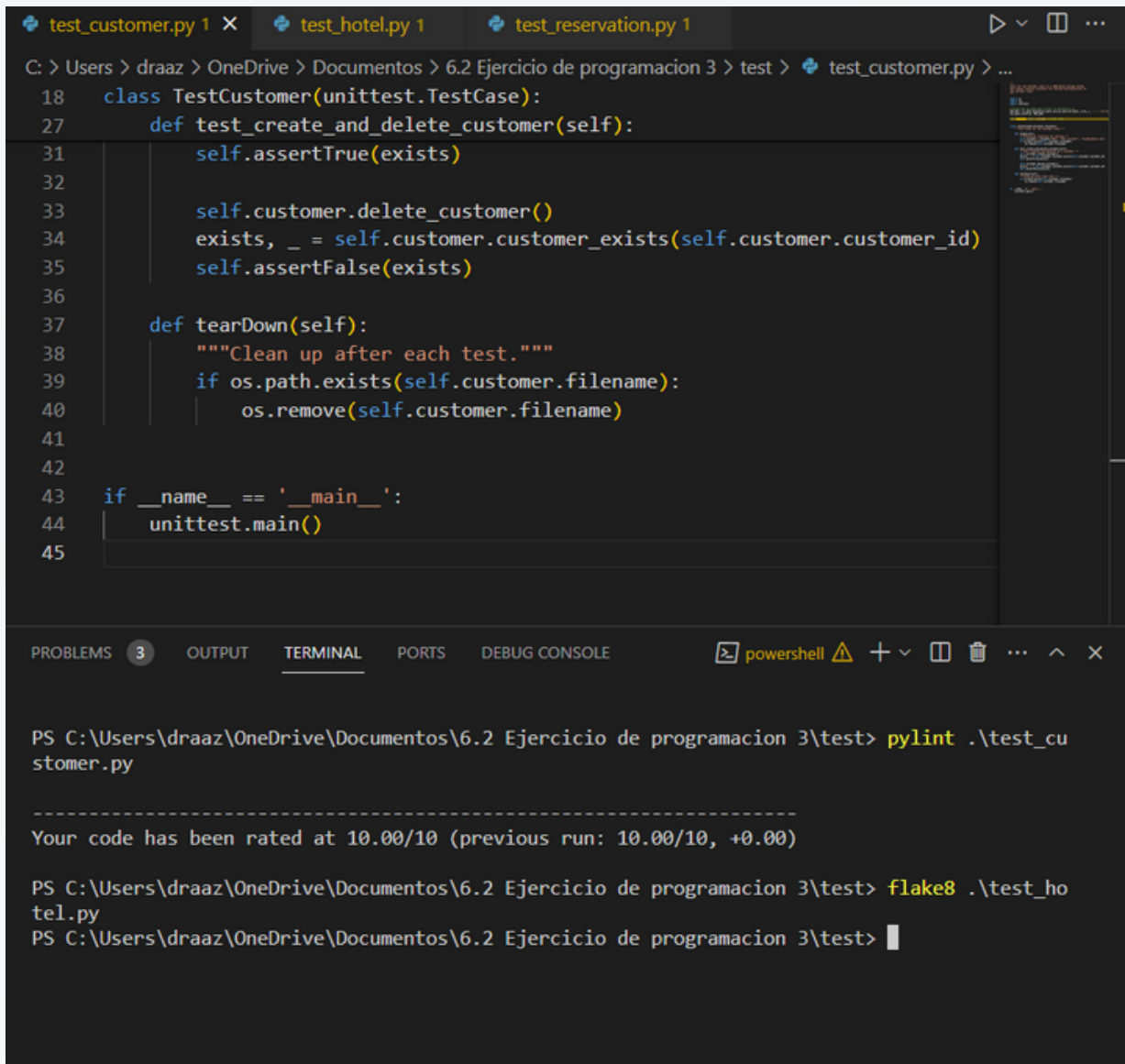
PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento de calidad\6.2 Ejercicio de programacion 3\src> flake8 .\reservation.py
PS C:\Users\draaz\OneDrive\Documentos\Maestria\Winter 24\Pruebas de software y aseguramiento de calidad\6.2 Ejercicio de programacion 3\src>
```

El código ha sido calificado con una puntuación perfecta de 10.00/10.

Esto significa que pylint no ha encontrado ningún problema en el código en términos de convenciones de estilo, errores o posibles problemas.

También se ejecutó flake8 y no se generaron comentarios, lo que indica que el código cumple con las convenciones de estilo definidas.

test_customer.py



```
test_customer.py 1 x test_hotel.py 1 test_reservation.py 1
C: > Users > draaz > OneDrive > Documentos > 6.2 Ejercicio de programacion 3 > test > test_customer.py > ...
18 class TestCustomer(unittest.TestCase):
27     def test_create_and_delete_customer(self):
31         self.assertTrue(exists)
32
33         self.customer.delete_customer()
34         exists, _ = self.customer.customer_exists(self.customer.customer_id)
35         self.assertFalse(exists)
36
37     def tearDown(self):
38         """Clean up after each test."""
39         if os.path.exists(self.customer.filename):
40             os.remove(self.customer.filename)
41
42
43 if __name__ == '__main__':
44     unittest.main()
45
```

```
PROBLEMS 3 OUTPUT TERMINAL PORTS DEBUG CONSOLE powershell
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3\test> pylint .\test_customer.py

-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)

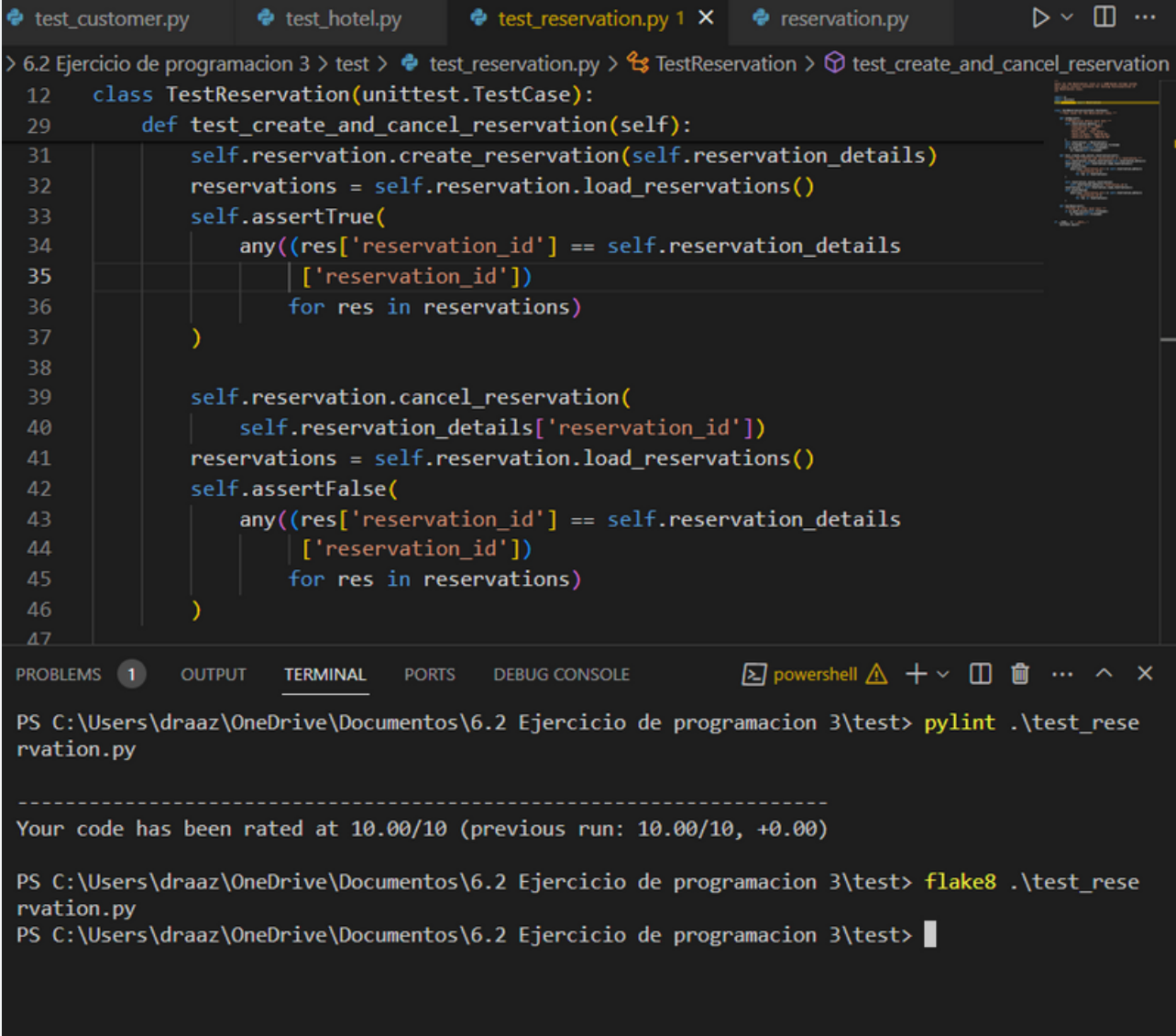
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3\test> flake8 .\test_hotel.py
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3\test>
```

El código ha sido calificado con una puntuación perfecta de 10.00/10.

Esto significa que pylint no ha encontrado ningún problema en el código en términos de convenciones de estilo, errores o posibles problemas.

También se ejecutó flake8 y no se generaron comentarios, lo que indica que el código cumple con las convenciones de estilo definidas.

test_reservation.py



The screenshot shows a code editor with several tabs: test_customer.py, test_hotel.py, test_reservation.py (active), and reservation.py. The active tab displays the following Python code:

```
> 6.2 Ejercicio de programacion 3 > test > test_reservation.py > TestReservation > test_create_and_cancel_reservation
12 class TestReservation(unittest.TestCase):
29     def test_create_and_cancel_reservation(self):
31         self.reservation.create_reservation(self.reservation_details)
32         reservations = self.reservation.load_reservations()
33         self.assertTrue(
34             any((res['reservation_id'] == self.reservation_details
35                 ['reservation_id'])
36                 for res in reservations)
37         )
38
39         self.reservation.cancel_reservation(
40             self.reservation_details['reservation_id'])
41         reservations = self.reservation.load_reservations()
42         self.assertFalse(
43             any((res['reservation_id'] == self.reservation_details
44                 ['reservation_id'])
45                 for res in reservations)
46         )
47
```

Below the code editor is a terminal window with the following output:

```
PROBLEMS 1 OUTPUT TERMINAL PORTS DEBUG CONSOLE powershell
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3\test> pylint .\test_reservation.py

-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)

PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3\test> flake8 .\test_reservation.py
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3\test>
```

El código ha sido calificado con una puntuación perfecta de 10.00/10.

Esto significa que pylint no ha encontrado ningún problema en el código en términos de convenciones de estilo, errores o posibles problemas.

También se ejecutó flake8 y no se generaron comentarios, lo que indica que el código cumple con las convenciones de estilo definidas.

Pruebas unitarias

test_hotel

```
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3> python -m unittest discover -s test -p "test_hotel.py"
Hotel 'Test Hotel' created successfully.
.Hotel 'Test Hotel' created successfully.
.Hotel 'Test Hotel' created successfully.
Hotel 'Test Hotel' deleted successfully.
.Hotel 'Test Hotel' created successfully.
Hotel 'Test Hotel' updated successfully.
.Hotel 'Test Hotel' created successfully.
Room 001 added successfully.
Room 001 reserved successfully.
Reservation for room 001 canceled successfully.
.Hotel 'Test Hotel' created successfully.
.
-----
Ran 6 tests in 0.011s

OK
```

test_customer

```
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3> python -m unittest discover -s test -p "test_customer.py"
Customer 'Test Customer' created successfully.
Customer 'Test Customer' deleted successfully.
.Customer 'Test Customer' created successfully.
.Customer 'Test Customer' created successfully.
Customer '001' updated successfully.
.
-----
Ran 3 tests in 0.007s

OK
```

test_reservation

```
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3> python -m unittest discover -s test -p "test_reservation.py"
Reservation R001 created successfully.
Reservation R001 canceled successfully.
.
-----
Ran 1 test in 0.005s

OK
```

Capturas de pantalla que muestran la ejecución de pruebas unitarias.

Pruebas de cobertura

```
PS C:\Users\draaz\OneDrive\Documentos\6.2 Ejercicio de programacion 3> coverage run -m unittest discover
-s test
Customer 'Test Customer' created successfully.
Customer 'Test Customer' deleted successfully.
.Customer 'Test Customer' created successfully.
.Customer 'Test Customer' created successfully.
Customer '001' updated successfully.
.Hotel 'Test Hotel' created successfully.
.Hotel 'Test Hotel' created successfully.
.Hotel 'Test Hotel' created successfully.
Hotel 'Test Hotel' deleted successfully.
.Hotel 'Test Hotel' created successfully.
Hotel 'Test Hotel' updated successfully.
.Hotel 'Test Hotel' created successfully.
Room 001 added successfully.
Room 001 reserved successfully.
Reservation for room 001 canceled successfully.
.Hotel 'Test Hotel' created successfully.
.Reservation R001 created successfully.
Reservation R001 canceled successfully.
.
-----
Ran 10 tests in 0.045s

OK
Name                               Stmt  Miss  Cover2 Ejercicio de programacion 3>
-----
src\customer.py                    62     7    89%
src\hotel.py                       80    11    86%
src\reservation.py                 32     3    91%
-----
TOTAL                             174    21    88%
```

coverage report

Capturas de pantalla que muestran la ejecución y un informe de cobertura de código mayor a 85%



LIGA DE GITHUB

<https://github.com/Astroцитos/A01793080A6.2>

