Тестирование программных модулей

**Організація тестування**

Для тестування використовувалося середовище розробки IntelliJ IDEA Ultimate та фреймворк автоматичного тестування Junit 5. Написання тестів велося на основі проекту тестових наборів, описаних в лабораторній роботі №8.

Створення тестів велося у спеціальній директорії проекту з назвами, що відображають клас, що тестується типу OrderTest. Кожен окремий набір тестових даних виділявся анотацією @Test та мав включати очікувані та реальні дані, на основі порівняння яких вирішувалося, чи був тест пройдений.

**Тестування**

Функції для тестування обиралися виходячи з найбільшої вірогідності виникнення в них помилки.

Специфікація функції getProfile класу SignUp має наступний опис.

Вхідні параметри відсутні.

Вихідні дані: текстовий рядок, що включае собою дані профіляю

Таблиця 1 – Опис тестових наборів для функції getProfile

|  |  |  |
| --- | --- | --- |
| **№ тесту** | **Опис значень початкових даних** | **Опис очікуваних значень результату** |
| 1 | *Поточні дані - Profile = {[id\_user=1, id\_post=1,* ***id\_ptofile=1*** *[id\_user=1=1 id\_post =1,* ***id\_ptofile =1****,* ***quantity=1*** *[* | Видача профіля користувача |

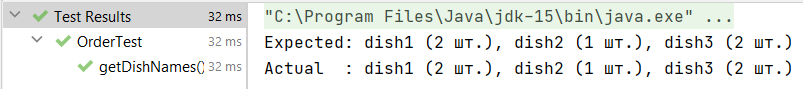


Рис. 2 – результат тесту 1

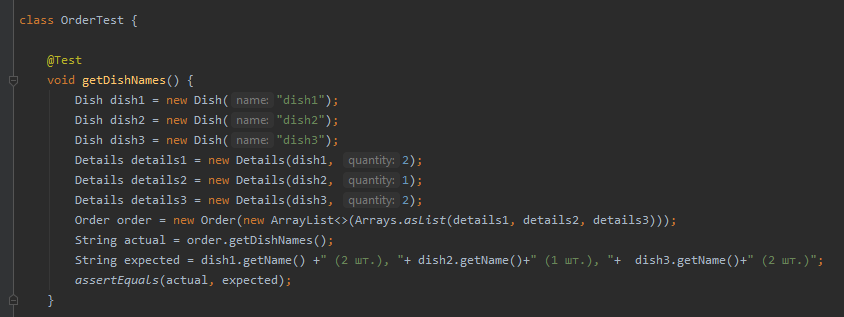


Рис. 1 – програмний код тесту 1

Специфікація функції Search класу Travel має наступний опис.

Вхідні параметри:

1. City – дані для пошуку сопутівника (Об’єкт класу Travel):

* містить посилання на об’єкт Користувач та також інші поля (власне id, id Користувача та id Міста, , посилання на об’єкт Користувача), які не задіяні в тестуванні або не впливають на очікуваний результат чи генеруються автоматично.

Значення не повертаються функцією. Для перевірки коректності роботи, викликається функція getTravelContact, що повертає Користувача, що додав такий самий запит для пошуку сопутівника

Таблиця 2 – Опис тестових наборів для функції Search

|  |  |  |
| --- | --- | --- |
| **№ тесту** | **Опис значень початкових даних** | **Опис очікуваних значень результату** |
| 2 | *Поточні дані –* Country= Ukraine  City= Kiev | User2, User5, User9 |

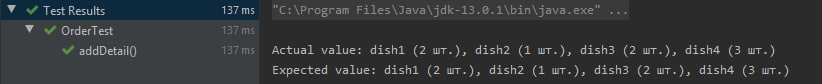
Специфікація функції Delete класу User має наступний опис.

Рис. 4 – результат тесту 2

Рис. 3 – програмний код тесту 2

Вхідні параметри:

1. id – ідентифікатор Користувача, що запитується (ціле число).

Значення, що повертаються функцією:

detail – об’єкт однойменного класу, що містить деталі Профилю;

null – якщо об’єкт з даним id не знайдено

|  |  |  |
| --- | --- | --- |
| **№ тесту** | **Опис значень початкових даних** | **Опис очікуваних значень результату** |
| 4 | *Поточні дані -*  *details = {id\_user=1, UserName=Shishkin, Email= LiliDog@ron.com, password:\*\*\*\*\*\* }*  Вхідні дані: id = 1 | Drop user1 |
| 5 | *Поточні дані -*  *details = {id\_user=2, UserName=Shishkin, Email= LiliDog@ron.com, password:\*\*\*\*\*\* }*  Вхідні дані: id = 1 | null |

Таблиця 3 – Опис тестових наборів для функції Delete

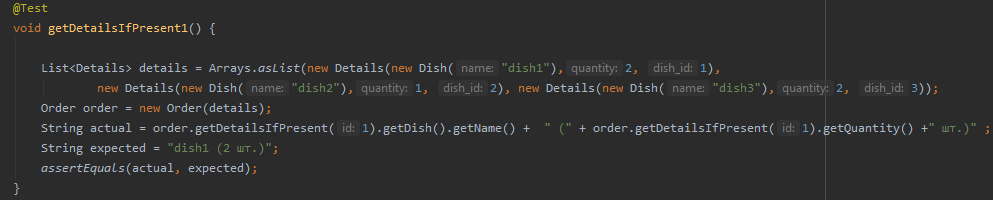


Рис. 5 – програмний код тесту 4

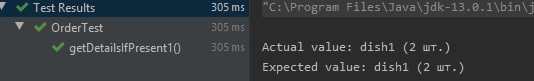
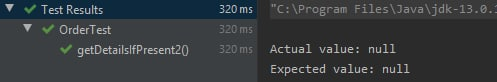


Рис. 8 – результат тесту 5

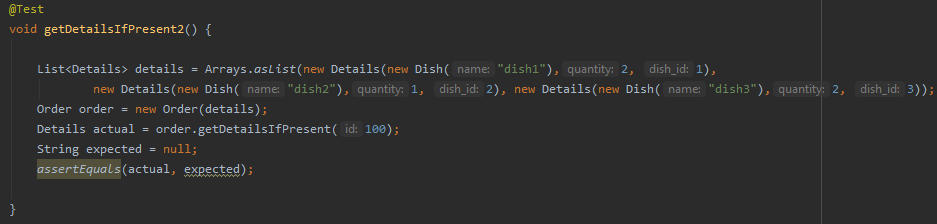


Рис. 7 – програмний код тесту 5

Рис. 6 – результат тесту 4