

# Отчёт по лабораторной работе

## «Написание автотестов»

**Выполнил:** Саков Евгений 4 курс

7 группа;

**Сайт:** <https://goldapple.ru>;

**Функция:** добавление товара в корзину  
(артикул, название);

**Инструменты:** язык программирования Python, библиотеки Selenium, Cypress или Playwright.

**Результат тестирования:** 7 успешно пройденных тестов из 7.

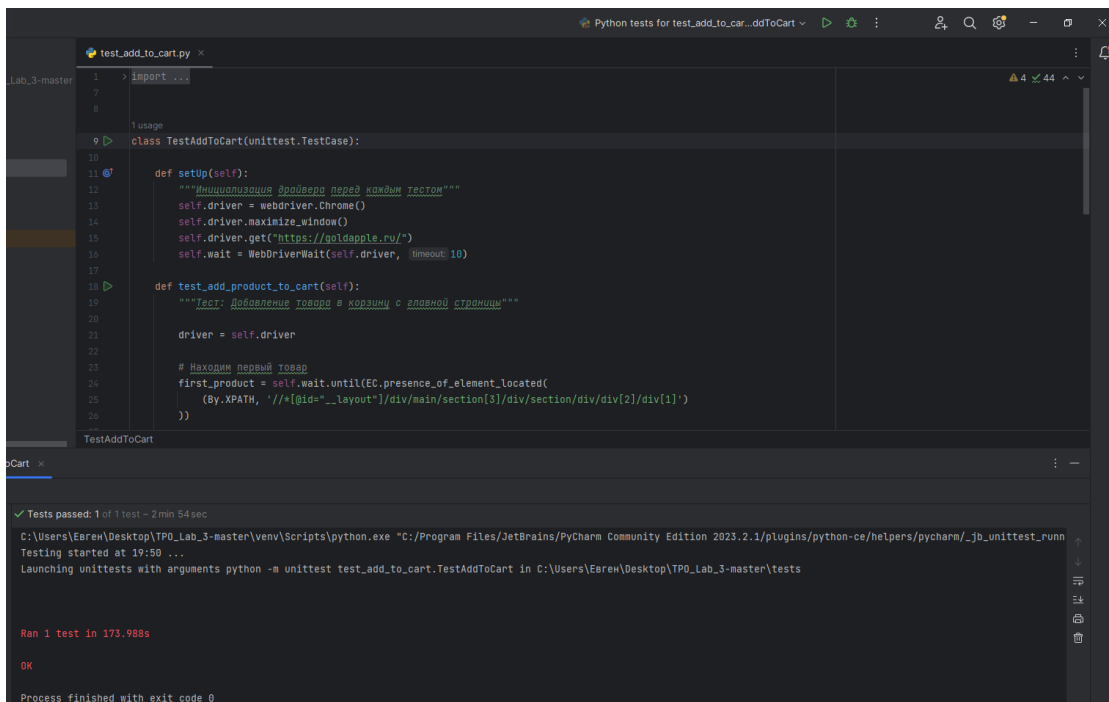
### Тест-кейс 1: Успешное добавление товара в корзину с главной страницы

**Цель:** Проверить, что товар успешно добавляется в корзину с главной страницы.

**Предусловия:** Пользователь находится на главной странице сайта.

1. Перейти на главную страницу сайта.
2. Выбрать любой товар на главной странице.
3. Нажать кнопку «Добавить в корзину».
4. Перейти в корзину.
5. Проверить, что товар присутствует в корзине с правильным названием, количеством и ценой.

**Ожидаемый результат:** Товар добавлен в корзину, отображается корректная информация.



The screenshot shows the PyCharm IDE interface. The top pane displays the code for a test file named `test_add_to_cart.py`. The code includes imports for `unittest` and `webdriver`, and defines a `TestAddToCart` class with two methods: `setUp` and `test_add_product_to_cart`. The `setUp` method initializes a Chrome WebDriver, maximizes the window, and navigates to `https://goldapple.ru/`. The `test_add_product_to_cart` method uses XPath to find a product element and adds it to the cart. The bottom pane shows the test execution results, indicating that the test passed successfully in 173.988 seconds.

```
test_add_to_cart.py
1  import unittest
2
3  from selenium import webdriver
4  from selenium.webdriver.common.by import By
5  from selenium.webdriver.support import expected_conditions as EC
6  from selenium.webdriver.support.ui import WebDriverWait
7
8  class TestAddToCart(unittest.TestCase):
9
10     def setUp(self):
11         """Инициализация драйвера перед каждым тестом"""
12         self.driver = webdriver.Chrome()
13         self.driver.maximize_window()
14         self.driver.get("https://goldapple.ru/")
15         self.wait = WebDriverWait(self.driver, timeout=10)
16
17     def test_add_product_to_cart(self):
18         """Тест: Добавление товара в корзину с главной страницы"""
19
20         driver = self.driver
21
22         # Находим первый товар
23         first_product = self.wait.until(EC.presence_of_element_located(
24             (By.XPATH, '//*[@id="__layout"]/div/main/section[3]/div/section/div/div[2]/div[1]')
25         ))
26
27 Tests passed: 1 of 1 test - 2 min 54 sec
C:\Users\Earew\Desktop\TP0_Lab_3-master\venv\Scripts\python.exe "C:/Program Files/JetBrains/PyCharm Community Edition 2023.2.1/plugins/python-ce/helpers/pycharm/_jb_unittest_runn
Testing started at 19:50 ...
Launching unittests with arguments python -m unittest test_add_to_cart.TestAddToCart in C:\Users\Earew\Desktop\TP0_Lab_3-master\tests
Ran 1 test in 173.988s
OK
Process finished with exit code 0
```

Сложность: `time.sleep(6)` -> на `time.sleep(8)`.

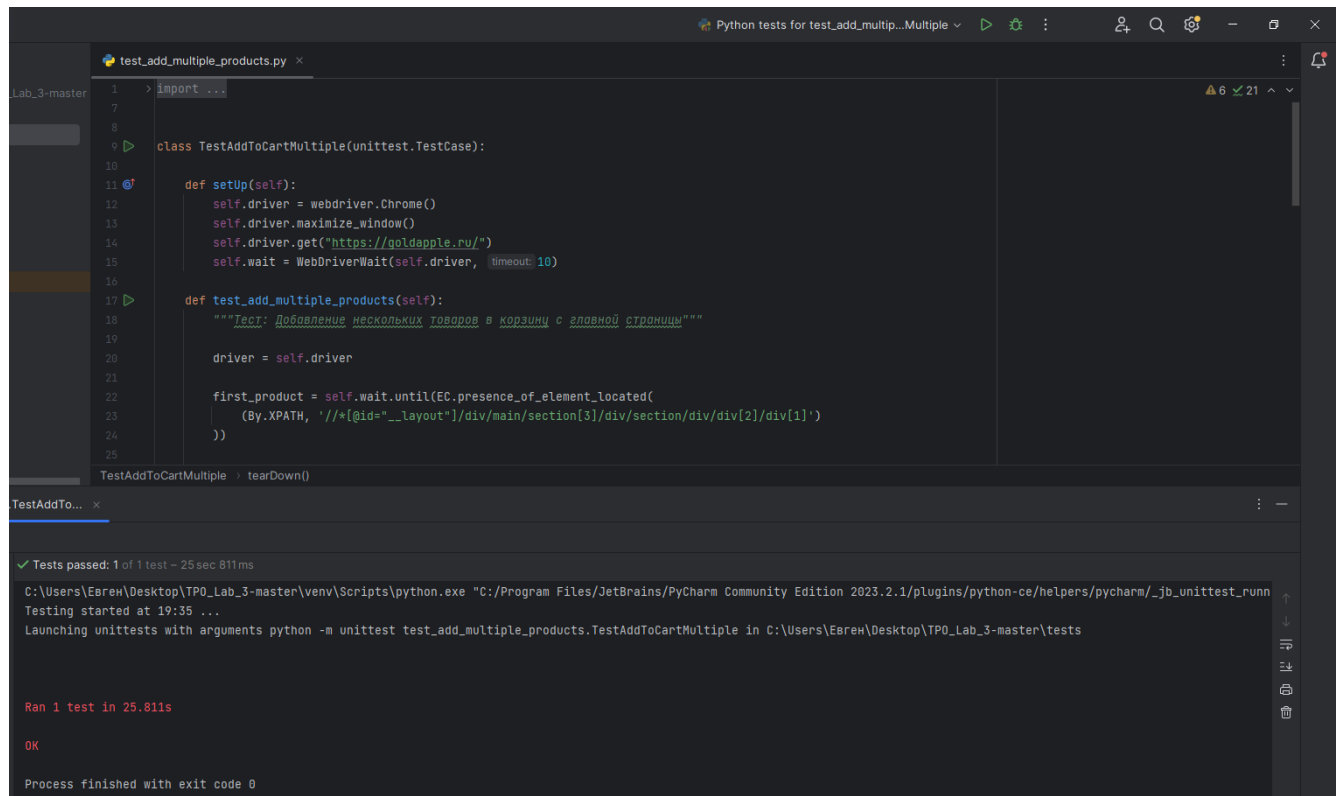
## Тест-кейс 2: Добавление нескольких товаров в корзину

**Цель:** Проверить, что несколько разных товаров можно добавить в корзину.

**Предусловия:** Пользователь находится на главной странице.

1. Добавить первый товар в корзину.
2. Вернуться на главную страницу и выбрать второй товар.
3. Добавить второй товар в корзину.
4. Перейти в корзину.

**Ожидаемый результат:** В корзине присутствуют оба товара с корректными названиями и количеством.



The screenshot shows the PyCharm IDE with a file named `test_add_multiple_products.py` open. The code defines a `TestAddToCartMultiple` class that inherits from `unittest.TestCase`. It includes a `setUp` method to initialize a WebDriver and a `test_add_multiple_products` method that performs a test. The test method uses `WebDriverWait` to wait for an element and then interacts with it using `By.XPATH`. Below the code editor, the test runner output is visible, showing that the test passed successfully in 25.811 seconds.

```
1 > import ...
2
3
4
5
6
7
8
9
10
11 class TestAddToCartMultiple(unittest.TestCase):
12
13     def setUp(self):
14         self.driver = webdriver.Chrome()
15         self.driver.maximize_window()
16         self.driver.get("https://goldapple.ru/")
17         self.wait = WebDriverWait(self.driver, timeout=10)
18
19     def test_add_multiple_products(self):
20         """Тест: Добавление нескольких товаров в корзину с главной страницы"""
21
22         driver = self.driver
23
24         first_product = self.wait.until(EC.presence_of_element_located(
25             (By.XPATH, '//*[@id="__layout"]/div/main/section[3]/div/section/div/div[2]/div[1]'))
26
27         TestAddToCartMultiple.tearDown()
```

TestAddTo... x

✓ Tests passed: 1 of 1 test – 25 sec 811ms

C:\Users\Eврен\Desktop\TP0\_Lab\_3-master\venv\Scripts\python.exe "C:/Program Files/JetBrains/PyCharm Community Edition 2023.2.1/plugins/python-ce/helpers/pycharm/\_jb\_unittest\_runn

Testing started at 19:35 ...

Launching unittests with arguments python -m unittest test\_add\_multiple\_products.TestAddToCartMultiple in C:\Users\Eврен\Desktop\TP0\_Lab\_3-master\tests

Ran 1 test in 25.811s

OK

Process finished with exit code 0

## Тест-кейс 3: Проверка обновления количества товара в корзине

**Цель:** Проверить, что пользователь может изменить количество товара в корзине.

**Предусловия:** В корзине находится хотя бы один товар.

1. Перейти в корзину.
2. Изменить количество товара с 1 на 2.
3. Проверить, что количество товара обновилось, и цена изменилась соответствующим образом.

**Ожидаемый результат:** Количество товара и цена обновились корректно.

```
1 > import ...
2
3
4
5
6
7
8
9
10 class TestChangeCount(unittest.TestCase):
11
12     def setUp(self):
13         self.driver = webdriver.Chrome()
14         self.driver.maximize_window()
15         self.driver.get("https://qoidapple.ru/")
16         self.wait = WebDriverWait(self.driver, timeout=10)
17
18     def test_add_multiple_same_products(self):
19         """Тест: Проверка обновления количества товара в корзине"""
20
21         driver = self.driver
22
23         first_product = self.wait.until(EC.presence_of_element_located(
24             (By.XPATH, '//*[@id="__layout"]/div/main/section[3]/div/section/div/div[2]/div[1]')
25         ))
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

TestChangeCount · test\_add\_multiple\_same\_products...

Products.Test... ·

✓ Tests passed: 1 of 1 test - 54sec 941ms

C:\Users\Egreh\Desktop\TP0\_Lab\_3-master\venv\Scripts\python.exe "C:/Program Files/JetBrains/PyCharm Community Edition 2023.2.1/plugins/python-ce/helpers/pycharm/\_jb\_unittest\_runn

Testing started at 19:37 ...

Launching unittests with arguments python -m unittest test\_add\_multiple\_same\_products.TestChangeCount in C:\Users\Egreh\Desktop\TP0\_Lab\_3-master\tests

Ran 1 test in 54.941s

OK

Process finished with exit code 0

## Тест-кейс 4: Проверка удаления товара из корзины

**Цель:** Проверить, что пользователь может удалить товар из корзины.

**Предусловия:** В корзине находится хотя бы один товар.

1. Перейти в корзину.
2. Нажать на иконку «Удалить» рядом с товаром.
3. Подтвердить удаление (если требуется).

**Ожидаемый результат:** Товар удаляется из корзины, корзина становится пустой или отображает оставшиеся товары.

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

TestDeleteFromCart · test\_remove\_from\_cart()

DeleteFro... ·

✓ Tests passed: 1 of 1 test - 37sec 706ms

C:\Users\Egreh\Desktop\TP0\_Lab\_3-master\venv\Scripts\python.exe "C:/Program Files/JetBrains/PyCharm Community Edition 2023.2.1/plugins/python-ce/helpers/pycharm/\_jb\_unittest\_runn

Testing started at 20:05 ...

Launching unittests with arguments python -m unittest test\_remove\_from\_cart.TestDeleteFromCart in C:\Users\Egreh\Desktop\TP0\_Lab\_3-master\tests

Ran 1 test in 37.707s

OK

Process finished with exit code 0

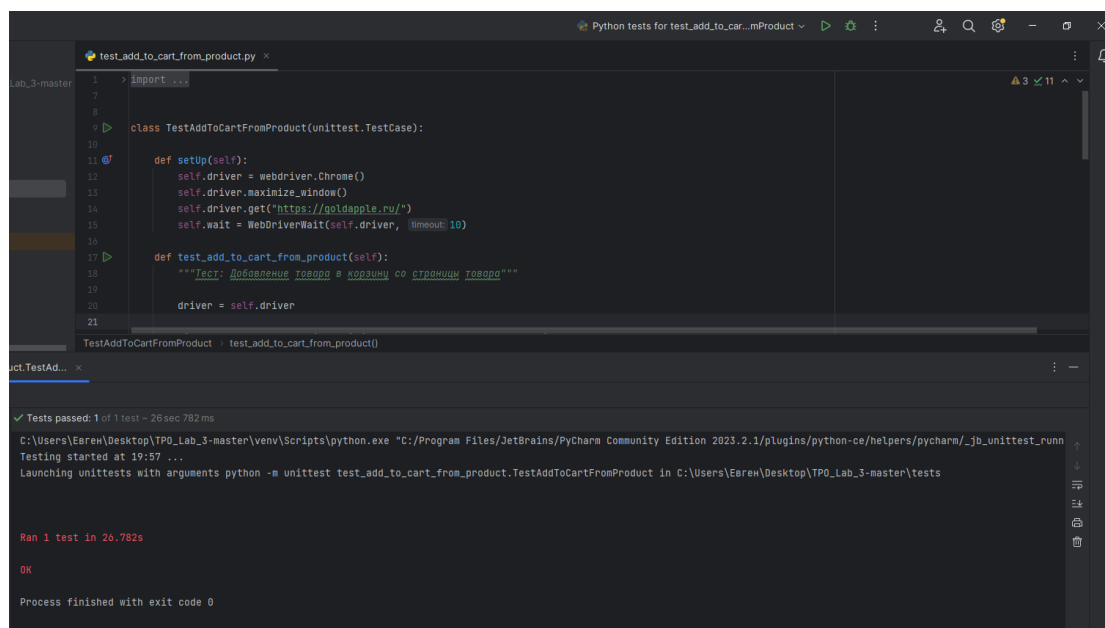
### Тест-кейс 5: Добавление товара в корзину из карточки товара

**Цель:** Проверить, что товар можно добавить в корзину из карточки товара.

**Предусловия:** Пользователь находится на странице карточки товара.

1. Перейти на страницу карточки товара.
2. Нажать кнопку «Добавить в корзину».
3. Перейти в корзину.

**Ожидаемый результат:** Товар успешно добавлен в корзину с корректными данными.



### Тест-кейс 6: Проверка сохранения товара в корзине после обновления страницы

**Цель:** Проверить, что товар успешно добавляется в корзину и при обновлении страницы не удален из нее.

**Предусловия:** В корзине находится хотя бы один товар.

1. Перейти на главную страницу сайта.
2. Выбрать любой товар на главной странице.
3. Нажать кнопку «Добавить в корзину».
4. Обновить страницу.
5. Перейти в корзину.
6. Проверить, что товар присутствует в корзине с правильным названием, количеством и ценой.

**Ожидаемый результат:** Товар добавлен в корзину, отображается корректная информация.

The screenshot shows the PyCharm IDE interface. The top editor pane displays a Python file named `test_cart_persistence.py` with the following code:

```
30 By.XPATH,
31     value: '//*[@id="__layout"]/div/main/section[3]/div/section/div/div[2]/div[1]/div/div/div/div/article/div/div/div/div[3]/button'
32 )
33 add_to_cart_button.click()
34
35 time.sleep(8)
36
37 cart_button = self.wait.until(EC.element_to_be_clickable(
38     (By.XPATH, '//*[@id="__layout"]/div/header/div[2]/div[2]/button[5]')
39 ))
40 cart_button.click()
41
42 driver.refresh()
43 time.sleep(8)
44
45 # Переходим в корзину
46 cart_button = self.wait.until(EC.element_to_be_clickable(
47     (By.XPATH, '//*[@id="__layout"]/div/header/div[2]/div[2]/button[5]')
48 ))
49 cart_button.click()
```

The bottom pane shows the test execution results for `TestCartPersistence`. It indicates that 1 test passed in 33.370 seconds. The command used to launch the test is:

```
python -m unittest test_cart_persistence.TestCartPersistence in C:\Users\Ebren\Desktop\TP0_Lab_3-master\tests
```

The process finished with exit code 0.

## Тест-кейс 7: Проверка добавления товара с отсутствующим на складе статусом

**Цель:** Проверить, что невозможно добавить товар с отсутствующим на складе статусом.

**Предусловия:** Пользователь находится на странице товара, который отсутствует в наличии.

1. Перейти на страницу товара с пометкой «Нет в наличии».
2. Попробовать добавить товар в корзину.

**Ожидаемый результат:** Товар не добавляется в корзину, отображается сообщение «Нет в наличии» или кнопка «Добавить в корзину» недоступна.

```
Python tests for test_add_out_of...mProduct
test_add_out_of_stock.py
1 > import ...
2
3
4
5
6
7
8
9
10
11 class TestAddEndedToCartFromProduct(unittest.TestCase):
12
13
14 def setUp(self):
15     self.driver = webdriver.Chrome()
16     self.driver.maximize_window()
17     self.driver.get("https://goldapple.ru/27582600015-tip-top-svetlo-rozovyj")
18     self.wait = WebDriverWait(self.driver, timeout=15)
19
20
21 def test_add_out_of_stock(self):
22     """Тест: Добавление товара, которого нет в наличии в корзину со страницы товара"""
23
24     driver = self.driver
25
26     # Ищем кнопку "Добавить в корзину"
27     add_to_cart_button = self.wait.until(EC.presence_of_element_located((
28         By.XPATH, '//*[@id="__layout"]/div/main/article/div[1]/div[1]/form/div[4]/div/button'
29     )))
30
31     TestAddEndedToCartFromProduct.setUp()
32
AddEnded...
Tests passed: 1 of 1 test - 7 sec 650 ms
C:\Users\Earen\Desktop\TP0_Lab_3-master\venv\Scripts\python.exe "C:\Program Files\JetBrains\PyCharm Community Edition 2023.2.1\plugins\python-ce\helpers\pycharm\_jb_unittest_runn
Testing started at 19:42 ...
Launching unittests with arguments python -m unittest test_add_out_of_stock.TestAddEndedToCartFromProduct in C:\Users\Earen\Desktop\TP0_Lab_3-master\tests
Ran 1 test in 7.650s
OK
Process finished with exit code 0
```

Изначально был выбран другой товар, но потом он появился в наличии и тест показывал такой результат:

```
Python tests for test_add_out_of...mProduct
test_add_out_of_stock.py
1 > import ...
2
3
4
5
6
7
8
9
10
11 class TestAddEndedToCartFromProduct(unittest.TestCase):
12
13
14 def setUp(self):
15     self.driver = webdriver.Chrome()
16     self.driver.maximize_window()
17     self.driver.get("https://goldapple.ru/19000105305-eros-man-eau-de-parfume")
18     self.wait = WebDriverWait(self.driver, timeout=15)
19
20
21 def test_add_out_of_stock(self):
22     """Тест: Добавление товара, которого нет в наличии в корзину со страницы товара"""
23
24     driver = self.driver
25
26     # Ищем кнопку "Добавить в корзину"
27     add_to_cart_button = self.wait.until(EC.presence_of_element_located((
28         By.XPATH, '//*[@id="__layout"]/div/main/article/div[1]/div[1]/form/div[4]/div/button'
29     )))
30
31     TestAddEndedToCartFromProduct
32
AddEnded...
Tests failed: 1 of 1 test - 7 sec 865 ms
Ran 1 test in 7.865s
FAILED (failures=1)
Тест провален: Товар отсутствует в корзине.
УЗНАТЬ О ПОСТУПЛЕНИИ != ДОБАВИТЬ В КОРЗИНУ
Expected :ДОБАВИТЬ В КОРЗИНУ
Actual   :УЗНАТЬ О ПОСТУПЛЕНИИ
<Click to see difference>
```

Также в ходе тестирования были трудности со временем исполнения кода и временем отклика сайта, поэтому были добавлены в некоторых тестах:

```
# Ожидаем добавления, иначе иногда падает  
time.sleep(6)
```