**Test Function Document**

|  |  |
| --- | --- |
| Project Name: Assignment 1 Testing | |
| Automation Title: test\_load\_products\_from\_csv | Version: 1.0 |
| Testing Phase: Phase 1 | Date of Test: 16 November 2023 |
| Module Name: checkout\_and\_payment.py | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Function Title: load\_products\_from\_csv | | | | Test Designed by: Edvin Eriksson | | | |
| Test Priority (Low/Medium/High): Low | | | | Test Designed Date: 16 November 2023 | | | |
| Description: The objective of the load\_products\_from\_csv function is to append products to a list, given products in a csv file with products containing name, price and units. The first row is identified as the header. | | | | Test Executed by: Edvin Eriksson | | | |
| Test Execution date: 16 November 2023 | | | |
|  | | | | | | | |
| Pre-conditions: None | | | | | | | |
| Dependencies: None | | | | | | | |
|  | | | | | | | |
| S. No | Equivalence Class | Test case data | Expected Results | | Actual Results | Status (Pass/Fail) | Notes |
| 1 | IEC1 | 1 | Function raises TypeError | | Function expected [] | Fail | Test name: test\_int\_input()  Int input |
| 2 | IEC2 | 0.5 | Function raises TypeError | | Function raises TypeError | Pass | Test name: test\_float\_input()  Float input. |
| 3 | IEC3 | [“copy\_products.csv”] | Function raises TypeError | | Function raises TypeError | Pass | Test name: test\_list\_input()  List input. Uses a fixture to create a temporary copy of products.csv. |
| 4 | VEC1 | “non\_existing.csv” | Function raises FileNotFoundError | | Function raises FileNotFoundError | Pass | Test name: test\_EC1()  Testing a csv file that does not exist as an input. |
| 5 | VEC2 | “empty\_products.csv” | len(empty\_products) == 0  empty\_products == [] | | len(empty\_products) == 0  empty\_products == [] | Pass | Test name: test\_EC2()  Testing an empty csv file as input. Uses a fixture to create an empty csv file. |
| 6 | VEC3 | “” | Function raises FileNotFoundError | | Function raises FileNotFoundError | Pass | Test name: test\_EC3  Testing an empty string as input |
| 7 | VEC4 | “copy\_products.csv” | products[0].name == 'Apple'  products[0].price == 2.0  products[0].units == 10  products[1].name == 'Banana'  products[1].price == 1.0  products[1].units == 15  len(products) == 71 | | products[0].name == 'Apple'  products[0].price == 2.0  products[0].units == 10  products[1].name == 'Banana'  products[1].price == 1.0  products[1].units == 15  len(products) == 71 | Pass | Test name: test\_EC4()  Uses a fixture to create a temporary copy of products.csv.  Tests a csv file containing 3 columns (Product, Price and Units) |
| 8 | VEC5 | “modify\_products.csv” | modified\_products[71].price == -3.0  modified\_products[72].price == 0 | | modified\_products[71].price == -3.0 modified\_products[72].price == 0 | Pass | Test name: test\_EC5()  Uses a fixture to create a temporary copy of products.csv that is modified for the corresponding test.  Tests a csv file containing 3 columns (Product, Price and Units) but some products have negative or no price. |
| 9 | VEC6 | “modify\_products.csv” | modified\_products[73].units == -10  modified\_products[74].units == 0 | | modified\_products[73].units == -10  modified\_products[74].units == 0 | Pass | Test name: test\_EC6()  Uses a fixture to create a temporary copy of products.csv that is modified for the corresponding test.  Tests a csv file containing 3 columns (Product, Price and Units) but some products have negative or no units. |
| 10 | VEC7 | “modify\_products.csv” | modified\_products[75].name == '1' modified\_products[76].name == '0.5' | | modified\_products[75].name == '1' modified\_products[76].name == '0.5' | Pass | Test name: test\_EC7()  Uses a fixture to create a temporary copy of products.csv that is modified for the corresponding test.  Tests a csv file containing 3 columns (Product, Price and Units) but some products have integer or floats as name |
| 11 | VEC8 | “modify\_products.csv” | modified\_products[77].name == 'Cheese'  modified\_products[78].name == 'Popcorn' | | modified\_products[77].name == 'Cheese'  modified\_products[78].name == 'Popcorn' | Pass | Test name: test\_EC8()  Uses a fixture to create a temporary copy of products.csv that is modified for the corresponding test.  Tests a csv file containing an empty row between products |
| 12 | VEC9 | “modify\_products.csv” | Function raises KeyError | | Function raises KeyError | Pass | Test name: test\_EC9()  Uses a fixture to create a temporary copy of products.csv that is modified for the corresponding test.  Tests a csv file containing only 2 columns. |
| 13 | VEC10 | “modify\_products.csv” | not any('Dairy' in str(product) for product in modified\_products) | | not any('Dairy' in str(product) for product in modified\_products) | Pass | Test name: test\_EC10()  Uses a fixture to create a temporary copy of products.csv that is modified for the corresponding test.  Tests a csv file containing 4 columns. |
| 14 | VEC11 | “modify\_products.csv” | modified\_products[0].name == 'Bread '  modified\_products[0].price == 3.0  modified\_products[0].units == 6 | | modified\_products[0].name == 'Bread '  modified\_products[0].price == 3.0  modified\_products[0].units == 6 | Pass | Test name: test\_EC11()  Uses a fixture to create a temporary copy of products.csv that is modified for the corresponding test. |