**Shashank Mondrati Task V**

**TDD Phase 1**:

Calculating Total Impure Price:

It is calculated by adding the multiplication of the amount and unit price of each product entered by the user.

Ex. 3 apples each 5$ should equal $15. This is Impure price.

When the test is run it is in the red phase, when the method is implemented the test is in green phase, the main answer lies in implementing the method and setting it to 75.

**TDD Phase 2**:

Calculating Total Discount:

It is calculated by adding multiplication of total price and discount of each product.

Ex: unit price of a product times quantity times user entered discount divided by 100..should equal total discount.

When the fixtures and asserts method are implemented in the test method, the result is in red phase, but when the method is implemented the test passes the red phase and enters the green cycle.

**TDD Phase 3**:

Calculating Total Pure Price:

This method is hard to implement, and hard to understand how the impure, pure prices works. An easy way to calculate total pure price would be subtraction between total Impure price and total discount, that way we can find the total pure price, think of it as gross income - taxes = net income.

Total Pure Price method implementation: impure price - total discount

Impure Price: total\_impure\_price += unit\_price \* quantity

Total Discount: : total\_discount += unit\_price \* quantity\* (discount / 100)

Total Pure Price would be a subtraction between those two logics above.

**TDD Phase 4**:

Calculating Tax:

There is no correct tax number to use, so I used calculating tax at 7.5%, in python numbering format that would be 0.075, one of the east ways to calculate it is tax percent times total pure price, since it already has subtraction so it should be easy to calculate tax that way.

Ex: Total Tax: total pure price \* 0.075 (7.5%).

Testing it is easy, adding test fixtures, and equaling to the answer we get in red cycle should make the test pass the red cycle.