A.) FR1. Test 1: check to make sure a negative number such as -1 is an invalid input

Test 2: check to make sure there is no discount for any input between 1 and 5, such as 3

Test 3: check to make sure there is a 5% discount for any input between 6 and 9

Test 4: check to make sure there is a 10% discount for any input of 10 or greater

No discount

Items > 5

Final price after discount

Discount = 5%

Discount = 10%

Items >= 10

Item <= 50

Start

-----------------------------------------------------------------------------------------------------------------------

FR2 Test 1: If input > 51 🡪 invalid

Test 2: if input <= 50 🡪 valid

start

valid

Input > 51

------------------------------------------------------------------------------------------------------------------------

FR3 Test 1: check to make sure that if a customer is a member comes out as TRUE, they receive an additional 10% off before taxes. So, if their total comes out to $100 before taxes, that number should change to $90

FR3 Test 2: If that Boolean comes out as FALSE, they do no t get an additional 10% off. So, if their total is $100 and the Boolean is false, the total remains $100

Deduct 10% off the price before taxes

true

Member or not

Start

Price remains the same

Final price

false

-----------------------------------------------------------------------------------------------------------------------------------

FR4 Test 1: if tax-exempt 🡪 false 🡪 add a 4.5% sales tax to the total cost after all the discounts

Test 2: if tax-exempt 🡪 true 🡪 return the total after all the discounts take place

Start

price remains the same

true

Tax - exempt

Final price

price increases by 4.5%

false

B.) Partition Tests:

FR1 Test 2: check to make sure there is no discount for an input of 3

FR1 Test 3: check to make sure there is a 5% discount for an input of 7

FR1 Test 4: check to make sure there is a 10% discount for an input of 15

FR2 Test 1: check to make sure 30 is a valid input

FR2 Test 2: check to make sure 55 is an invalid input

Boundary Tests:

FR1 Test 1: check to make sure that -1 is an invalid input

FR1 Test 2: check to make sure there is no discount for any input of 0 and 4

FR1 Test 3: check to make sure there is a 5% discount for any input of 6 and 9

FR1 Test 4: check to make sure there is a 10% discount for an input of 10

FR2 Test 1: check to make sure 50 is a valid input

FR2 Test 2: check to make sure 51 is an invalid input

C.) Partition Tests:

Test 1: the total should be rounded to two decimal places

Boundary Tests:

Test1: the total should be rounded to two decimal places

Test 2: if the total ends up having a number to the thousandths place, if that number is >= 5, then the total will be rounded up

Test 3: if the total ends up having a number to the thousandths place, if that number is < 5, then the first two digits in the first two decimal places will remain the same