**HW02 Conditions and Software Development Life Cycle**

Due Date: **See Canvas**

Purpose: Learn how to design, implement and test a quality maintainable solution following the software development life cycle. The focus is on using selection statements.

Effort: **Individual**: Read [CS Academic Integrity .pdf](https://drive.google.com/file/d/1vHXGQPPRUXZQrUxfnM6hJQid9jio7xlt/view?usp=sharing)

Points: **100 (see rubric in canvas)**

Deliverables: **Upload this document with your answers and your .java file as separate files. Do not upload as a zip file.**

**Assignment Description**

Analyze the requirements and break into smaller tasks, create test data for each task, and design the algorithms for each task for a "very" simple Taco Shop program that allows an employee to take an order and produce a receipt. The program reports the number and type of tacos the customer ordered, cost for specified number of tacos, taxes, and total cost. The program must handle ***invalid user input*** as described.

**Requirements**

* The Taco Shop can process only 1 customer and 1 order.
* All tacos in the order will have the same filling (beef, chicken, etc.)
* The Taco Shop offers the 5 types of tacos and price per taco as listed in table below

| **Option** | **Taco Filling** | **Price** |
| --- | --- | --- |
| 1 | Veggie | $3.49 |
| 2 | Chicken | $3.99 |
| 3 | Carnitas | $4.19 |
| 4 | Beef | $4.59 |
| 5 | Shrimp | $5.19 |

* The customer can order 1 to 12 tacos
* On all orders with 5 or more tacos, an 8% discount is applied to the total cost before taxes
* There is a 6.5% charge for taxes

**Acceptance Criteria**

* Displays a **menu** with the different fillings for the tacos
* Prompts user for **filling for tacos** and ensures a valid menu option is entered
  + If number is not valid option (1 to 5) the **program displays error message and ends**
* Prompts user for **number of tacos** and ensures a valid number of tacos is entered (only if valid option was selected for filling)
  + If the number of tacos is not valid (1 to 10) the **program displays and error message and ends**
* Calculate cost and display receipt if valid values were entered for filling and number of tacos (Only if valid number of tacos was entered)
  + **Number** of tacos order and what **filling** they have
  + **Cost** for the number of tacos ordered
  + **Taxes**
  + **Total cost**

**Specification**

Should implement the following:

* Create a Java class called **LastNameFirstNameHW02** within the **CS1150HW** project
* Using proper indentation, follow naming conventions, commenting code, etc.
* Use **correct data types** and **constants** where possible
  + Use constants for numeric values that will not change while the code runs.
  + Use String variable to store filling name: String fillingName = "";
* Format output for money amounts to two decimal places using **System.out.printf**
* Avoid repeating code
  + You may find yourself repeating code in this assignment, and repeating code is never a good thing.
  + If you find yourself in this position, work to pull out the common code and write that code in one place.
* Be sure you understand the difference between nested and multiway before writing the code.
  + Use a nested if and multiway if but do not use a switch statement for this assignment.
  + Use **NESTED IF** statements to make program termination occur properly.
  + **To simplify if-statements use logical operators**
  + **DO NOT** use System.exit(0) to exit program if an error occurs
  + **DO NOT** use break or return statements if an error occurs
  + The purpose is to learn to write properly nested if-statements.
  + Using System.exit, return, break will result in loss of points for correctness.
* **Do not implement concepts not covered in class yet.**

**Output**

Your output should look like the following:

**Output - Example 1 – Valid Filling, Valid Number of Tacos**

Hungry for some Tacos?!

Option Filling Price

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

1 Veggie $3.49

2 Chicken $3.99

3 Beef $4.59

4 Carnitas $4.19

5 Shrimp $5.19

Select a filling (1, 2, 3, 4, or 5): 1

Select number of tacos (from 1 to 12)? 3

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

3 Veggie Tacos $10.47

Taxes $ 0.68

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

Total Cost $11.15

**Output - Example 2 – Valid Filling, Valid Number of Tacos, Discount Applied**

Hungry for some Tacos?!

Option Filling Price

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

1 Veggie $3.49

2 Chicken $3.99

3 Beef $4.59

4 Carnitas $4.19

5 Shrimp $5.19

Select a filling (1, 2, 3, 4, or 5): 3

Select number of tacos (from 1 to 12)? 5

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

5 Carnitas Tacos $20.95

Discount -$ 1.68

Taxes $ 1.25

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

Total Cost $20.53

**Output - Example 3 – Invalid Filling Selection**

Hungry for some Tacos?!

Option Filling Price

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

1 Veggie $3.49

2 Chicken $3.99

3 Beef $4.59

4 Carnitas $4.19

5 Shrimp $5.19

Select a filling (1, 2, 3, 4, or 5): 0

0 is not a valid menu option. Please run program again, adios!

**Output - Example 4 – Invalid Number of Tacos Entry**

Hungry for some Tacos?!

Option Filling Price

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

1 Veggie $3.49

2 Chicken $3.99

3 Beef $4.59

4 Carnitas $4.19

5 Shrimp $5.19

Select a filling (1, 2, 3, 4, or 5): 2

Select number of tacos (from 1 to 12) 13

We cannot make 13 tacos. Please run program again, adios!

**Part 1: Requirements Analysis - Test Cases**

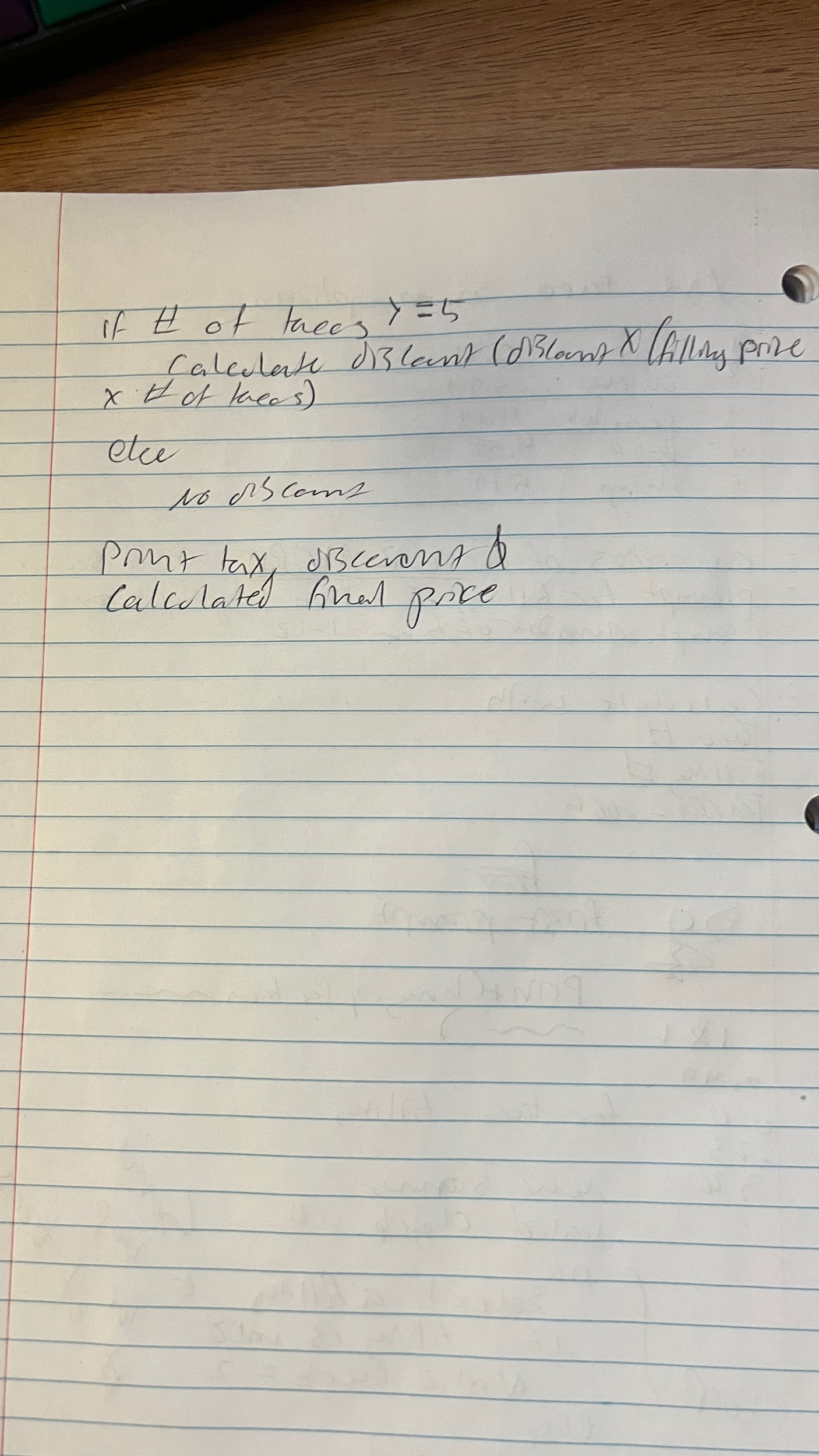
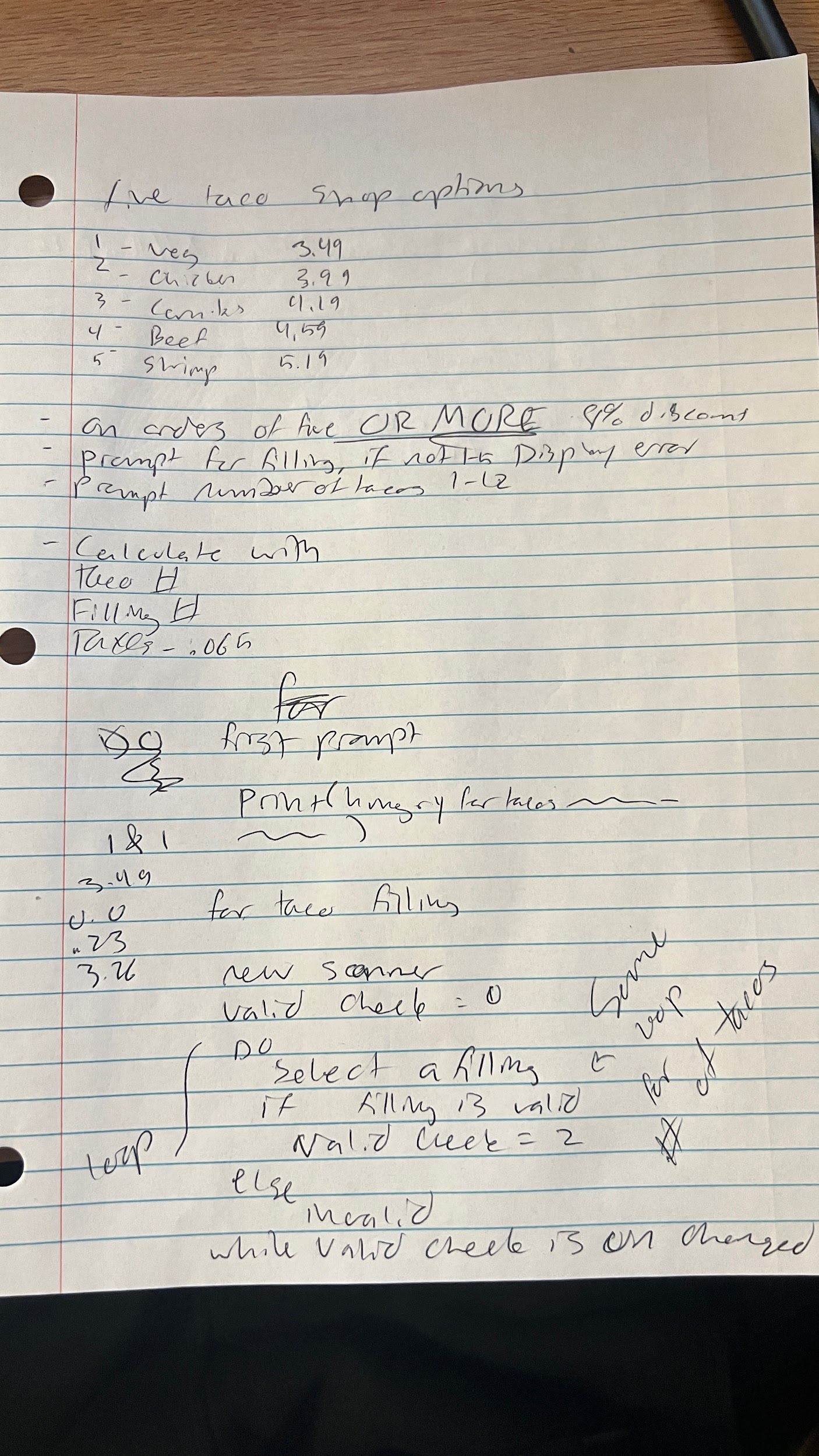
Create test cases for the program. Fill in the missing test cases.

| **Input** | **Output** |
| --- | --- |
| Filling option: 0 | 8 is not a valid filling option, unable to process your online order. |
| Filling option: 2  Number of Tacos: 13 | Invalid input, please select a valid quantity  How many tacos would you like (from 1 to 12:) |
| Filling option: 3  Number of Tacos: 5 | 5 Carnitas Tacos $20.95  Discount -$ 1.68  Taxes $ 1.25  Total Cost $20.53 |
| Filling option:2  Number of Tacos: 11 | Select a filling (1, 2, 3, 4, or 5):  2  How many tacos would you like (from 1 to 12:)  11  -------------------------------  11 Chicken tacos $43.89  Discount: -$3.51  Taxes: $2.85  -------------------------------  Total Cost: $44.55 |
| Filling option: 4  Number of Tacos: 7 | Select a filling (1, 2, 3, 4, or 5):  4  How many tacos would you like (from 1 to 12:)  7  -------------------------------  7 Beef tacos $32.13  Discount: -$2.57  Taxes: $2.09  -------------------------------  Total Cost: $32.61 |

**Part 2: Design Possible Solution as Pseudocode**

* Hand written - paper, whiteboard or tablet.
* Complete before you write your code.
* Be sure your pseudocode is an English version of your code, not a copy of your Java code. No credit if it is typed or is written as java code. Review [Design Algorithm: Pseudocode](https://docs.google.com/presentation/d/e/2PACX-1vSC4tM0BsOaYjRmd_emNeHtKrWaExKXfvvLxQC10rCt77CPT_WvF_s3W--o_yuj8XczYGo7qHxgjMIq/pub?start=false&loop=false&delayms=3000&slide=id.g10cd20069f2_0_5)

Insert image of your pseudocode for this problem below.



**Part 3: Resources**

* List at least 4 resources of concepts used in this assignment. T

| Concept | Resource Location - can be any of the following   * Lecture Number and slide * Section from the book * Resource from [Javatpoint](https://www.javatpoint.com) |
| --- | --- |
| Java Booleans | w3schools.com |
| LO CH5 Loops slide | slide 4 |
| LO CH5 Loops slide | slide 8 |
| LO CH5 Loops slide | slide 9 |
|  |  |

**Part 4: Implement Code**

Review [Coding Mindset and Quality Coding](https://docs.google.com/presentation/d/e/2PACX-1vSC4tM0BsOaYjRmd_emNeHtKrWaExKXfvvLxQC10rCt77CPT_WvF_s3W--o_yuj8XczYGo7qHxgjMIq/pub?start=false&loop=false&delayms=3000&slide=id.g10cd20069f2_0_24)

**Tip: Write code incrementally.**

* First, get the if statement working for the taco filling menu option
* Second, add the nested if statement for the number of tacos
* Third, add the code to process the costs

**Part 5: Learnings and Reflection**

* Explanations provide elaboration of the concepts using supporting evidence and vocabulary.
* Thoroughly reflects making clear the connection(s) with the experience and assignment.

| 1. What is the difference between using a variable and a constant?  A variable is something that represents a value that can be called on and changed. A constant is hard coded and cannot be called upon.  2. List the constants you used in the program.  **double** veg\_taco\_price = 3.49;  **double** chicken\_taco\_price = 3.99;  **double** carnitas\_taco\_price = 4.19;  **double** beef\_taco\_price = 4.59;  **double** shrimp\_taco\_price = 5.19;  **double** tax\_rate = .065;  **double** discount = .08;  For example **final** **double** BEEF\_TACO\_PRICE = 4.59;  3. Give an example of a variable in your program that uses an integer data type and an example using a double data type.  **int** min\_filling = 1;  **double** final\_order\_final\_price = 0;  4. Explain what declaring a variable or constant means in terms of memory. |
| --- |
|  |
| 6. What are naming conventions?  Give an example for each type of naming convention you used in your program when creating identifiers for class name, variable and constant  naming conventions is the way you name your variables. I used final\_ when a variable is not being calculated anymore and is ready to be printed. |
| 7. What are comments? Why are comments important?  comments are unexpressed lines of code ignored by the compiler that you can use to document and explain lines or sections of code. |
| 8. Explain when you should use a nested if and when you should use a multi-way if.  nested way if statements allow you to test multiple things, multi way if statements only check fro one criteria. |
| 9. Copy and paste an example from your code where you use a logical operator and explain the code.(min\_taco <= user\_taco\_num && user\_taco\_num <= max\_tacos)  checks to see if user taco number is greater than or equal to 1 and less than or equal to 12 |
| 10. Explain when you can use a switch statement instead of if/else. What part of this solution could have been implemented using a switch statement?did not use a switch statement, but I could have when testing for which number corresponds to which filling name. |
| 11. How did you test to make sure your program works? Ran it a whole bunch |
| 12. Why do you think a process like the software development life cycle is followed when creating a software program? to ensure that the software is scaleable and doesnt only work for one use case. |