1. Allow the user to enter a quantity and price, use ctl+z to stop. Use a function to compute the total (quantity times price). The function should be passed the quantity and price and then return the total. In the function, provide a 10% discount if the total is over $10,0000.00. Display quantity, price and total. Sum and display the extended price.

Input	Process	Output
	CompExtPrice(qty, unitprice) Extprice = qty*unitprice If extprice > 10000 Discamt = extprice * 0.10 Else Discamt = 0 newExtPrice = extPrice – discamt return newExtPrice	
Qty		Extprice
price	Main totalExtPrice = 0 Do you want to do this program (Yes or No) While (Yes) Input qty, price Extprice = CompExtPrice(qty,price) Display qty, price, Extprice totalExtPrice = totalExtPrice + extprice Do you want to continue with this program?	
	Display totalExtPrice	totalExtPrice

2. Enter players last name, number of hits and at bats at the keyboard, use ctl+z to stop. Use a function to compute batting average. Pass the hits and at bats to the function. The function should return batting average. Display last name and batting average. Give a count of the number of players entered.

input	process	ouput
player last name	Initialize player count to 0	player last name
number of hits	Use a loop to continuously accept input until `Ctrl+Z` is	Batting average

	pressed	
number of bats	Convert number of hits and at bats to integers	total number of players entered
player last name	Compute batting average using a function	
	Increment player count for each player	
	Display player's last name and batting average	
	Catch `EOFError` to end input loop	
	Display total number of players entered	

3. Enter the destination city, miles travelled and gallons used for a trip, use ctl+z to stop. Use a function to compute miles per gallon. Pass miles travelled and gallons used to the function. The function should return miles per gallon. Count the number of entries made (number of trips) Display destination city, miles and mpg. At end display the number of entries made.

input	output	process
Destination city	initialize trip count to 0	
Miles traveled	Use a loop to continuously accept input until `Ctrl+Z` is pressed	
gallons used	Convert miles traveled and gallons used to integers	
	Compute miles per gallon using a function	
	Increment trip count for each trip	

4. Allow the employee to enter last name, job code and hours worked, use ctl+z to stop. Use a function to determine the pay rate. Pass to this function the job code and it should return rate of pay. Use Job code L is $25/hr, A is $30/hr and J is $50/hr for respective pay rates. Compute

gross pay. Give time and a half for overtime. Display last name and gross pay. Sum and display total of all gross pay.

Input	process	output
Last name (string)	Prompt user to enter last name, job code, and hours worked.	Last name and gross pay for each employee.
Job code (character)	Use a function to determine the pay rate based on job code.	Total gross pay for all employees.
Hours worked (integer)	Compute gross pay: ● Regular hours (up to 40 hours) at regular rate. ● Overtime hours (above 40 hours) at time and a half rate.	
	Display last name and gross pay.	
	Accumulate total gross pay for all employees.	
	Stop when the user inputs Ctrl+Z (EOF).	

5. Allow the user to enter student last name, credit hours and district code, use ctl+z to stop. Use a function to compute tuition owed. Charge In district (code of I) $250 per credit hour. Out of district (code of O) is $550 per credit hour. The function should receive credit hours and district code and return tuition owed. Display student name and tuition owed. Sum and display total of all tuition owed.

Examples

1. Enter the number of Points and redemption code. For redemption code C then compute value as 2 x rewards points. Redemption code X then they get 3 x rewards points. All other codes get 1.5 x rewards points. Write a function that receives points and redemption code and computes rewards points. Display points, redemption code and rewards points.

2. Enter two numbers and operation code (A, S, M, D). Write a function that receives the two numbers and uses the operation code to perform an operation on the two numbers (A=addition, S=Subtraction, M=Multiplication, D=Division). Check for dividing by 0. If the second number is 0 then set result to -999. Display two number, operation code, result and message if attempt to divide by zero.
3. Allow the user to enter a string. The string can be entered with any case (all upper, all lower of mixed). Write a function that accepts the string and returns all lower case when the original string is all upper or mixed. If the original string is all lower then make the string all upper case. The function should return the new string. Display both the original and new string.

Input	process	output