Problem Set – Introduction to Functions.

1. Allow the user to repeatedly enter a quantity and price. Prompt the user on whether they want to do the program (Yes or No). 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: User inputs quantity, price, and user decision to continue (Yes or No)**

**Process: Call a function with quantity and price, calculate total = quantity × price, if total > 10,000.00, apply 10% discount, Keep running sum of all extended prices**

**Output: Display quantity, price, and total, display total of all extended prices**

1. Enter players last name, number of hits and at bats at the keyboard. Prompt the user on whether they want to do the program (Yes or No). Use a function to compute batting average. Pass the hits and at bats to the function. The function should return to the batting average. Display last name and batting average. Give a count of the number of players entered.

**Input: User inputs last name of player, number of hits, number of at-bats and decision to continue**

**Process: Call a function with hits and at-bats, compute batting average = hits / at-bats and count number of players entered**

**Output: Display last name and batting average and display total number of players entered**

1. Enter the destination city, miles travelled and gallons used for a trip. Prompt the user on whether they want to do the program (Yes or No). Use a function to compute miles per gallon and cost of gas. Pass miles travelled and gallons used to the function. The function should return miles per gallon anf compute gas cost to be gallons times 3.00. Count the number of entries made (number of trips) Display destination city, miles, mpg and gas cost. At end display the number of entries made, total miles travelled for all trips and total gas cost of all trips.

**Input: Destination city, miles travelled, gallons used and decision to continue**

**Process: Call a function with miles and gallons, compute MPG = miles / gallons, compute gas cost = gallons × $3.00, count number of trips and accumulate total miles and total gas cost**

**Output: Display city, miles, MPG, and gas cost and display number of entries, total miles, and total gas cost**

1. Allow the employee to enter last name, job code and hours worked. Prompt the user on whether they want to do the program (Yes or No). Use a function to determine the pay rate. Pass to this function the job code and it should return rate of pay and gross 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,hours, pay rate and gross pay. Sum and display total of all gross pay.

**Input: Last name, job code, hours worked and decision to continue**

**Process: Call a function with job code, determine pay rate from job code, compute gross pay overtime and accumulate total gross pay**

**Output: Display last name, hours, pay rate, and gross pay. Next display will total gross pay for all employees**

1. Allow the user to enter student last name, credit hours and district code. Prompt the user on whether they want to do the program (Yes or No). 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.

**Input: Student last name, credit hours, district code followed by the user’s decision to continue**

**Process: Call a function with credit hours and district code, determine tuition rate, compute tuition owed and accumulate total tuition**

**Output: Display student last name and tuition owed and display total tuition owed for all students**