Program Assignments – While Loops. Develop an IPO for each of the problems below. Place the IPO into your repository. Then write code for each problem and place those files (py) into your repository. Paste the link to your repository into the Assignment Completion Link on Blackboard.

1. Develop flowgorithm and code this problem. Display the odd numbers starting at 1 and ending with 25. Use a while loop structure for this problem.

|  |  |  |
| --- | --- | --- |
| **Input** | **Process** | **Output** |
| Startnumber=1 | while startnumber less than or equal to 25, display startnumber  compute starnumber by summing 2 to starnumber |  |

#Input Phase

startnumber=1

#Process phase

while startnumber <= 25:

print("The Odd number is: ", startnumber)

startnumber=startnumber+2

1. Allow the user to enter a start value, stop value and increment value from the keyboard. Display all the numbers from the start value to stop value using the increment value as you proceed. Use a while loop structure for this problem.

|  |  |  |
| --- | --- | --- |
| **Input** | **Process** | **Output** |
| Enter Startvalue  Enter Stopvalue  Enter incrementvalue | while startvalue less than or equal to stopvalue, display star-value  compute startvalue by summing to starvalue the incrementvalue | starvalue |

#Input Phase

startvalue=float(input("Input the Start Value: "))

stopvalue=float(input("Input the Stop Value: "))

incrementvalue=float(input("Input the Increment Value: "))

#Process phase

while startvalue <= stopvalue:

print("The value is: ", startvalue)

startvalue=startvalue+incrementvalue

1. Prompt the user on whether they want to do this program **(just before the while loop**). “Yes” entry means they want to continue. Any other response indicates they will stop the program. This response is the loop control variable. If the user answers “Yes “then go into the while loop.

Once in the while loop. You are to prompt the user for their last name and two exam scores. Compute the average exam score. Display last name and average. After the loop, display a count of the number of students who entered data.

Finally, the **last statements** **within the while loop** will ask the user if they want to do this loop again. In other words the user needs to be prompted again. The reason is that the end of the loop takes execution to the while condition to be evaluated again. It can not take us to the first few lines of code that prompt the user for the first time. That code is out of the loop. Therefore, we need a second prompt at the bottom, inside the loop.

|  |  |  |
| --- | --- | --- |
| **Input** | **Process** | **Output** |
| Want\_to\_continue  Last\_name  exam\_score1  exam\_score2 | Make count=0  Input first line of data want\_to\_continue  while want\_to\_continue equal to ”yes”.  Enter the last name, enter exam\_score1, enter exam\_score2.  compute average by summing exam\_score1 to exam\_score2 and divide the result by 2  compute count equal to count plus one  display the last name and the average  ask the user if he wants to continue, If the user answer yes, continue the while loop, if the user answer No,  ask if count greater than zero, if yes  Display the message “the number of students who entered the data is, and display the variable count  If not, display no data entered | Last\_name  Average  count |

#Input Phase

want\_to\_continue=input("Do you Want to Continue? Yes/No: ").lower()

count=0

#Process phase

while want\_to\_continue=="yes":

last\_name=input("Enter your Last Name: ")

exam\_score1=float(input("Enter your First Exam Score: "))

exam\_score2=float(input("Enter your Second Exam Score: "))

average=(exam\_score1+exam\_score2)/2

count=count+1

print("Your Last Name is: ", last\_name)

print("Your Average is: ", average)

want\_to\_continue=input("Do you Wanto to Continue? Yes/ No ").lower()

# Output Phase

if count>0

print("The Number of Students who Entered the Data is: ", count)

else:

print("No Data Entered”)

1. Prompt the user on whether they want to do this program **(just before the while loop**). Yes means they want to continue. Any other response indicates they will stop the program. This response is the loop control variable. If the user answers Yes then go into the while loop.

Once in the while loop. You are to prompt the user for employee last name, hours worked and rate of pay. Compute gross pay. Give the employee time and a half for hours worked over 40. Sum the gross pay and count the number of employees.

For each employee display their last name and gross pay.

After the loop (all data entered) display the sum of all the gross pays, and count of the number of employees. Compute and display the average pay.

Finally, the **last statements** **within the while loop** will ask the user if they want to do this loop again. In other words the user needs to be prompted again. The reason is that the end of the loop takes execution to the while condition to be evaluated again. It can not take us to the first few lines of code that prompt the user for the first time. That code is out of the loop. Therefore, we need a second prompt at the bottom, inside the loop.

|  |  |  |
| --- | --- | --- |
| **Input** | **Process** | **Output** |
| Want\_to\_continue  Employee\_last\_name  Hours\_worked  Rate\_of\_pay | Count=0  Input first line of data Do you want to continue?  while want\_to\_continue equal to ”yes”  ask If the quantity of Hours\_worked  are less or equal to 40  compute gross\_pay by multiplying Hours\_worked by Rate\_of\_pay.  if Hours\_worked is greater than 40, compute overtime\_hour by substracting 40 from hours\_worked. Then compute overtime\_hour by multiplying overtime\_hour by1.5. After compute gross\_pay by summing hours\_worked to overtime\_hour and multiply this by rate\_of\_pay  display the employee last name and the gross pay  compute total\_gross\_pay by summing total\_gross\_pay with gross pay  compute count by summing one to count.  ask the user if he wants to continue if not,  ask if count greater than zero, if yes  compute total\_average by dividing total\_gross\_pay into count  display the number of employees, the total gross pay and the total average  if not, display no data entered | Employee\_last\_name  gross\_pay  Total\_gross\_pay  Total\_average  count |

#Input Phase

want\_to\_continue=input("Do you Want to Continue? Yes/No: ").lower()

count=0

totalgross\_pay=0

overtime\_hour=0

#Process phase

while want\_to\_continue=="yes":

employee\_last\_name=input("Enter your Employee Last Name: ")

hours\_worked=float(input("Enter Hours Worked: "))

rate\_of\_pay=float(input("Enter Rate of Pay: "))

if hours\_worked <=40:

gross\_pay=hours\_worked\*rate\_of\_pay

else:

overtime\_hour=hours\_worked-40

overtime\_hour=overtime\_hour\*1.5

gross\_pay=(hours\_worked+overtime\_hour)\*rate\_of\_pay

print("Your Employee Last Name is: ", employee\_last\_name)

print("Your Gross Pay is: ", gross\_pay)

total\_gross\_pay=total\_gross\_pay+gross\_pay

count=count+1

want\_to\_continue=input("Do you Want to Continue? Yes/ No ").lower()

# Output Phase

if count >0:

total\_average=total\_gross\_pay/count

print("The Total Gross Pay of all the Employeees are: ", total\_gross\_pay)

print("The Average Gross Pay of all the Employeees are: ", total\_average)

else:

print("No Data Entered:")

1. Prompt the user on whether they want to do this program **(just before the while loop**). Yes means they want to continue. Any other response indicates they will stop the program. This response is the loop control variable. If the user answers Yes then go into the while loop.

Once in the while loop. You are to prompt the user for quantity and price of an item. Compute extended price (quantity times price of an item. If the extended price is greater than 10.000.00 compute a discount of 25%. All other orders get a 10% discount. For each order display extended price, discount amount (extended price x discount percent), total (extended price – discount amount).

For each order sum the discount amount.

After the loop (all data entered) display the sum of all the discounts.

Finally, the **last statements** **within the while loop** will ask the user if they want to do this loop again. In other words the user needs to be prompted again. The reason is that the end of the loop takes execution to the while condition to be evaluated again. It can not take us to the first few lines of code that prompt the user for the first time. That code is out of the loop. Therefore, we need a second prompt at the bottom, inside the loop.

|  |  |  |
| --- | --- | --- |
| **Input** | **Process** | **Output** |
| Want\_to\_continue  quantity  price\_of\_item | Count=0  Input first line of data Want\_to\_continue  while want\_to\_continue equal to ”yes”  compute extended\_price by multiplying quantity by price\_of\_item  if the extended\_price is greater than 10.000, compute discount by multiplying extended\_price by 0,25, if not compute discount by multiplying extended\_price by 0,10  compute total by subtracting discount from extended\_price  compute total\_discount by summing total\_discount with discount  display extended\_price  display discount  display total  ask the user if he wants to continue, If not ask if count greater than zero, if yes  Display total\_discount  if not, display No Data Entered | extended\_price  discount  total  Total\_discount |

#Input Phase

want\_to\_continue=input("Do you Want to Continue? Yes/No: ").lower()

count=0

total=0

total\_discount=0

#Process phase

while want\_to\_continue=="yes":

quantity=float(input("Enter the Quantity: "))

price\_of\_item=float(input("Enter the Price of the Item: "))

extended\_price=quantity\*price\_of\_item

if extended\_price>10000:

discount=extended\_price\*0.25

else:

discount=extended\_price\*0.10

total=extended\_price-discount

total\_discount=total\_discount+discount

print("Your Extended Price is: ", extended\_price)

print("Your Discount is: ", discount)

print("Your Total is: ", total)

count=count+1

want\_to\_continue=input("Do you Want to Continue? Yes/ No ").lower()

# Output Phase

if count >0:

print("The Total Discount was: ", total\_discount)

else:

print("No Data Entered:")