CIS 106 – Loops Part 2

For each problem prepare an IPO chart. Then write the code for each. Save the IPO within this document and upload to your repository. After code is complete upload the files (.py) to your repository. Paste the link to your repository into the assignment completion link in Blackboard.

1. Allow the user to enter a principle amount and interest rate repeatedly (need a loop to control the program execution). Compute the annual interest (principle x rate). Compute ending balance to be principle (beginning balance + interest). Display year, beginning balance and ending balance for each of the 5 years. Display the accumulated interest for the 5 years. Note: the new balance by year (this will be the principle for the following year. Format the output.

Example:

Enter principle amount: 10000.00

Enter interest rate: 0.10

Year Beginning Ending

Balance Balance

1. $10,000.00 $11,000.00
2. $11,000.00 $12,100.00
3. $12,100.00 $13,310.00
4. $13,310.00 $14,641.00
5. $14,641.00 $16,105.00

Total interest earned: $6,156.00

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
| Principle amount | Loop through 5 years | Year number |
| Interest rate | For each year: |  |
|  | - Compute interest = Principle \* Rate |  |
|  | - Compute Ending balance = Beginning balance + Interest |  |
|  | - Update Principle for next year = Ending balance |  |
|  | Accumulate total interest |  |
|  | Display Year, Beginning balance, Ending balance |  |
|  | Display Total interest earned |  |

1. Fibonacci sequence is a sequence of natural order. The sequence is:

1, 1, 2, 3, 5, 8 etc

Use of for loop compute and display first 20 numbers in the sequence. Hint: start with 1 , 1.

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
| None | Loop through first 20 numbers | Display first 20 Fibonacci numbers |
|  | Calculate next number in sequence as the sum of the previous two numbers | Fibonacci sequence |

1. Create a text file that contains employee last name and salary. Read in this data. Determine the bonus rate based on the chart below. Use that rate to compute bonus. For each line display the employee last name, salary and bonus. After the loop display the sum of all bonuses paid out.

Salary Bonus Rate

100,000.00 and up 20%

50,000.00 15%

All other salaries 10%

Example file (create your own data with at least 5 lines:

Adams

50000.00

Baker

75000.00

Smith

45000.00

Etc

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
| Employee last name | Read employee salary from file | Employee last name, salary, bonus |
| Employee salary | Determine bonus rate based on salary | Sum of all bonuses |
|  | Compute bonus = salary \* bonus rate |  |
|  | Display employee's data |  |
|  | Accumulate total bonuses |  |

1. Create a text file with item, quantity and price. Read through the file one line at a time. Compute the extended price (quantity x price). For each line display the item, quantity, price and extended price. After the loop display the sum of all the extended prices, the count of the number of orders and the average order.

Example Data File

Widget

10

50

Hammer

2

10

Saw

4

8

Etc

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
| Item name | Read item data (name, quantity, price) from file | Item name, quantity, price, extended price |
| Quantity | Compute extended price = quantity \* price | Sum of extended prices |
| Price | Display each order details | Count of orders, Average order |
|  | Accumulate sum of extended prices |  |

1. Create a text file with student last name, district code (I or O) and number of credits taken. Compute tuition owed (credits taken x cost per credit). Cost per credit for in district students (district code I) is 250.00. Out of district students pay 500.00 per credit. For each line display student last name, credits taken and tuition owed. After the loop display sum of all tuition owed and the number of students.

Example file

Jones

I

12

Adams

I

10

Baker

O

12

Smith

O

16

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
| Student last name | Read student data from file (name, district code, credits) | Student name, credits taken, tuition owed |
| District code | Determine cost per credit based on district code | Sum of all tuition owed |
| Number of credits | |  | | --- | | Compute tuition owed = credits \* cost per credit |  |  | | --- | |  | | Number of students |
|  | Display student data |  |