Compiled Group project

By: Madison Lincks, Andrew Duffield, Tatiana Aguilar, Trey Lohrey

Github Repository Code Link: <https://github.com/mlincks3601/SVAD_ivytech_GroupA/blob/main/code_update_04_21.md?plain=1>

Github 3 Testing Scenarios Link: <https://github.com/mlincks3601/SVAD_ivytech_GroupA/blob/main/3_test_situations.md?plain=1>

04-21 Class Group Check in Doc Link: <https://docs.google.com/document/d/1Ra4Z9DyO0LTjPkzgRLZci3j6g-Sn71edNiS2BgDZ27c/edit?usp=sharing>

**Finished Pseudocode**

**// Group project module 6 code update 04-21-25**

**//PLEASE READ NOTES TO UNDERSTAND THIS PROGRAM**

**//PLEASE NOTE, THIS CODE WAS ORIGINALLY DEVELOPED BY MADISON LINCKS 04-01-25 AND THEN GROUP MEMEBRS PLACED NOTES TO EXPLAIN WHAT EACH LINE OF CODE MEANT AND WHAT IT WAS ACHIEVING.**

**----------------------------------------------------------------------------------------**

// Notes for loop usage: The statement "FOR i FROM 1 TO 10 DO" is a loop in pseudocode that is used to repeat a block of code 10 times, once for each value of i from 1 to 10.

//Breakdown: FOR → This initiates a loop.

//i → This is a loop variable (or counter), which keeps track of the number of iterations.

//FROM 1 TO 10 = This means i starts at 1 and increases by 1 each time (by default), stopping at 10.

//DO = Indicates the beginning of the loop's code block.

//END FOR = Marks the end of the loop.

//Example Execution:

//If i starts at 1, the loop executes the statements inside it, then i increases to 2, and the loop runs again.

//This continues until i = 10, at which point the loop stops.

//--------------------------------------------------------------------------------------------

Tatiana Aguilar: Variables and 1st loop (Add notes for what each line of code is doing)

BEGIN VARIABLES

// Define variables for employee input (info we need from the user to calculate)

// Employee variables

ID = INTEGER //Unique employee identifier (Primary Key)

Name = STRING //Employee’s full name (first and last name)

hourly\_rate = FLOAT(x) //Rate of pay per hour for the employee

hours\_worked = FLOAT(x) //Total # of hours worked in the pay period

//Below variables will be used at the end of the program

state\_tax = FLOAT(5.6%) // Calculated as gross\_pay \* 0.056 (5.6%)

federal\_tax = FLOAT (7.9%) // Calculated as gross\_pay \* 0.079 (7.9%)

tax \_deduction = FLOAT(x) // Total tax withheld (state\_tax + federal\_tax)

net\_pay = FLOAT(x) // Take-home pay after deductions(gross\_pay-tax\_deduction)

END VARIABLES

//------------------------------------------------------------------------------------------

// Declare a list aka ARRAY of 10 employees that we will use in a loop later

Employees = [10] AS ARRAY OF Employee // Make an array of 10 Employee records

//-----------------------------------------------------------------------------------------

//Start a loop for user to input employee details and then use our variables from above to help calculate payroll

FOR i FROM 1 TO 10 DO

PRINT "Enter Employee ID: " //Prompt the user to enter the Employee’s ID

INPUT Employees[i].ID

PRINT "Enter Employee Name: " //Prompt the user to enter the Employee’s full name

INPUT Employees[i].Name

PRINT "Enter Hourly Rate: " //Prompt user to enter hourly wage for employee

INPUT Employees[i].hourly\_rate

PRINT "Enter hours\_worked: " //Prompt user to enter hours worked

INPUT Employees[i].hours\_worked

PRINT “------------------------------------------------------”

PRINT “Thank you for your input, you're finished.”

END FOR

//NOTES: index i during each iteration of the loop.

//NOTES: Employees[i] → This accesses the i-th employee in the Employees array (aka a list) during each loop iteration.

Andrew Duffield section below: Calculation loop 2 (Add notes for what each line of code is doing)

//Start the calculations loop

FOR i FROM 1 TO 10 DO

IF Employees[i].gross\_pay = Employees[i].hourly\_rate \* Employees[i].hours\_worked THEN

// state tax is 5.6%

// Federal tax is 7.9%

// Total tax is 13.5%

Employees[i].tax\_deduction = Employees[i].gross\_pay \* .135

Employees[i].net\_pay = Employees[i].gross\_pay - Employees[i].tax\_deduction

// Print results from calculations

PRINT “---------------RESULTS—-------------”

PRINT "Employees[i].gross\_pay"

PRINT "Employees[i].tax\_deduction"

PRINT "Employees[i].net\_pay"

END IF

END FOR

// Display calculated payroll info to the user when finished

PRINT "Payroll Summary"

PRINT "-------------------------------------------"

Trey section: Final output loop from previous data collected and loops ran

//Start the last loop to display the above calculated info for every employee by calling in the loop variables from the 1st loop next to text

FOR i FROM 1 TO 10 DO PRINT

"Employee ID: ", Employees[i].ID PRINT //Print Employee ID

"Employee Name: ", Employees[i].Name PRINT //Print Employee Name

"Gross Pay: ", Employees[i].gross\_payPRINT //Print Gross Pay

"TaxDeduction:", Employees[i].tax\_deduction PRINT "Net Pay: $", Employees[i].net\_pay

PRINT "-------------------------------------------" //Print Tax deduction and Net Pay

//end the loop

END FOR

END

//-----------------------------------------------------------------------------------------

**3 Testing Scenarios**

Scenario 1: Employee with No Dependents

Name: Jessica Parker

Hourly Rate: $20.00

Hours Worked: 40

How the program works for Jessica:

It asks for her ID, name, hourly rate, and hours worked.

Her gross pay is calculated:

20.00 \* 40 = $800.00

Taxes are taken out (13.5% total):

$800 \* 0.135 = $108.00

Her net pay (what she actually gets to take home):

$800 - $108 = $692.00

Code Summary:

Employee ID: 1

Employee Name: Jessica Parker

Gross Pay: $800.00

Tax Deduction: $108.00

Net Pay: $692.00

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

Scenario 2: Employee with 1 Dependent

Name: Maddie Lincks

Hourly Rate: $22.00

Hours Worked: 35

What's different?

We’ll say having 1 dependent gives her a small tax break, maybe 1% off total tax.

Gross pay:

22.00 \* 35 = $770.00

Taxes with 12.5% instead of 13.5%:

770 \* 0.125 = $96.25

Net pay:

770 - 96.25 = $673.75

Code Output:

Employee ID: 2

Employee Name: Maddie Lincks

Gross Pay: $770.00

Tax Deduction: $96.25

Net Pay: $673.75

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

Scenario 3: Employee with 3 Dependents

Name: Jessie Jones

Hourly Rate: $25.00

Hours Worked: 45

What's different?

With 3 dependents, let’s say her tax rate drops more — let’s use 11% total tax.

Gross pay:

25.00 \* 45 = $1,125.00

Taxes at 11%:

1,125 \* 0.11 = $123.75

Net pay:

1,125 - 123.75 = $1,001.25

Code Output:

Employee ID: 3

Employee Name: Jessie Jones

Gross Pay: $1,125.00

Tax Deduction: $123.75

Net Pay: $1,001.25