// =======================

// Included: HW 1a, 1b

// =======================

// HW 1a

// =======================

// Christian Falucho

// CMPR 121

// =======================

#include <iostream>

#include <iomanip>

using namespace std;

/\*================ FUNCTION PROTOTYPES ==================\*/

double getSalesAmt();

double calcCommission(double);

double calcPay(double, double);

void displayPay(double, double, double, double);

void runAgain();

/\*================ FUNCTION PROTOTYPES ==================\*/

/\*

============================================================== MAIN FUNCTION BEGINS ===

===========================================================

\*/

int main () {

// Variable that has the base pay value

double basePay = 2500;

// Variables to store the values from functions

double salesAmt = 0.0;

double commissionAmt = 0.0;

double totalPay = 0.0;

// Call functions, then pass arguments and store the

// values into variables.

salesAmt = getSalesAmt();

commissionAmt = calcCommission(salesAmt);

totalPay = calcPay(commissionAmt, basePay);

// Call display() function, pass the following arguments.

// Arguments to pass - salesAmt, commissionAmt, basePay,totalPay

displayPay(salesAmt, commissionAmt, basePay, totalPay);

// Call runAgain() to ask user to run the program again

runAgain();

}

/\*

============================================================== MAIN FUNCTION ENDS ===

===========================================================

\*/

/\*

============================================================== CODE OUTPUT ===

===========================================================

\*/

A screenshot of a computer screen

AI-generated content may be incorrect.

/\*

============================================================== HELPER FUNCTION BEGINS ===

===========================================================

\*/

/\*================ getSales() begins ======================

Desc:

The function asks for the user's input and stores the input into variable salesAmount and returns the value.

Input:

N/A

Output:

A value that the user has inputted and would be returned to main().

=========================================================\*/

double getSalesAmt(){

double salesAmount = 0.0;

cout << "Enter monthly sales amount: ";

cin >> salesAmount;

cout << endl;

return salesAmount;

}

/\*================== getSales() ends ======================

/\*================= calcCommission() begins ===============

Desc:

The function uses an if-else statement and based on the statement, uses that commission's % to calculate the sales amount.

Input:

Takes in the value of the sales amount.

Output:

Returns the value of commission to main.

=========================================================\*/

double calcCommission(double salesAmount){

double commissionRate = 0.0;

double commissionAmt = 0.0;

if (salesAmount > 50000)

{

commissionRate = 0.02;

commissionAmt = salesAmount \* commissionRate;

return commissionAmt;

}

else if (salesAmount >= 25000 && salesAmount <= 50000)

{

commissionRate = 0.015;

commissionAmt = salesAmount \* commissionRate;

return commissionAmt;

}

return commissionAmt;

}

/\*================ calcCommission() ends ================\*/

/\*================= calcPay() begins ======================

Desc:

Calculates the total monthly pay by adding the

base salary and commission rate.

Input:

Takes in two values, the sales amount and commission rate.

Output:

Return the total pay value to main.

=========================================================\*/

double calcPay(double commission, double basePay){

double totalPay = 0.0;

totalPay = basePay + commission;

return totalPay;

}

/\*================== calcPay() ends =====================\*/

/\*================= displayPay() begins ==================

Desc:

Calculates the total monthly pay by adding the base salary and commission rate.

Input:

Takes in two values, the sales amount and commission rate.

Output:

Return the total pay value to main.

=========================================================\*/

void displayPay(double salesAmt, double commissionAmt, double basePay, double totalPay){

cout << setprecision(2) << fixed << showpoint;

cout << "Monthly Sales: " << setw(3) << "$ " << salesAmt << endl;

cout << "Commission: " << setw(6) << "$ " << setw(9) << commissionAmt << endl;

cout << "Base Pay: " << setw(8) << "$ " << setw(9) << basePay << endl;

cout << "Total Pay: " << setw(7) << "$ " << setw(9) << totalPay << endl << endl;

}

/\*================= displayPay() ends =================\*/

/\*================= runAgain() begins ====================

Desc:

Asks user if they want to run the program again. If not, the program ends.

Input:

N/A

Output:

Asks a question to user

==========================================================\*/

void runAgain(){

string userInput;

cout << "Do it again? (Y/N) ";

cin >> userInput;

if (userInput == "y" || userInput == "Y")

{

main();

}

else if (userInput == "n" || userInput == "N")

{

cout << "Program ending...";

}

}

/\*================= runAgain() ends ====================\*/

/\*

============================================================== HELPER FUNCTION BEGINS ===

===========================================================

\*/// =======================

// Included: HW 1a, 1b

// =======================

// HW 1b

// =======================

// Christian Falucho

// CMPR 121

// =======================

#include <iostream>

#include <iomanip>

using namespace std;

/\*================= FUNCTION PROTOTYPES =================\*/

void getTemps(double&);

double calcAvg(double, double, double);

void displayAvg(double&, double&, double&, double);

/\*================= FUNCTION PROTOTYPES =================\*/

/\*

============================================================== MAIN FUNCTION BEGINS ===

===========================================================

\*/

int main () {

double temp\_1 = 0.0;

double temp\_2 = 0.0;

double temp\_3 = 0.0;

double totalAvg = 0.0;

getTemps(temp\_1);

getTemps(temp\_2);

getTemps(temp\_3);

totalAvg = calcAvg(temp\_1, temp\_2, temp\_3);

displayAvg(temp\_1, temp\_2, temp\_3, totalAvg);

return 0;

}

/\*

============================================================== MAIN FUNCTION ENDS ===

===========================================================

\*/

/\*

============================================================== CODE OUTPUT ===

===========================================================

\*/

A screen shot of a computer

AI-generated content may be incorrect.

/\*

============================================================== HELPER FUNCTION BEGINS ===

===========================================================

\*/

/\*=================== getTemps() begins ==================

Desc:

Ask the user to input 3 temperatures.

Input:

Pass a reference variable, which will be used to store the user's input.

Output:

Display’s text to the user.

=========================================================\*/

void getTemps(double& temp){

cout << "Enter a temperature: ";

cin >> temp;

cout << endl;

}

/\*================= getTemps() begins ==================\*/

/\*================= calcAvg() begins ====================

Desc:

Calculates the average of 3 temperatures.

Input:

Passes in 3 arguments - temp\_1, temp\_2 and temp\_3

Output:

Returns the total average back to main. =========================================================\*/

double calcAvg(double temp\_1, double temp\_2, double temp\_3){

double totalAvg = 0.0;

totalAvg = (temp\_1 + temp\_2 + temp\_3) / 3;

return totalAvg;

}

/\*================= calcAvg() ends ======================\*/

/\*================= displayAvg() begins ==================

Desc:

Formats the information to be properly displayed.

Input:

Takes in 4 arguments - 3 reference variables and 1 pass-by-value.

Output:

Displays the temperatures of 3 cities and the average.

=========================================================\*/

void displayAvg(double& temp\_1, double& temp\_2, double& temp\_3, double totalAvg){

cout << "Enter temperatures of 3 cities." << endl;

cout << setprecision(1) << fixed << showpoint;

cout << "#1: " << setw(5) << temp\_1 << endl;

cout << "#2: " << setw(5) << temp\_2 << endl;

cout << "#3: " << setw(5) << temp\_3 << endl << endl;

cout << "The average temperature is " << totalAvg << degrees." << endl;

}

/\*

============================================================== HELPER FUNCTION ENDS ===

===========================================================

\*/