# Programming Fundamentals

Programming Day - Week 05

## Introduction

Welcome to your favorite day of the week which is programming day🎉. This week, we shall work together to learn and implement new programming concepts.

## Skills to Learn:

- Distinguish between Local and Global Variables
- Categorize the code into meaningful functions to make the code more modular, readable, structured, and reusable.

## Let's do some coding.

**Skill**: Distinguish between Local and Global Variables

### Introduction

So far, we have learned about various kinds of variables depending on datatypes such as int, float, char, etc. However, in a broader sense, the variables can be categorized into two major categories.

1. Local Variables
2. Global Variables

**Local Variables**

Local variables can be accessed inside the functions where they have been declared. These variables can only be accessed inside that function.

Consider the following example:

```
#include <iostream>
using namespace std;
void myFunction()
{
    int x = 20;
}
main()
{
    cout << "The value of the x is: " << x;
}
```

x is declared inside the myFunction() and therefore cannot be accessed inside the main() function

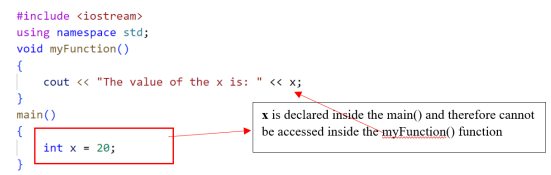
```
D:\PF codes>c++ test03.cpp -o test.exe
test03.cpp: In function 'int main()':
test03.cpp:9:41: error: 'x' was not declared in this scope
    9 |     cout<< "The value of the x is: " << x;
      |                                         ^

D:\PF codes>
```

Here is the other scenario for you to consider:

**Skill**: Distinguish between Local and Global Variables

# Programming Fundamentals

Programming Day - Week 05



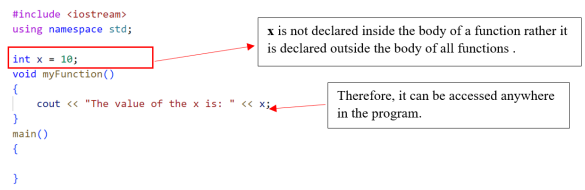

```
D:\PF codes>c++ test03.cpp -o test.exe
test03.cpp: In function 'void myFunction()':
test03.cpp:5:42: error: 'x' was not declared in this scope
    5 |     cout << "The value of the x is: " << x;
      |                                          ^

D:\PF codes>
```

The above-mentioned examples define that the variables declared inside the body of a function cannot be accessed inside the body of other functions.
Such variables are called **Local Variables**.

## Global Variables

Global variables can be accessed anywhere in the program and are not limited to a function only. For example:

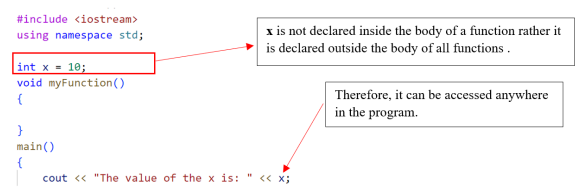
Consider the following example:




```
D:\PF codes>c++ test03.cpp -o test.exe

D:\PF codes>test.exe

D:\PF codes>
```

Can you tell why there is no output?

Here is the other scenario for you to consider:




```
D:\PF codes>c++ test03.cpp -o test.exe

D:\PF codes>test.exe
The value of the x is: 10
D:\PF codes>
```

The above-mentioned examples show that variables declared outside the bodies of functions can be accessed anywhere in the program and therefore are often referred to as **Global Variables**.

## Question!

Can you guess the output of the below-mentioned code snippets?

**Task 01(CA):** Write the below-mentioned program to verify whether your prediction is true or false.

**Skill**: Distinguish between Local and Global Variables

# Programming Fundamentals

Programming Day - Week 05

```cpp
#include <iostream>
using namespace std;

int x = 10;
void myFunction()
{
    int x = 20;
}
main()
{
    cout << "The value of the x is: " << x;
}
```

```cpp
#include <iostream>
using namespace std;

int x = 10;
void myFunction()
{
    cout << "The value of the x is: " << x;
    int x = 20;
}
main()
{
    myFunction();
    int x =30;
    cout << "The value of the x is: " << x;
    myFunction();
}
```

```cpp
#include<iostream>
using namespace std;

int value1 = 10;
int value2 = 20;

int sum (){
  value1 = 40;
  return value1 + value2;
}

main(){

   int x = value1;
   value1 = 100;
   x = 20;
   value2 = sum();
   cout<<value1<<"  "<<value2;

}
```

## Conclusion

| Variables | Description |
|---|---|
| Local Variables | This type of variable can only be **accessed only** inside the function(s) where they have been declared. |
| Global Variables | This type of variable can be **accessed anywhere** in the program. |

**Skill**: Distinguish between Local and Global Variables

# Programming Fundamentals

Programming Day - Week 05

**Congratulations !! You just have learned another skill.**

## Task 02(CP):

Create a function that takes the length, width, height (in meters) and output unit in which you want to see the answer and returns the volume of a pyramid in the correct unit.

**Notes:**

- The units used are limited to: millimeters, centimeters, meters and kilometers.
- Ensure you return the answer and add the correct unit in the format cubic <unit>.

**Test Cases:**

- pyramidVolume(4, 6, 20, "centimeters") → "160000000.000 cubic centimeters"
- pyramidVolume(1843, 1823, 923, "kilometers") → "1.034 cubic kilometers"
- pyramidVolume(18, 412, 93, "millimeters") → "229896000000000.000 cubic millimeters"

```
Enter the length of the pyramid (in meters): 4
Enter the width of the pyramid (in meters): 6
Enter the height of the pyramid (in meters): 20
Enter the desired output unit (millimeters, centimeters, meters, kilometers): meters
The volume of the pyramid is: 160.000000 cubic meters
```

## Task 03(CP):

You've been hired by an Automobile company to write a program to help the tax collector calculate vehicle taxes. Vehicle taxes are based on two pieces of information; the price of the vehicle and the vehicle type code.

$$\text{Tax Amount} = \text{Item Price} * \frac{\text{TaxRate}}{100}$$

The formula of calculating the final price of an item is:
Final Price = Item Price + Tax Amount
Tax rates are in the table below

| Vehicle Type | Vehicle Code | Tax Rate |
|---|---|---|
| Motorcycle | M | 6% |

Skill: Distinguish between Local and Global Variables

| | | |
|---|---|---|
| Electric | E | 8% |
| Sedan | S | 10% |
| Van | V | 12% |
| Truck | T | 15% |

After the tax has been calculated, the program should display the following on the screen;
The final price on a vehicle of type xxx after adding the tax is $xxx.
with xxx replaced by the vehicle type and $xxx with the final price.

Your job is to write a function
float taxCalculator(char type, float price);
and then write the main function for taking the input from the user and then displaying the final output.

```
G:\Semesters\Programming Fundamentals (Fall 2023)\Week 5\PD Tasks>Task3.exe
Enter the vehicle type code (M, E, S, V, T): E
Enter the price of the vehicle: $300
The final price of a vehicle of type E after adding the tax is $324.00.
```

**Task 04(CP):**

A firm gets a request for creating a project for which a certain number of hours are needed. The firm has a certain number of days. During 10% of the days, the workers are being trained and cannot work on the project. A normal working day is 8 hours long. The project is important for the firm and every worker must work on it with overtime of 2 hours per day.
Final answer in hours must be rounded down to the nearest integer (for example, 6.98 hours are rounded to 6 hours).
Write a program that calculates whether the firm can finish the project on time and how many hours more are needed or left.
You have to make a function **projectTimeCalculation** that takes needed hours, days that the firm has and number of workers as input and then returns the string as answer.

**Input Data**

The input data is read from the console and contains exactly three lines:

**Skill**: Distinguish between Local and Global Variables

- On the first line are the needed hours – an integer in the range of [0 … 200 000].
- On the second line are the days that the firm has – an integer in the range of [0 … 20 000].
- On the third line are the number of all workers – an integer in the range of [0 … 200].

**Output Data**

Print one line on the console:

- If the time is enough:
  - "Yes!{the hours left} hours left.".
- If the time is NOT enough:
  - "Not enough time!{additional hours} hours needed.".

| Input | Output | Input | Output |
|---|---|---|---|
| 90<br>7<br>3 | Yes!99 hours left. | 99<br>3<br>1 | Not enough time!72 hours needed. |

```
G:\Semesters\Programming Fundamentals (Fall 2023)\Week 5\PD Tasks>Task4.exe
Enter the needed hours: 99
Enter the days that the firm has: 3
Enter the number of all workers: 1
Not enough time! 72 hours needed.

G:\Semesters\Programming Fundamentals (Fall 2023)\Week 5\PD Tasks>Task4.exe
Enter the needed hours: 90
Enter the days that the firm has: 7
Enter the number of all workers: 3
Yes!99 hours left.
```

**Skill**: Distinguish between Local and Global Variables

# Programming Fundamentals

Programming Day - Week 05

**Skill:** Categorize the code into meaningful functions to make the code more modular, readable, structured, and reusable

**Task05 (Business Project):**

Develop the First version of your individual business application. You have to input 3 entities data, perform some calculation on it and then display the result in tabular form. You have to make the menu based system.

**Task06 (Game Project):**

Develop the First version of your individual Game Project. You have to make 3 types of enemies (horizontal movement, vertical movement, diagonal movement). Also you have to make a player that will move using arrow keys. Also limit the player and enemies in the maze using getCharAtxy function as discussed in the class.

**Good Luck and Best Wishes !!**

**Happy Coding ahead :)**

**Skill:** Categorize the code into meaningful functions to make the code more modular, readable, structured, and reusable