Activity No. <1.3>	
<Hands-on Activity 1.3: Writing First Program using C++ Language>	
Course Code: CPE007	**Program:** Computer Engineering
Course Title: Hands-on Activity 1.3: Writing First Program using C++ Language	**Date Performed:8/28/2025**
Section:CPE11S1	**Date Submitted:8/28/2025**
Name(s):Will Stuart D. Ponce Jr.	**Instructor: Engr. Jimlord Quejado**

1. Output

Check the program below. Find all possible compilation errors and logic errors. Fix them. Your version of the program must print the same result as the expected output. Before you use your compiler, try to find the errors only by manual code analysis.

```
#include <iostream>
        int main()
        {
        cout("The value of five is:"<< 5int);
        return 0;
        }
```

Issues Identified:

Missing using namespace std; or the std: prefix before cout.

Incorrect usage of cout: It is being called like a function.

5int is invalid syntax: 5 is already an integer literal, and int after it is incorrect.

Output:

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

int main()
{
    cout << "The value of five is: " << 5;
    return 0;
}
```

```
The value of five is: 5

=== Code Execution Successful ===
```

2.Output

Check the program below. Find all possible compilation errors and logic errors. Fix them. Your version of the program must print the same result as the expected output. Before you use your compiler, try to find the errors only by manual code analysis.

2. **Check the program below**. Find all possible compilation errors and logic errors. Fix them. Your version of the program must print the same result as the expected output. B
you use your compiler, try to find the errors only by manual code analysis.

```
int main()
{
cout<<"The value of six is:"<<16,0-10-;
return 0;
}
```

Example output

The value of six is: 6

Errors Identified:

cout is used without #include <iostream> or using namespace std;

0-10-; is invalid syntax — you cannot have a dangling - operator.

Comma operator misuse: << 16, 0 - 10 -; causes confusion. In C++, the comma operator evaluates both expressions but returns the last one.

Incorrect calculation or logic: You want to print the value 6.

Output:

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

int main()
{
    cout << "The value of six is: " << 6;
    return 0;
}
```

```
The value of six is: 6

=== Code Execution Successful ===
```

3.Output

Check the program below. Find all possible compilation errors and logic errors. Fix them. Your version of the program must print the same result as the expected output. Before you use your compiler, try to find the errors only by manual code analysis. If you want to improve the variable names, then do so, but remember that variable names have to be as descriptive as possible, and also as short as possible.

3. Check the program below. Find all possible compilation errors and logic errors. Fix them. Your version of the program must print the same result as the expected output. Before you use your compiler, try to find the errors only by manual code analysis. If you want to improve the variable names, then do so, but remember that variable names have to be as descriptive as possible, and also as short as possible.

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

int main()
{
int simpleVariable = 10;
cout<<"The value of ten is:"<<otherVariable);
return 0;
}
```

Example output

The value of ten is: 10

Errors Identified:

other Variable is undeclared: The variable that should be printed is named simple Variable.

Logic mismatch: The code intends to print the value 10, which is correctly stored in simple Variable but incorrectly printed using an undefined variable.

Output:

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

int main()
{
    int simpleVariable = 10;
    cout << "The value of ten is: " << simpleVariable;
    return 0;
}
```

```
The value of ten is: 10

=== Code Execution Successful ===
```

4.Output

Check the program below. Find all possible compilation errors and logic errors. Fix them. Your version of the program must print the same result as the expected output. Before you use your compiler, try to find the errors only by manual code analysis. If you want to improve the variable names, then do so, but remember that variable names have to be as descriptive as possible, and also as short as possible.

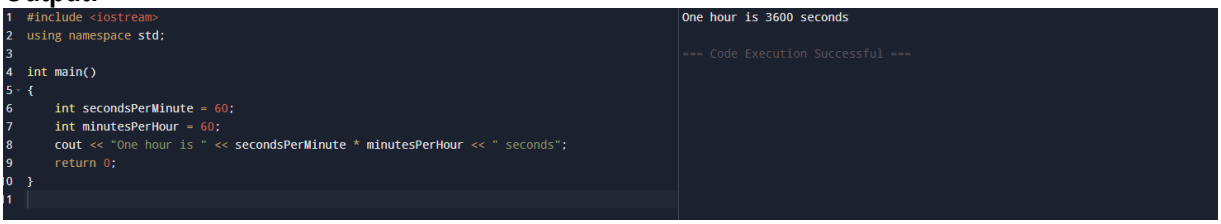
```cpp
#include <iostream>
using namespace std;

int main()
{
int 60seconds = 60;
int 60minutes = 50;
cout<<"One hour is "<<60seconds * 60minutes);
return 0;
}
```

Errors Identified:

Invalid variable names: 60seconds and 60minutes are invalid because variable names cannot start with digits in C++.

Incorrect value for 60minutes: It should be 60, not 50, to correctly compute the seconds in an hour.

Missing text in the output: Expected output is "One hour is 3600 seconds", but the " seconds" string is missing.
Output:



5. Output:
Check the program below. Find all possible compilation errors and logic errors. Fix them. Your version of the program must print the same result as the expected output. Before you use your compiler, try to find the errors only by manual code analysis. If you want to improve the variable names, then do so, but remember that variable names have to be as descriptive as possible, and also as short as possible.

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

int main()
{
int ip Part1 = 027;
int ip Part2 = 0;
int ip Part3 = 0;
int ip Part4 = 1;
cout<<"Localhost IP is "<< ip Part1, ip Part2, ip Part3, ip Part4);
}
```

Example output

Localhost IP is 127.0.0.1

Errors Identified:

Invalid variable names: You cannot write int ip Part1 — it must be int ipPart1 or similar. No spaces allowed in variable names.

Octal literal: 027 is interpreted as octal (base 8), which equals 23 in decimal. You need it to be 127, so you must use 127 directly.

Incorrect cout syntax: Commas do not chain output in cout. You must use << between each value and also insert '.' between IP parts.

Output:

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

int main()
{
    int ipPart1 = 127;
    int ipPart2 = 0;
    int ipPart3 = 0;
    int ipPart4 = 1;

    cout << "Localhost IP is "
         << ipPart1 << "."
         << ipPart2 << "."
         << ipPart3 << "."
         << ipPart4;

    return 0;
}
```

```
Localhost IP is 127.0.0.1

=== Code Execution Successful ===
```

7. Supplementary Activity

8. Conclusion: It's hard to analyze some output and it's kinda challenging in creating a Output and the Errors of the example and I know this things will improve In the near future sometimes you need to learn more about this things so in the assignment it will be ez.

9. Assessment Rubric