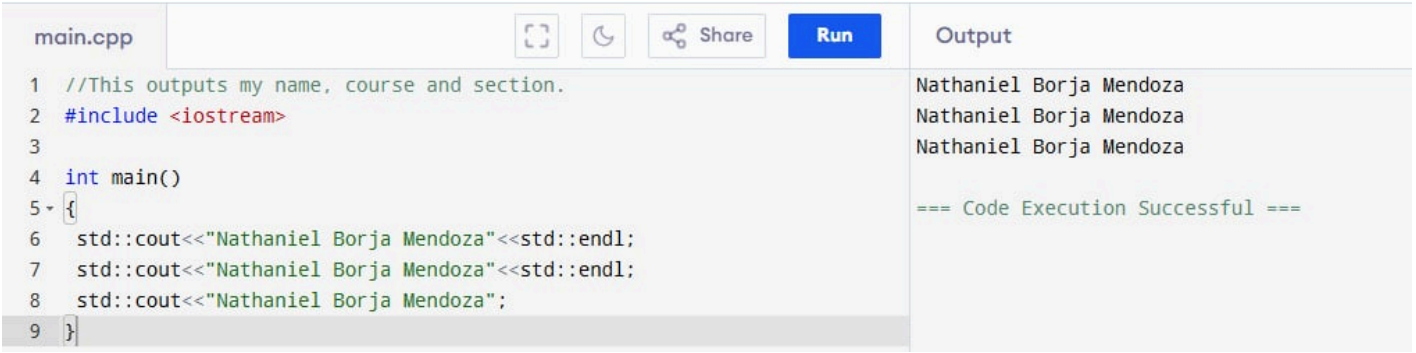
Activity No. 1.3	
Writing First Program Using C++ Language	
Course Code: CPE007	**Program:** Computer Engineering
Course Title: Programming Logic and Design	**Date Performed:** 8/29/2025
Section: CPE11S1	**Date Submitted:** 9/1/2025
Name(s): Mendoza, Nathaniel Borja	**Instructor: Engr. Jimlord M. Quejado**

6. Output

Exercise 4.1

Exercise 4.2



7. Supplementary Activity

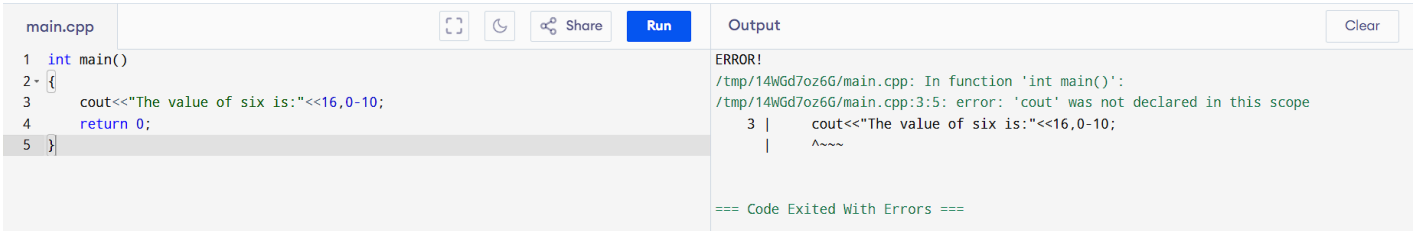
1. Given Code and Output

Errors:

1. Line 2 is missing “using namespace std”. Or did not use it and still proceeds to write without “std::” after cout.
2. Line 5 used invalid use of parenthesis “()” with cout.
3. Line 5 invalid 5int. Must be after “{“ and do int five = 5 or int x = 5, number works as well.
4. Line 5 did not use “<<” after cout. Also missing cout before semicolon.
5. Did not use the end line or \n for code clarity. Optional in this situation but better to be included.

Correct Code and Output

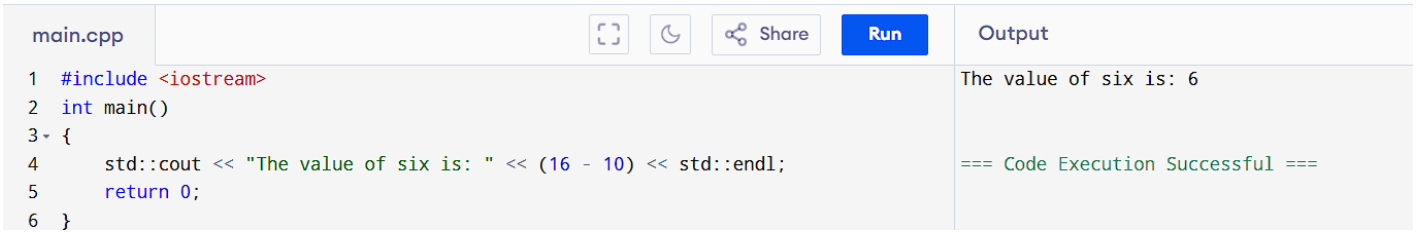
2. Given Code and Output



Errors:

1. Missing “ #include <iostream>.
2. Missing “using namespace std”. Or did not use it and still proceeds to write without “std::” after cout.
3. Line 3 wrong use of Arithmetic Operations and use of integers. If the expected value is “6” then, it must be (16 - 10) or 16 is subtracted by 10 will equal to 6. Therefore to conclude a logical error.

Correct Code and Output

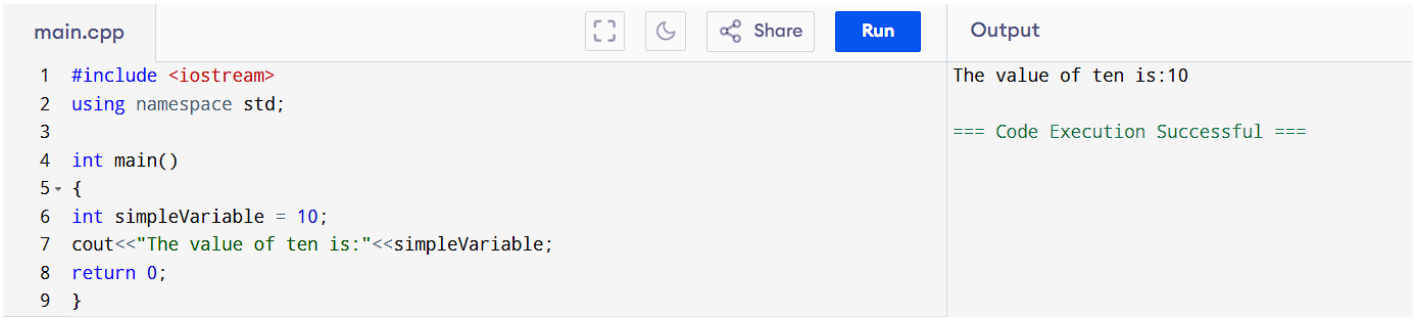


3. Given Code and Output

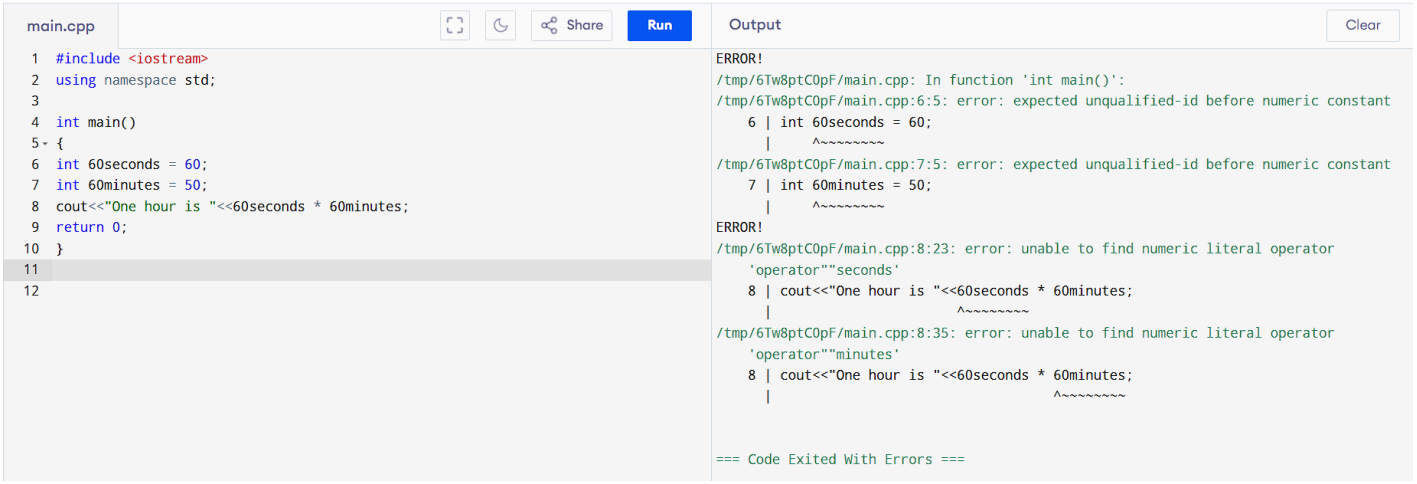
Errors:

1. Line 7 used “otherVariable” instead of “simpleVariable”.

Correct Code and Output



4. Given Code and Output



Errors:

1. Line 6 and 7 variables cannot start with digits. (60 seconds and 60 minutes is invalid).
2. Line 7 Logical Error since the written integer was 60minutes = 50 which is invalid.
3. Line 8 missing output of seconds. (Expected Output)
4. Line 8 incorrect use of Arithmetic Operations. Must be (seconds * minutes) without the digits.

Correct Code and Output

5. Given Code and Output

main.cpp

```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;
}
```

Output

```
ERROR!
/tmp/6ICs8xXAvw/main.cpp: In function 'int main()':
/tmp/6ICs8xXAvw/main.cpp:6:8: error: expected initializer before 'Part1'
    6 | int ip Part1 = 027;
      |        ^~~~
/tmp/6ICs8xXAvw/main.cpp:7:8: error: expected initializer before 'Part2'
    7 | int ip Part2 = 0;
      |        ^~~~
/tmp/6ICs8xXAvw/main.cpp:8:8: error: expected initializer before 'Part3'
    8 | int ip Part3 = 0;
      |        ^~~~
/tmp/6ICs8xXAvw/main.cpp:9:8: error: expected initializer before 'Part4'
    9 | int ip Part4 = 1;
      |        ^~~~
ERROR!
/tmp/6ICs8xXAvw/main.cpp:10:28: error: 'ip' was not declared in this scope
   10 | cout<<"Localhost IP is "<< ip Part1, ip Part2, ip Part3, ip Part4;
      |                            ^~


=== Code Exited With Errors ===
```

Errors:

1. Invalid use of variables, which is the use of spaces between variables like “ipPart 1,2,3 and 4”.
2. Wrong integer “027” it will be read as octal in c++ language and be translated to 23.
   Why did It become 23? 027(8) = 2×8^1 + 7×8^0 = 16 + 7 = 23
3. Wrong use of cout syntax “<<”. The code was written and used commas which is invalid.
4. Line 10 has space between variables, also invalid.

Correct Code and Output

main.cpp

```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 << endl;
    return 0;
}
```

Output

```
Localhost IP is 127.0.0.1


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

8. Conclusion

I was able to accomplish these tasks in a short amount of time which made me realize that I am slowly improving on coding C++ Languages slowly but progressively which is good for my learning as I cannot speed things up since it is a sign of improvement for me as a learner. Therefore I can conclude that I was able to code simple instructions of C++ instructions and debug given programs and fix it myself for it to be executable. It is a good learning for the upcoming exams as well.