Laboratory Work No. 1

C++ Basic Console I/O

This laboratory work covers the following concepts in C++ programming language:

- usage of the `iostream` and `iomanip` libraries for the console input and output
- usage of the output stream object `cout` with stream insertion operator `<<`
- usage of the input stream object `cin` with stream extraction operator `>>`
- usage of the stream manipulators `endl`, `setw`, `setprecision`, `fixed`, `right`, `left`
- C++ style static type conversion using `static_cast<type>(…)`

⇨ Create a Win32 Console application and an empty C++ source file in Visual Studio IDE to be able to start typing programs.

Task-1: Write a C++ program that displays "Hello World!" to the console screen for each case as specified below.

a. Using a "`std::cout`" for each cout statement
b. Using a "`using std::cout`" for solving namespace problem regarding the `cout` object
c. Using a "`using namespace std`" for making all standard namespace object usages possible without using "`std::`"

Task-2: Write a C++ program adds two user-specified integer values and displays the result to the screen.

Sample scenario:

```
Enter first value:  3
Enter second value:  4
Addition of 3 and 4 is 7
```

Try to obtain the last line as below.

```
3 + 4 = 7
```

Task-3: Write a C++ program that uses a couple of `cout` statements to print the following values to the console screen.

```
4                   // print an integer
_ _ 4               // print an integer within 3 spaces
4 _ _               // print an integer within 3 spaces but left adjusted
3.141559            // print a floating-point value
3.14                // print a floating-point value using 2 digits after the decimal point
_ _ 3.14            // print a floating-point value within 6 spaces, 2 digits after the decimal point
```

Task-4: Try the following code snippet and explain how each line works.

```
int a = 2;
char b = 'f';
float c = 3.1415f;
double d = 3;

cout << setw(3) << a << endl;
cout << setw(3) << left << a << endl;
cout << setw(3) << right << a << endl;
cout << '\t' << a << '\t' << b << '\t' << c << endl;
cout << setw(9) << a << setw(8) << b << setw(13) << c << endl;
cout << d << '\t' << setprecision(1) << d << '\t' << fixed << setprecision(1) << d << endl;
cout.unsetf( ios::fixed );
cout << d << endl;
```

Task-5: Try the following code snippet and explain how each line works.

```
int a = 3;
char b = 'f';
cout << a << '\t' << static_cast<char>(a) << endl;
cout << b << '\t' << static_cast<int>(b) << endl;
cout << ( 2/3 ) << '\t' << ( static_cast<float>(2) / 3 ) << endl;
```