CS-181 Lab 2

Student Learning Outcomes

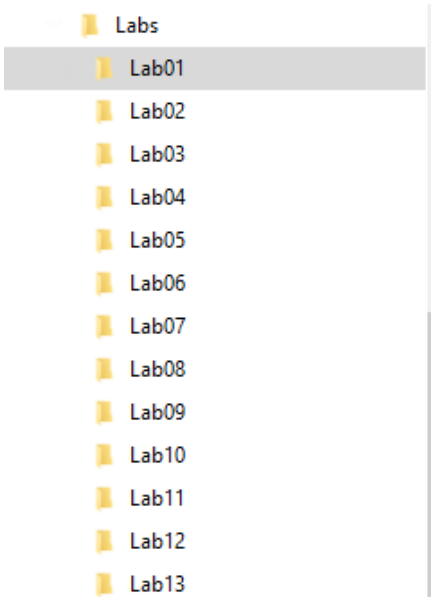
- Gaining experience with the Microsoft Visual Studio Integrated Development Environment (IDE)
- Creating C++ projects in MS Visual Studio
- Using the code editor
- Sequential code
- Console input and output

Overview

This lab provides a hands-on opportunity to apply some of the terminology and concepts presented in chapter 2. In this lab, you will have multiple exercises to complete. For each exercise, you will need to create a new project in MS Visual Studio. Be sure to create the project for each exercise in your Lab02 folder.

In the software industry, it is a common practice to place a comment header at the top of each source code file that has things like the file name, purpose, developer, and revision history. With this lab, you will also start placing a comment header at the top of each file. The header is provided as a text file and you may simply copy and paste its contents into each Visual Studio C++ file you create.

Quick throwback to Lab 1: Before starting this lab, make sure you have created a folder structure for your labs like the one shown below. As mentioned before in Lab 1, having your work organized is a critical survival skill in software development! If you are using the college lab computers, be sure to create this on your USB drive and be sure to save to it! Saving on the C: drive of the lab computers is a disaster waiting to happen! If you are working on your own computer, you may use the C: drive or your USB drive.



If you haven't done so already, download and install the Microsoft Visual Studio. See the instructions provided in Lab 1. Better yet, make sure you completed Lab 1!

Exercise 1: Fix the mixed up code

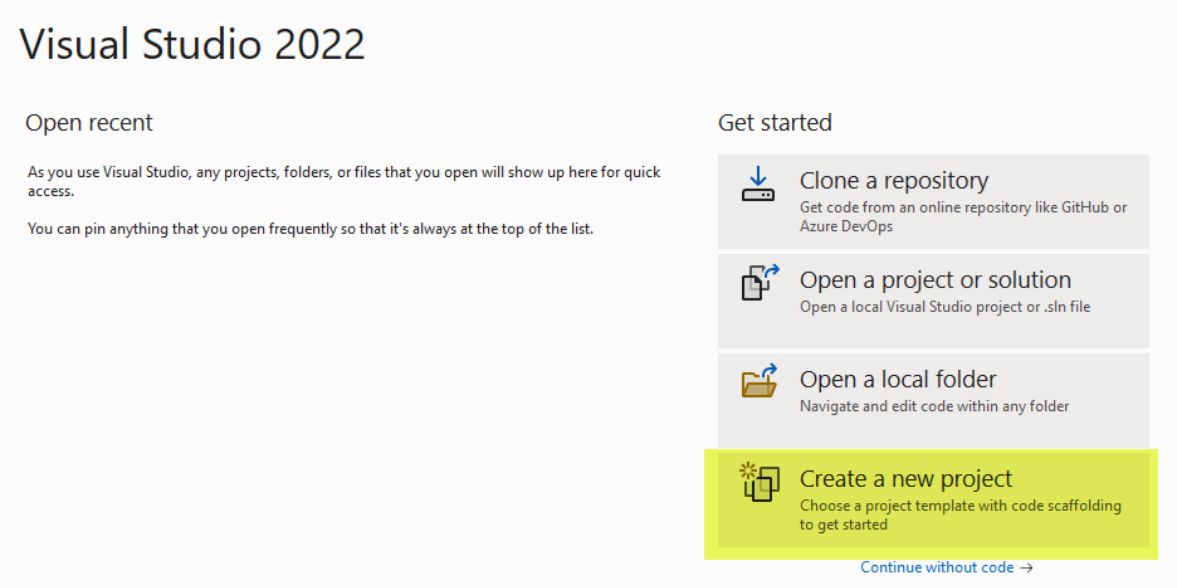
Given this totally mixed-up mess of C++ code from page 30:

```
int main()
}
// A crazy mixed up program
return 0;
#include <iostream>
cout << "In 1492 Columbus sailed the ocean blue.";
{
using namespace std;
```

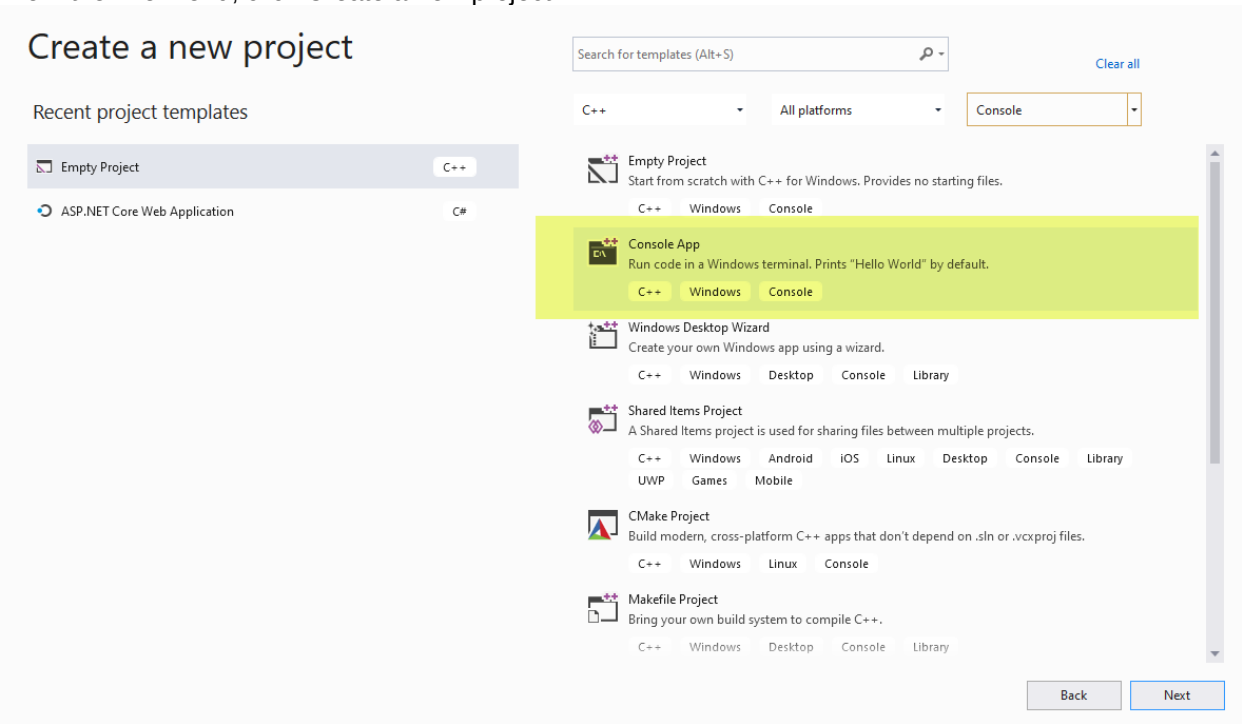
Rewrite the code so the program compiles, runs and produces the correct output:

Steps

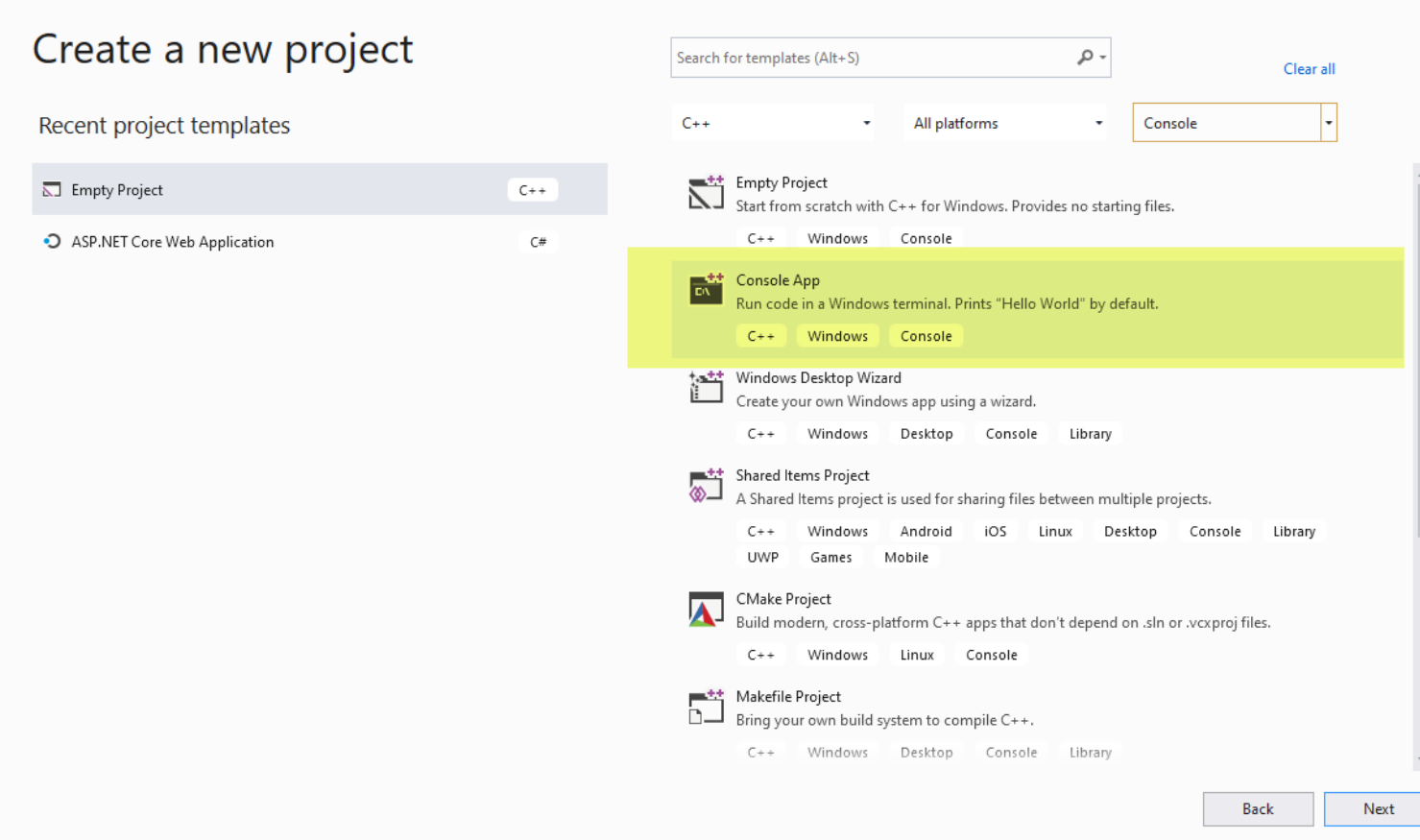
1. Start MS Visual Studio.



2. From the File menu, click Create a new project.



3. Locate and select Console App (C++)and click the Next button



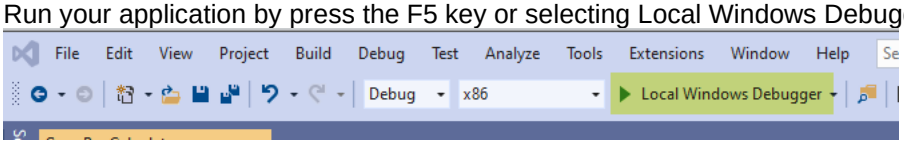
4. Browse to and select the Lab02 folder. Name the project Lab2Exercise1, and click the Create button:

5. Remove the “Hello World” code and extra comments. Copy and paste the program header at the top. Fill in the lab number, exercise number/short description and your name.

6. With the code given at the top, rewrite the program so it compiles, runs, and produces the correct output. Hint: the MS Visual Studio project wizard will do some of the work for you!

7. Be sure to save your work!

8. Run your application by press the F5 key or selecting Local Windows Debugger.



9. Verify you get the correct output below:

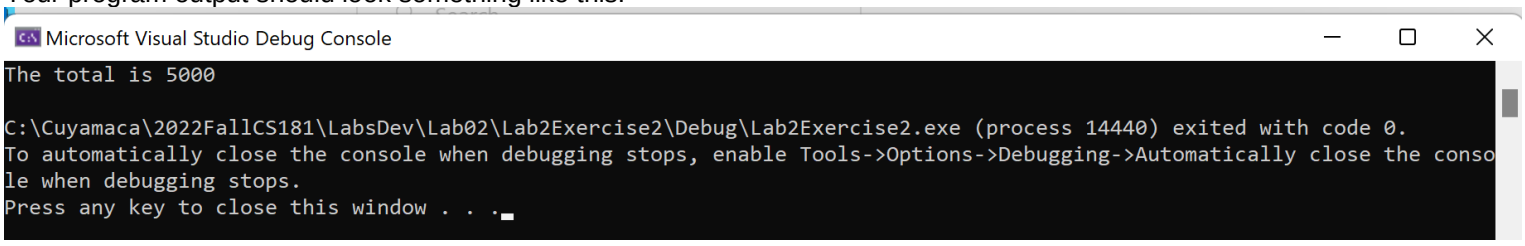
```
In 1492 Columbus sailed the ocean blue.
```

Exercise 2: Product of Two Numbers

Now that you’ve done a couple exercises between Lab 1 and the last exercise, the detailed instructions on how to create a project in Visual Studio, coding and running your program are going to be left out.

Write a program that stores the integers 50 and 100 in variables, stores the product (multiplication) of these two in a variable named total, and displays the total.

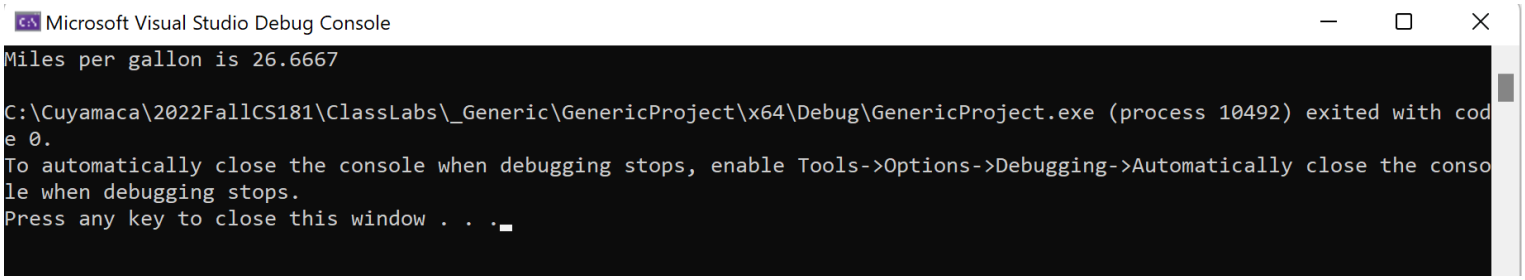
Your program output should look something like this:



Exercise 3: Miles per Gallon

Write a program that computes the miles per gallon of a car that holds 15 gallons of gas and can travel 400 miles before refueling. Hint: MPG = Miles driven / gallons of gas used

Your output should look something like this:



Exercise 4: Annual Pay

An employee gets paid every two weeks and earns $2,200 each pay period. In a year, the employee gets paid 26 times. Write a program that defines the following variables:

payAmount - This variable will hold the amount of pay the employee earns each pay period. Initialize the variable with 2200.0.
payPeriods - This variable will hold the number of pay periods in a year. Initialize the variable with 26.
annualPay - This variable will hold the employee's total annual pay, which will be calculated.

The program must calculate the employee's total annual pay by multiplying the employee's pay amount by the number of pay periods in a year and store the result in the `annualPay` variable. Display the total annual pay on the screen.

Your program output should look something like this:

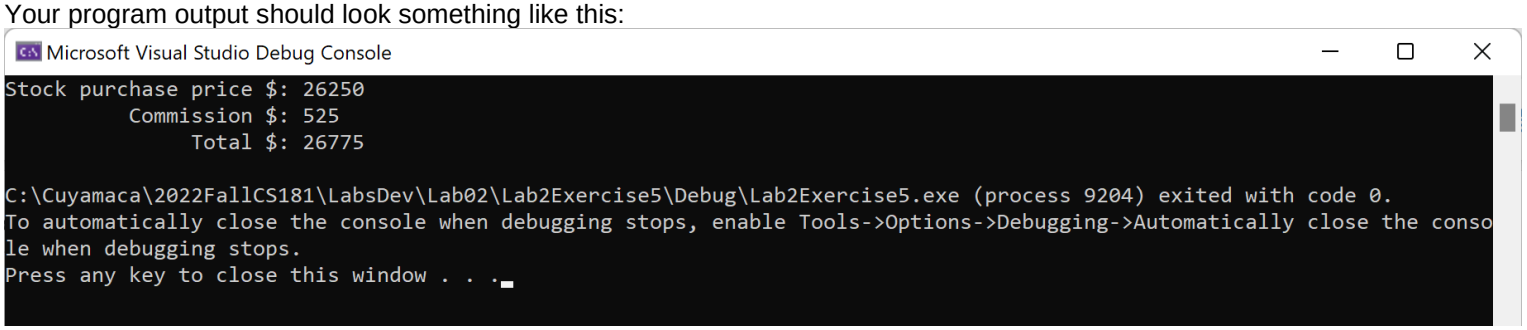
Exercise 5: Stock Commission

You just bought 750 shares of stock at a price of $35 per share. You need to pay your stockbroker a 2% commission for the transaction. Write a program that calculates and displays the following:

Amount paid for the stock without commission.
The commission amount.
Total (stock purchase + commission)

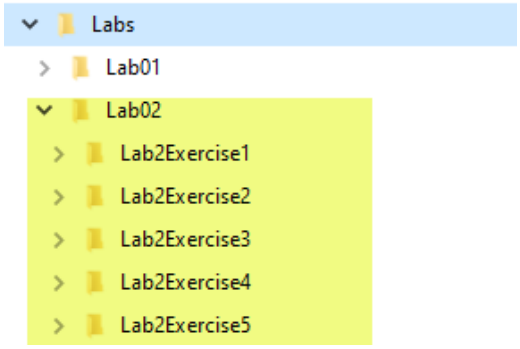
Note: Specific variable names are left to your discretion.

Your program output should look something like this:



Assignment Submission

You can use Windows File Explorer to zip up and submit your work. When all exercises are completed, your Lab02 folder should look like this:



If everything is there, right-click on your Lab02 folder, click “Send to”, and then click “Compressed (zipped) folder”.

Your completed assignment must be uploaded to Canvas in a zip file format. See Lab 1 for detailed instructions.

Grading Criteria:

Deliverable	Points	Breakdown
Exercise 1	**10**	**Coded, opens in MS Visual Studio, compiles, runs and produces correct results.**
Exercise 2	**10**	**Complete, code is clear, descriptive variable names, appropriate use of comments**
Exercise 3	**10**	**Compiles, runs, produces correct output**

Deliverable	Points	Breakdown
Exercise 4	**10**	**Input, process and output clearly identified. Clear programming logic and calculation formulas.**
Exercise 5	**10**	**Complete, code is clear, descriptive variable names, appropriate use of comments**
Lab Total	**50**	