CS 31 Introduction to CS 1 Discussion 2H

Who am I?

Shweta Sood
Learning Assistant : TBA

Office Hours: Thursday - 1:30 – 4:30pm, BH 3256

Discussion: Section 2H, Friday 12:00-1:50pm,
Dodd 147
Email: shwetasood@cs.ucla.edu

What to expect?

- Worksheets: Practicing questions, discussing their solutions
- Solving problems under time-constraints
- Pairing up approach, volunteer based approach

Some wisdom!

- Attend professor’s class (*physically and mentally*) every week.
- Revise the concepts that were taught in the week
- Take notes (either on paper or laptop, as you prefer).

- Attend discussions :D

Some more wisdom!

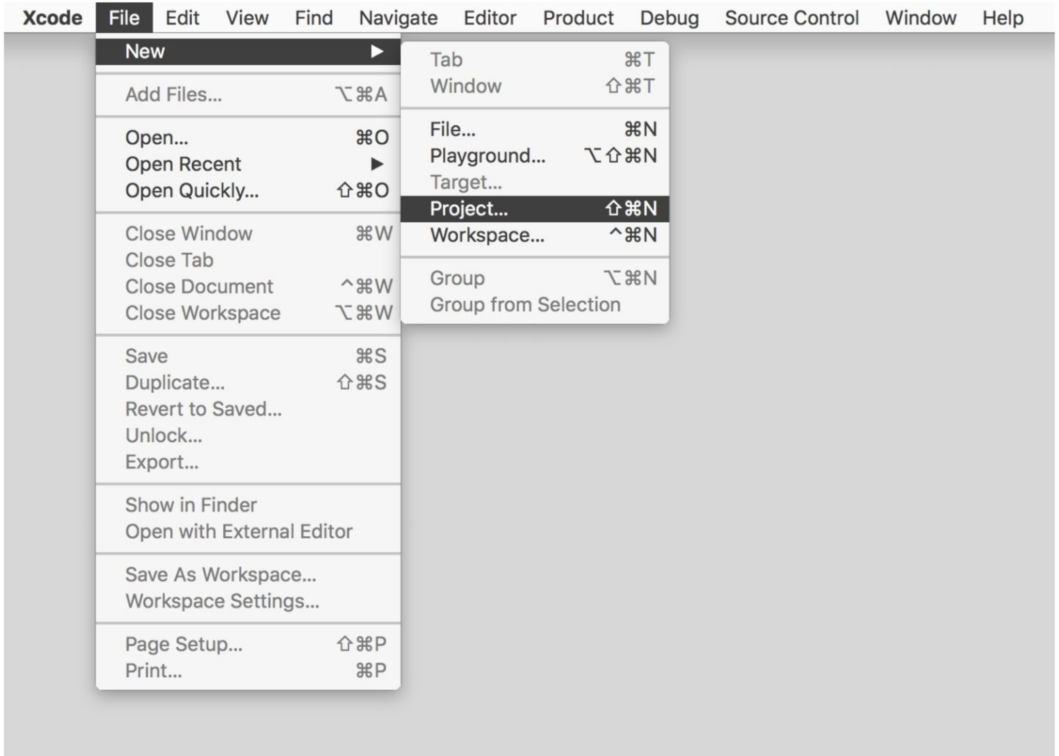
- Start working on projects early on.
- Add bits of code at a time, then compile, and run.
- This will help in identifying bugs.
- SAVE A BACKUP ON THE CLOUD!!!
- Read the spec very carefully (at least 2-3 times).
- Check and re-check before submitting the zip folder.

Objective

1. Demo Xcode
2. Demo Visual Studio
3. Connecting to Linux server
4. Discussion Project 1

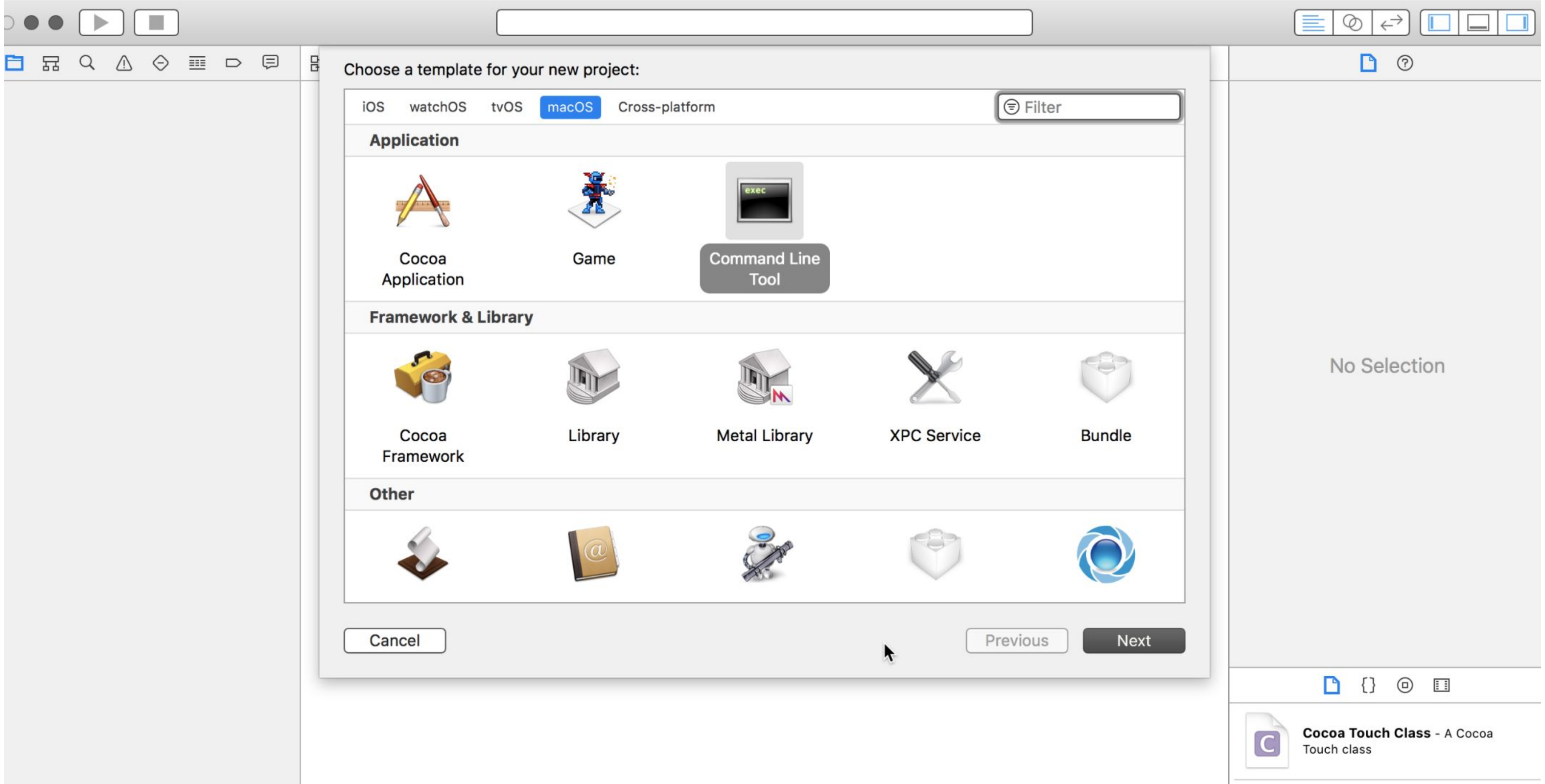
Demo: Xcode

- Step 1: Dismiss the welcome window, and select file -> New -> Project



Demo: Xcode

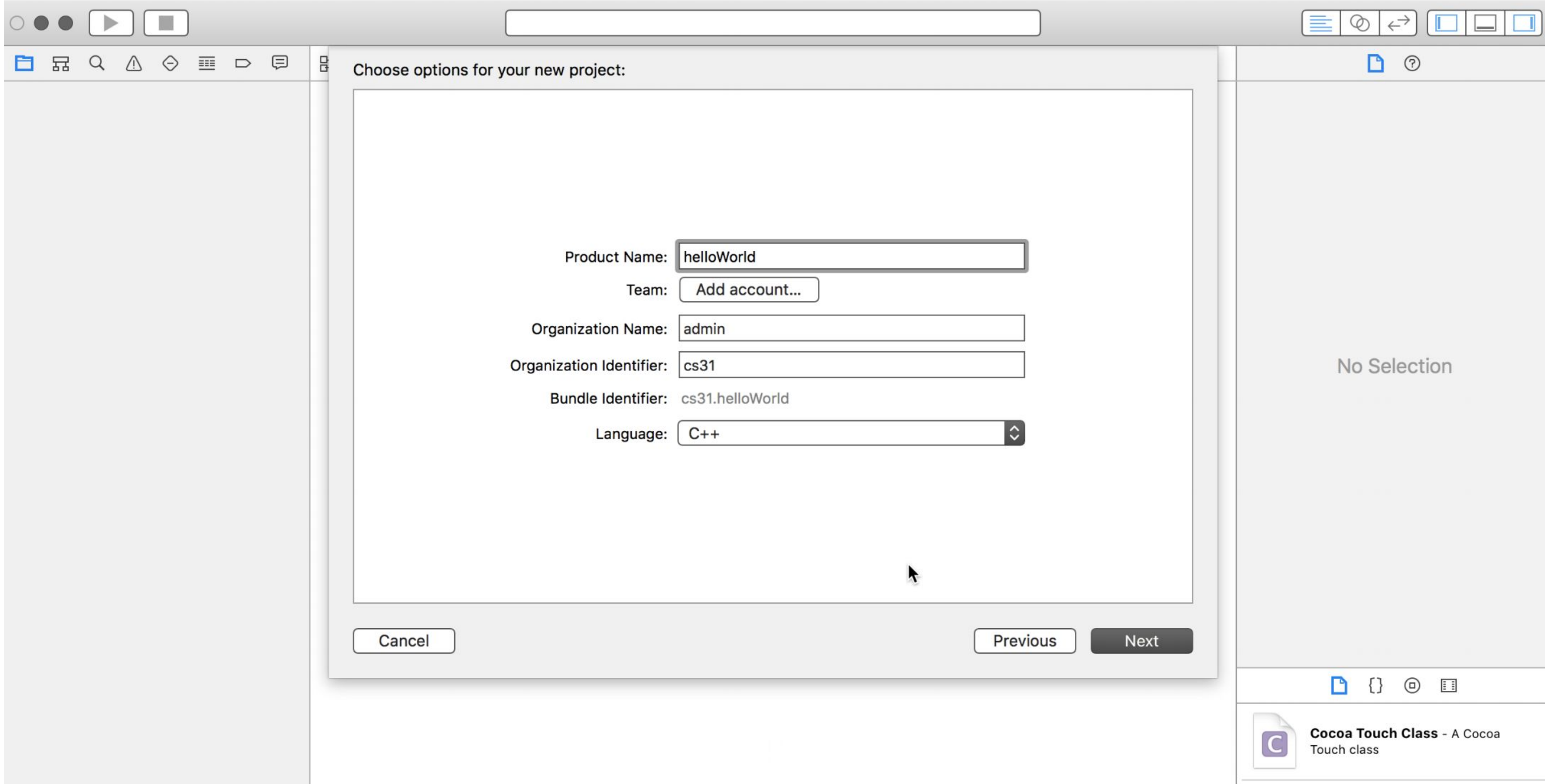
- Step 2: select macOS Application, and Command Line Tool. Click Next.



Choose a template for your new project:

Demo: Xcode

- Step 3: Product Name = your project name, e.g. "helloWorld".



Choose options for your new project:

Product Name: helloWorld

Team: Add account...

Organization Name: admin

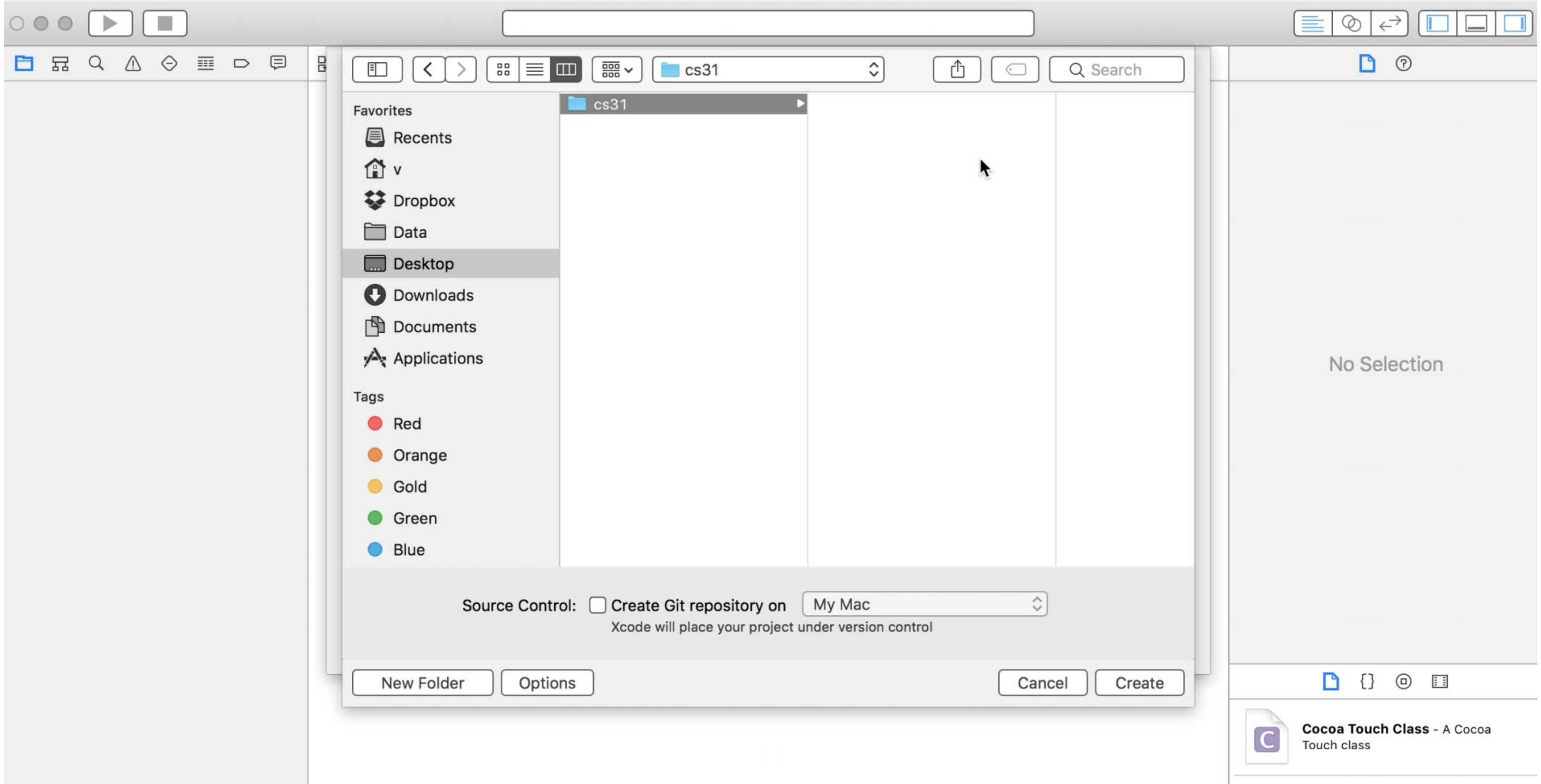
Organization Identifier: cs31

Bundle Identifier: cs31.helloWorld

Language: C++

Demo: Xcode

- Step 4: Choose a folder where you can find for your project. Click Create.

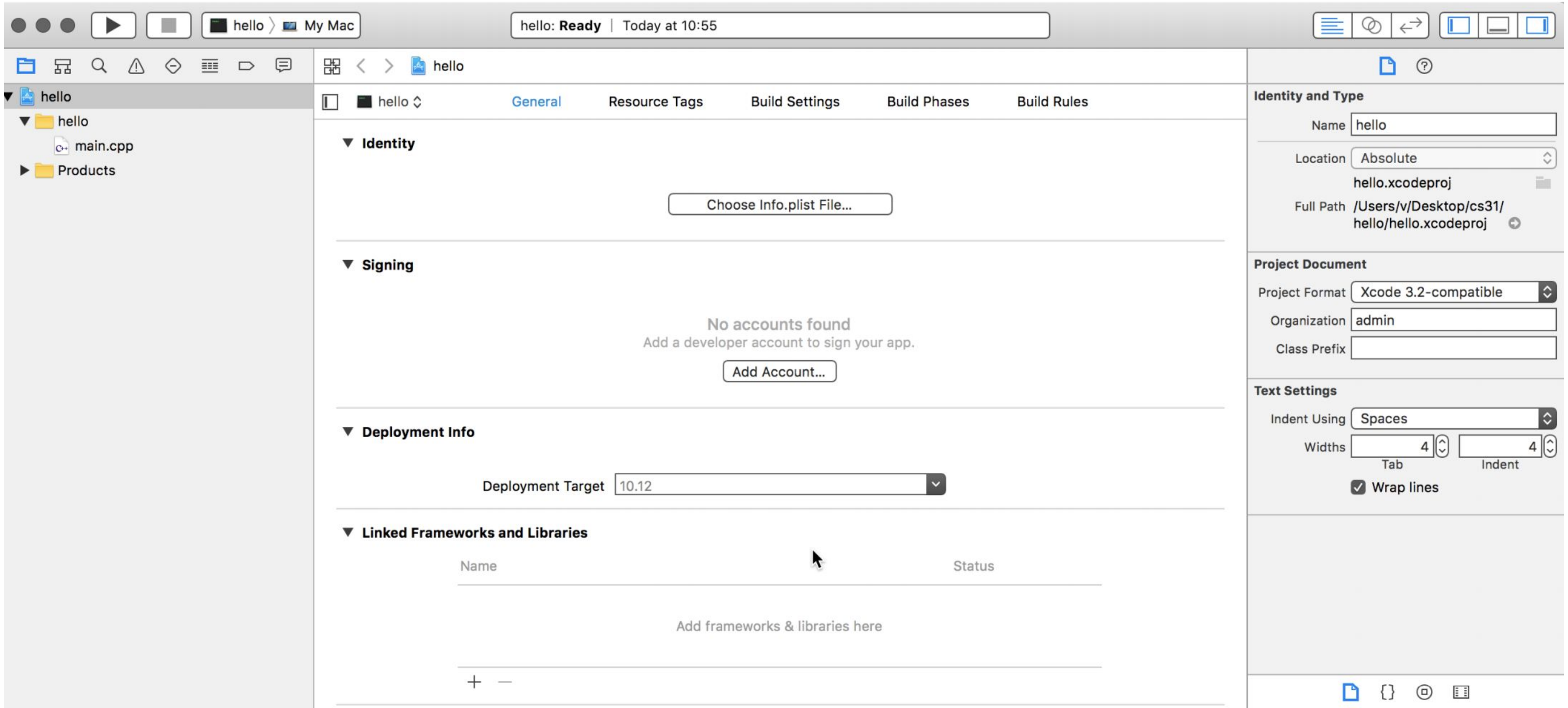


Source Control: □ Create Git repository on My Mac

Xcode will place your project under version control

Demo: Xcode

- Step 4: Choose a folder where you can find for your project. Click Create.



hello: **Ready** | Today at 10:55

hello
- hello
  - main.cpp
- Products

General | Resource Tags | Build Settings | Build Phases | Build Rules

Identity

Choose Info.plist File...

Signing

No accounts found
Add a developer account to sign your app.

Add Account...

Deployment Info

Deployment Target: 10.12

Linked Frameworks and Libraries

Name	Status

Add frameworks & libraries here

Identity and Type

Name: hello
Location: Absolute
hello.xcodeproj
Full Path: /Users/v/Desktop/cs31/hello/hello.xcodeproj

Project Document

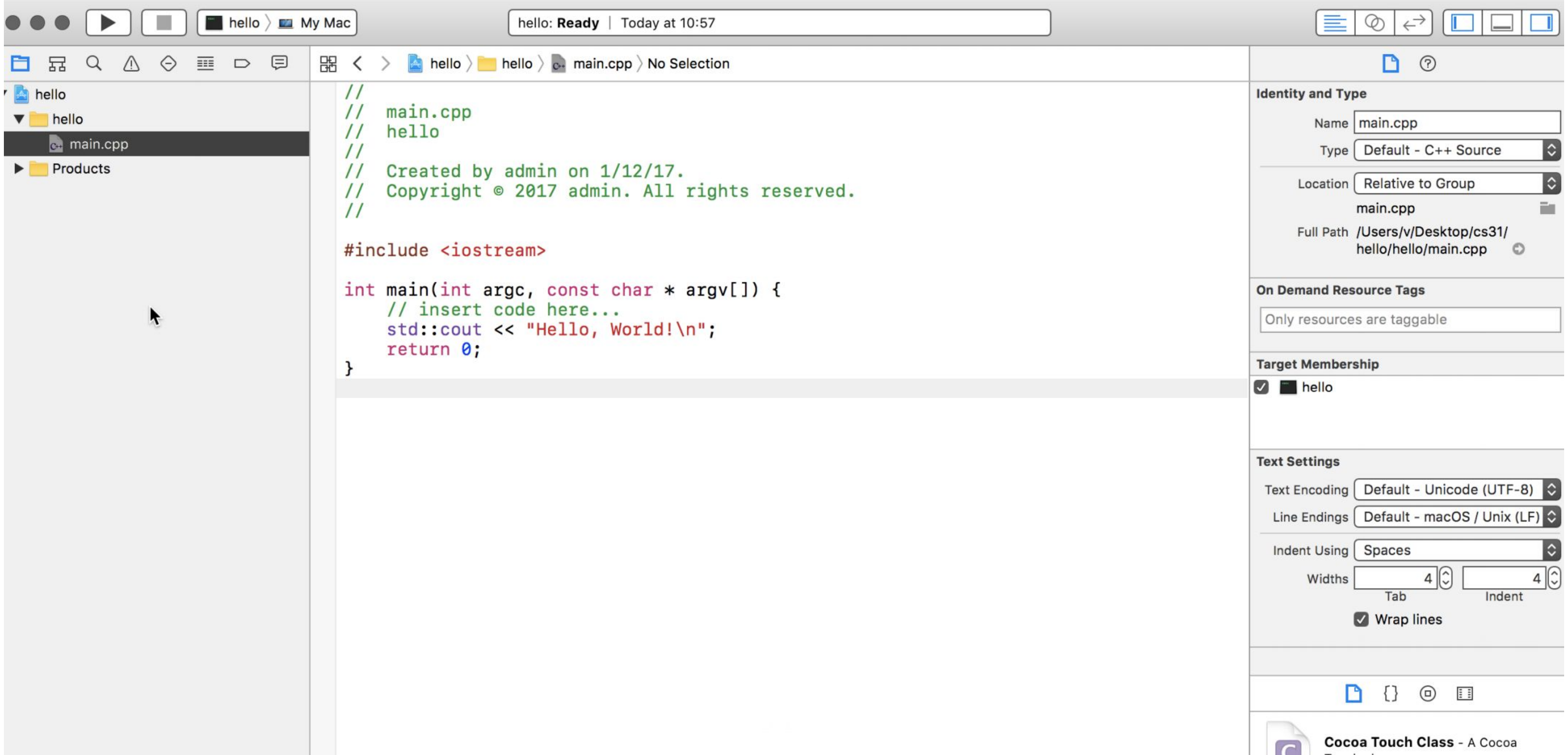
Project Format: Xcode 3.2-compatible
Organization: admin
Class Prefix:

Text Settings

Indent Using: Spaces
Widths: 4 (Tab), 4 (Indent)
☑ Wrap lines

Demo: Xcode

- Step 5: “main.cpp” is generated with some sample code. You may change the name of it.



hello 〉 My Mac
hello: Ready | Today at 10:57
hello 〉 hello 〉 main.cpp 〉 No Selection
hello
hello
main.cpp
Products
Identity and Type
On Demand Resource Tags
Only resources are taggable
Target Membership
Text Settings
Cocoa Touch Class - A Cocoa

```
//
//  main.cpp
//  hello
//
//  Created by admin on 1/12/17.
//  Copyright © 2017 admin. All rights reserved.
//

#include <iostream>

int main(int argc, const char * argv[]) {
    // insert code here...
    std::cout << "Hello, World!\n";
    return 0;
}
```

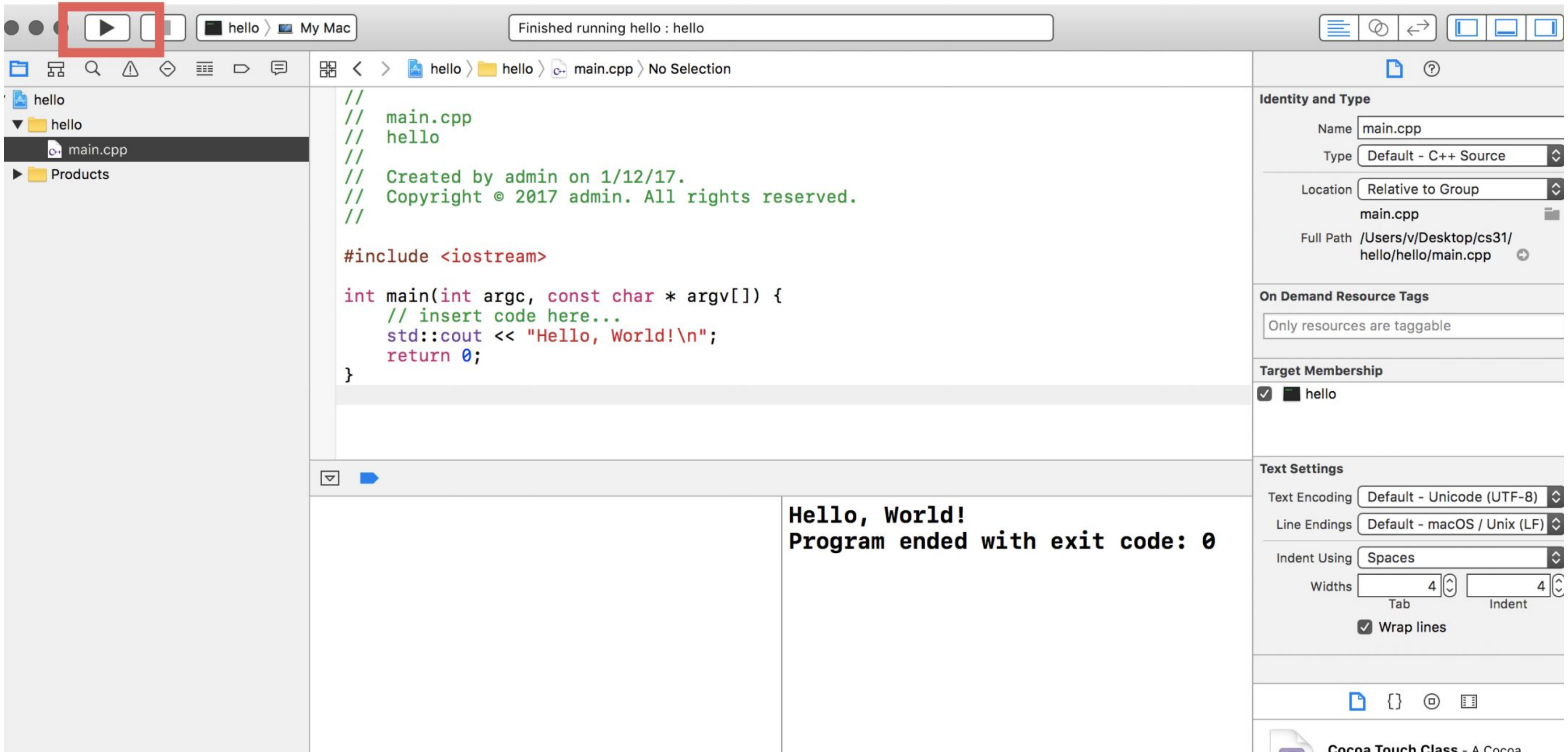
Name: main.cpp
Type: Default - C++ Source
Location: Relative to Group
main.cpp
Full Path: /Users/v/Desktop/cs31/hello/hello/main.cpp

☑ hello

Text Encoding: Default - Unicode (UTF-8)
Line Endings: Default - macOS / Unix (LF)
Indent Using: Spaces
Widths: 4 (Tab) 4 (Indent)
☑ Wrap lines

Demo: Xcode

- Step 6: You may click the triangle button to run it.


hello 〉 My Mac
Finished running hello : hello
hello
hello
main.cpp
Products
hello 〉 hello 〉 main.cpp 〉 No Selection

```
//
//  main.cpp
//  hello
//
//  Created by admin on 1/12/17.
//  Copyright © 2017 admin. All rights reserved.
//

#include <iostream>

int main(int argc, const char * argv[]) {
    // insert code here...
    std::cout << "Hello, World!\n";
    return 0;
}
```

```
Hello, World!
Program ended with exit code: 0
```

Identity and Type

Name: main.cpp

Type: Default - C++ Source

Location: Relative to Group

main.cpp

Full Path: /Users/v/Desktop/cs31/hello/hello/main.cpp

On Demand Resource Tags

Only resources are taggable

Target Membership

☑ hello

Text Settings

Text Encoding: Default - Unicode (UTF-8)

Line Endings: Default - macOS / Unix (LF)

Indent Using: Spaces

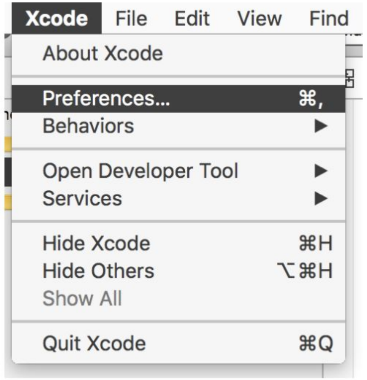
Widths: 4 (Tab) 4 (Indent)

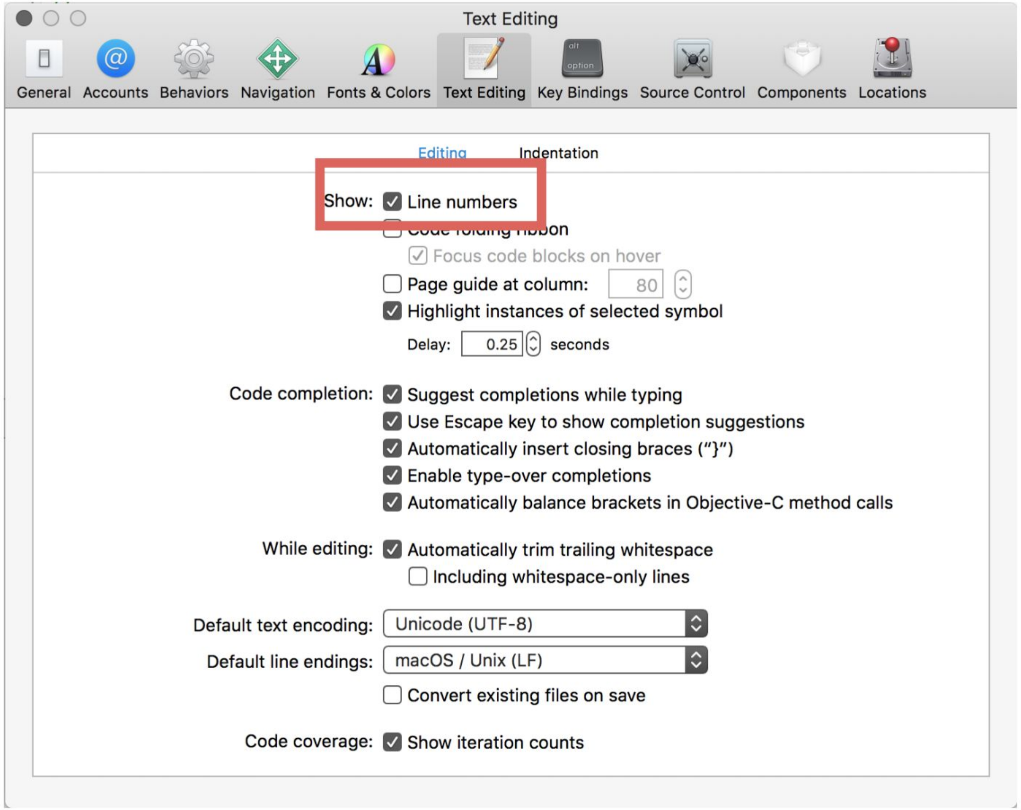
☑ Wrap lines

Cocoa Touch Class - A Cocoa

Demo: Xcode

- Show the line numbers:





Show: ☑ Line numbers
☐ Code folding ribbon
☑ Focus code blocks on hover
☐ Page guide at column: 80
☑ Highlight instances of selected symbol
Delay: 0.25 seconds

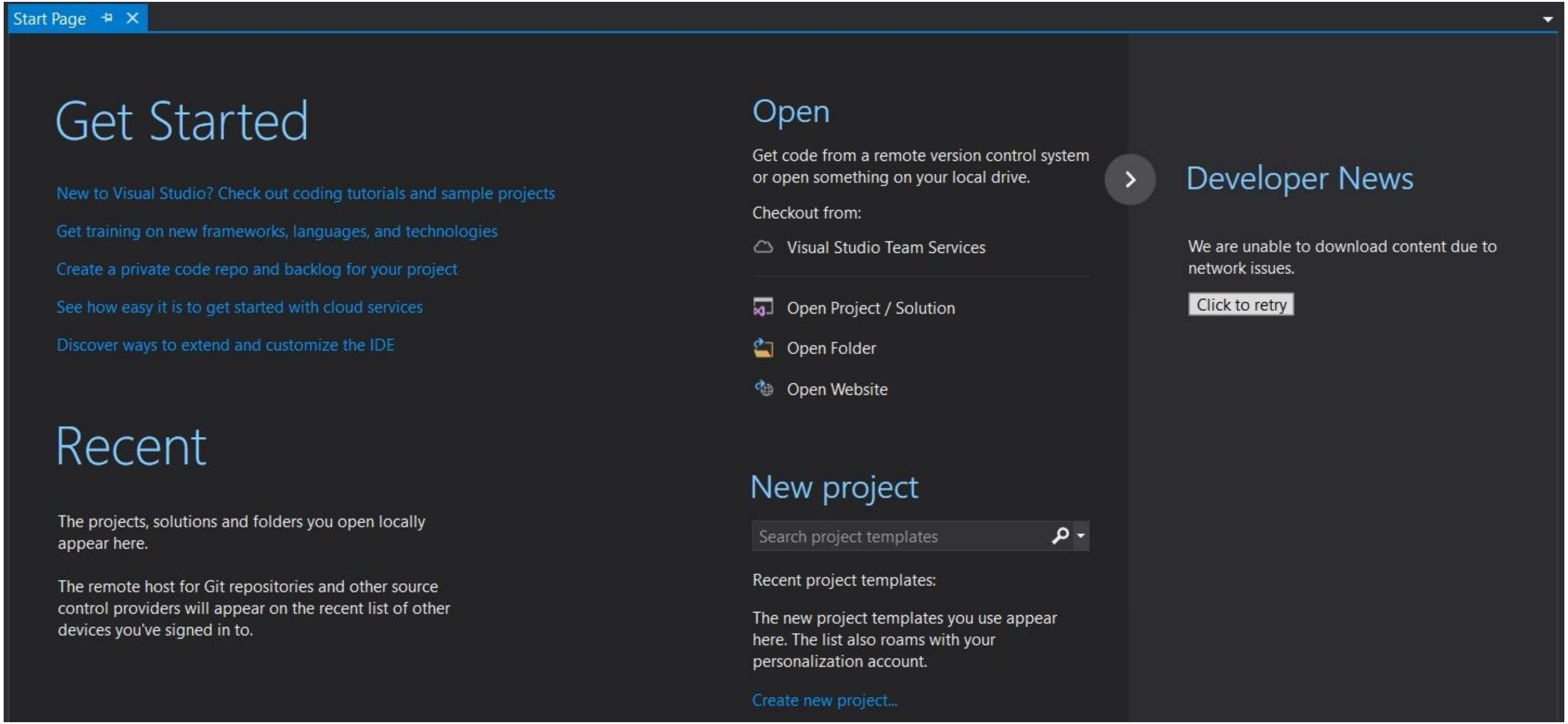
Code completion: ☑ Suggest completions while typing
☑ Use Escape key to show completion suggestions
☑ Automatically insert closing braces ("}")
☑ Enable type-over completions
☑ Automatically balance brackets in Objective-C method calls

While editing: ☑ Automatically trim trailing whitespace
☐ Including whitespace-only lines

Default text encoding: Unicode (UTF-8)
Default line endings: macOS / Unix (LF)
☐ Convert existing files on save

Code coverage: ☑ Show iteration counts

Demo: Microsoft Visual Studio



Get Started

New to Visual Studio? Check out coding tutorials and sample projects

Get training on new frameworks, languages, and technologies

Create a private code repo and backlog for your project

See how easy it is to get started with cloud services

Discover ways to extend and customize the IDE

Recent

The projects, solutions and folders you open locally appear here.

The remote host for Git repositories and other source control providers will appear on the recent list of other devices you've signed in to.

Open

Get code from a remote version control system or open something on your local drive.

Checkout from:

Visual Studio Team Services

Open Project / Solution

Open Folder

Open Website

New project

Search project templates

Recent project templates:

The new project templates you use appear here. The list also roams with your personalization account.

Create new project...

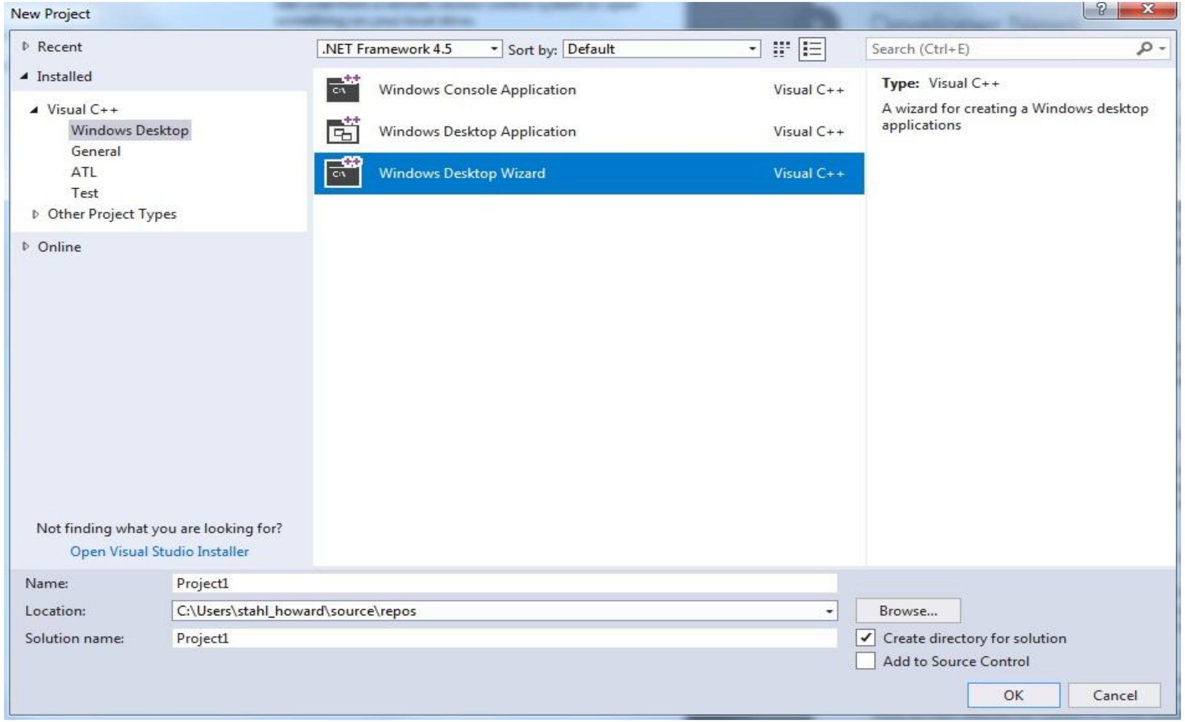
Developer News

We are unable to download content due to network issues.

Click to retry

Demo: Microsoft Visual Studio

Select File -> New -> Project. Open the Visual C++ group named Windows Desktop. Select the Windows Desktop Wizard



.NET Framework 4.5 Sort by: Default

Windows Console Application Visual C++

Windows Desktop Application Visual C++

Windows Desktop Wizard Visual C++

Type: Visual C++

A wizard for creating a Windows desktop applications

Not finding what you are looking for?

Open Visual Studio Installer

Name: Project1

Location: C:\Users\stahl_howard\source\repos

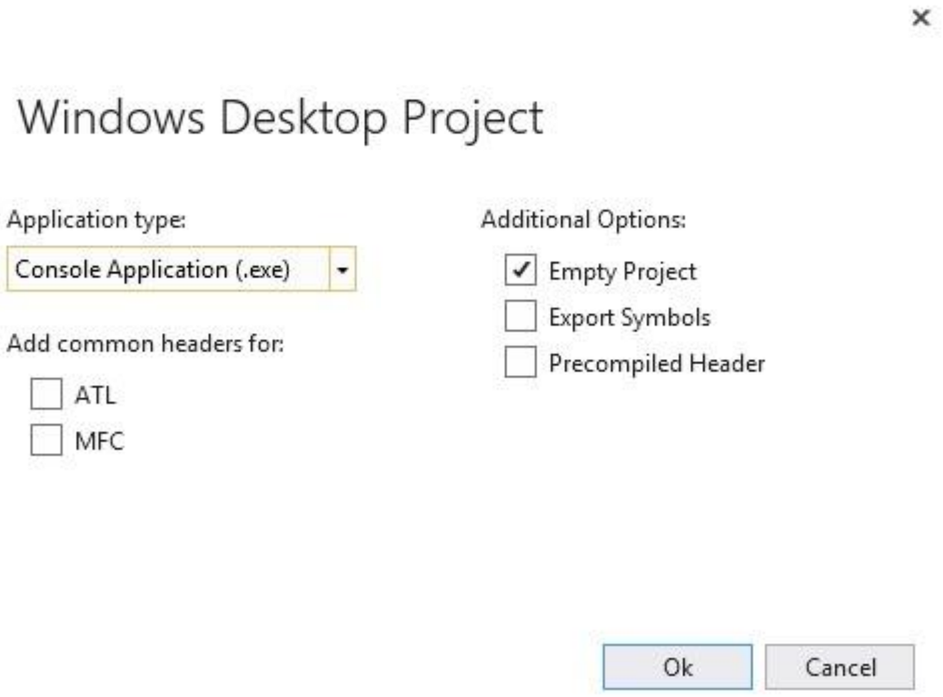
Solution name: Project1

☑ Create directory for solution

□ Add to Source Control

Demo: Microsoft Visual Studio

Click on the words "Empty Project" and click off "Precompiled Headers"



Windows Desktop Project

Application type:

Console Application (.exe)

Add common headers for:

□ ATL

□ MFC

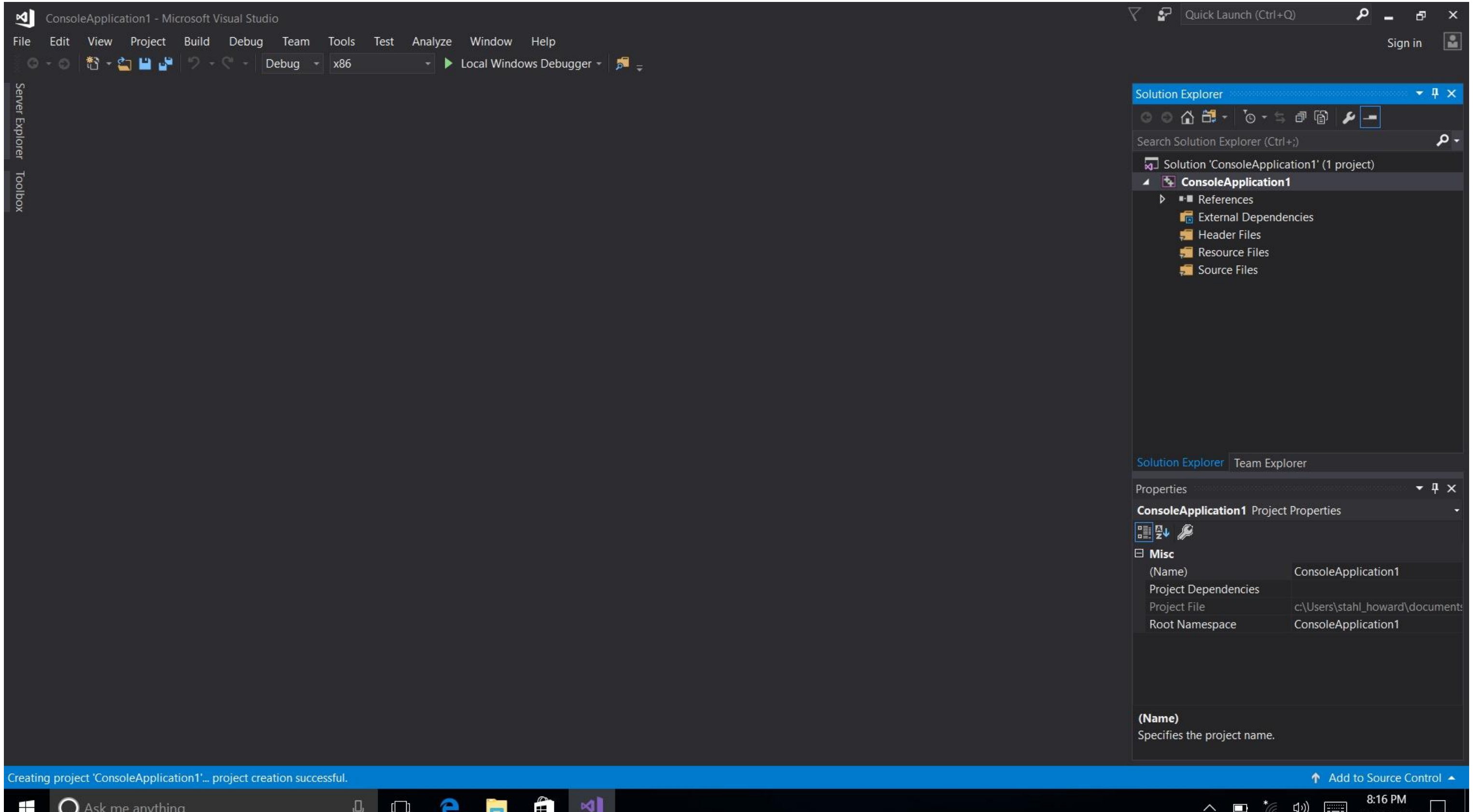
Additional Options:

☑ Empty Project

□ Export Symbols

□ Precompiled Header

Demo: Microsoft Visual Studio



ConsoleApplication1 - Microsoft Visual Studio

Solution 'ConsoleApplication1' (1 project)
- ConsoleApplication1
  - References
  - External Dependencies
  - Header Files
  - Resource Files
  - Source Files

Properties

ConsoleApplication1 Project Properties

Misc

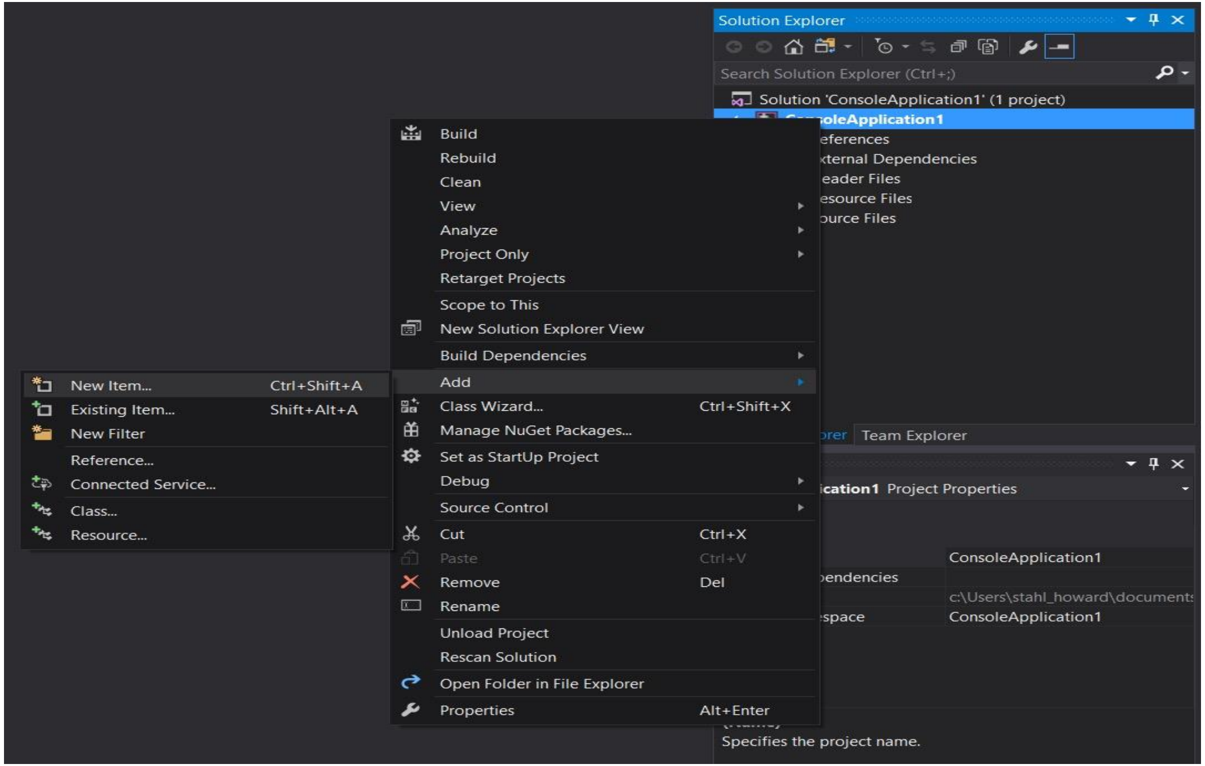
(Name)	ConsoleApplication1
Project Dependencies	
Project File	c:\Users\stahl_howard\document:
Root Namespace	ConsoleApplication1

(Name)
Specifies the project name.

Creating project 'ConsoleApplication1'... project creation successful.

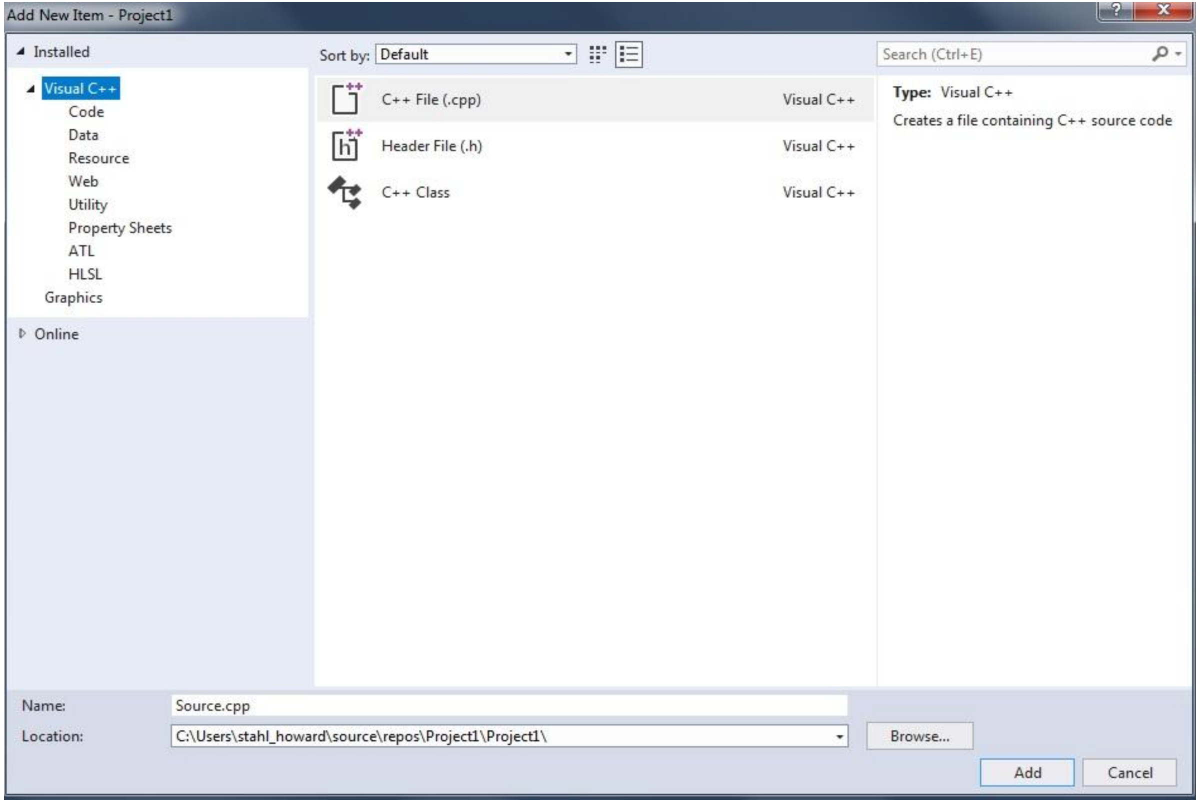
Demo: Microsoft Visual Studio

Add - > New Item



Demo: Microsoft Visual Studio

Select C++ File (.cpp)



Type: Visual C++

Creates a file containing C++ source code

Name: Source.cpp

Location: C:\Users\stahl_howard\source\repos\Project1\Project1\

Demo: Microsoft Visual Studio

Edit the .cpp file. And you may build and run it

Debug -> Start Without Debugging

Connecting to Linux server on Windows Machine

Go to http://www.seasnet.ucla.edu/how-to-log-into-terminal-server/

VPN: Cisco Client

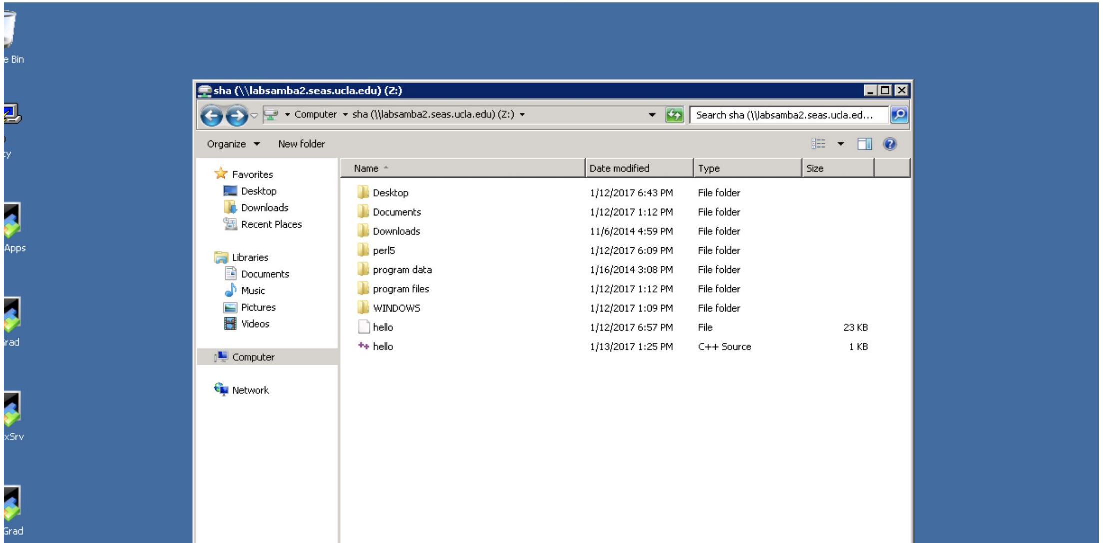
Another alternative:

Filezilla!

https://filezilla-project.org/

Demo: Linux with g++

- Step 1 for Windows users: Copy your C++ source file to the Windows desktop on a SEASnet machine.



sha (\\labsamba2.seas.ucla.edu) (Z:)

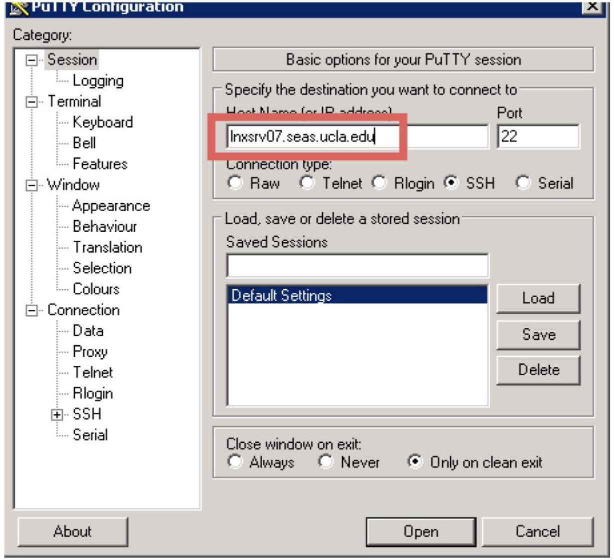
Computer ▸ sha (\\labsamba2.seas.ucla.edu) (Z:)

Name	Date modified	Type	Size
Desktop	1/12/2017 6:43 PM	File folder	
Documents	1/12/2017 1:12 PM	File folder	
Downloads	11/6/2014 4:59 PM	File folder	
perl5	1/12/2017 6:09 PM	File folder	
program data	1/16/2014 3:08 PM	File folder	
program files	1/12/2017 1:12 PM	File folder	
WINDOWS	1/12/2017 1:09 PM	File folder	
hello	1/12/2017 6:57 PM	File	23 KB
hello	1/13/2017 1:25 PM	C++ Source	1 KB

Connecting to Linux server on Windows Machine

Demo: Linux with g++

- Step 2 for Windows users: Configure and use putty
- Change Hosst Name to: lnxsrv07.seas.ucla.edu
- Click Open. Then put in your Seasnet username and password



Specify the destination you want to connect to

Host Name (or IP address): lnxsrv07.seas.ucla.edu

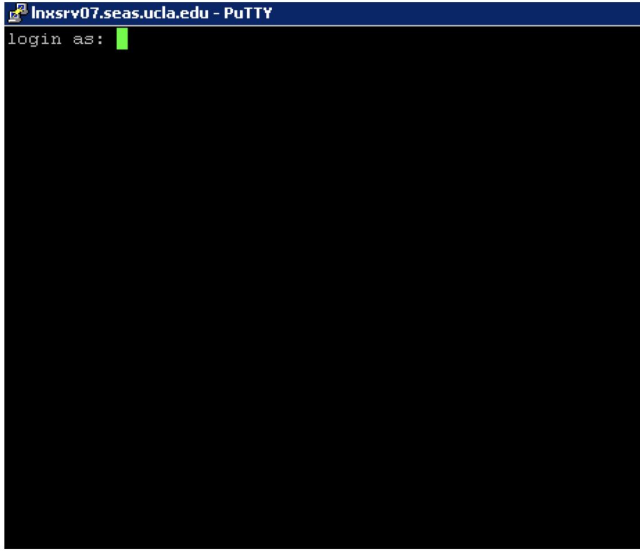
Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions: Default Settings

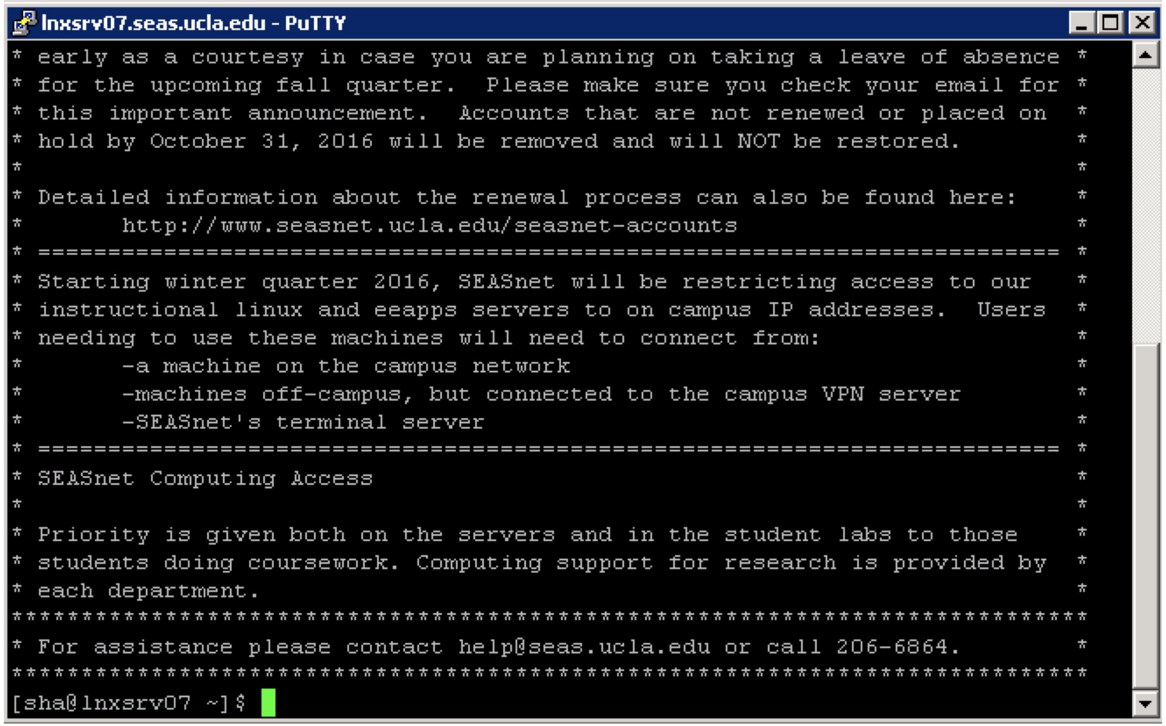
Close window on exit: ○ Always ○ Never ● Only on clean exit



```
login as: 
```

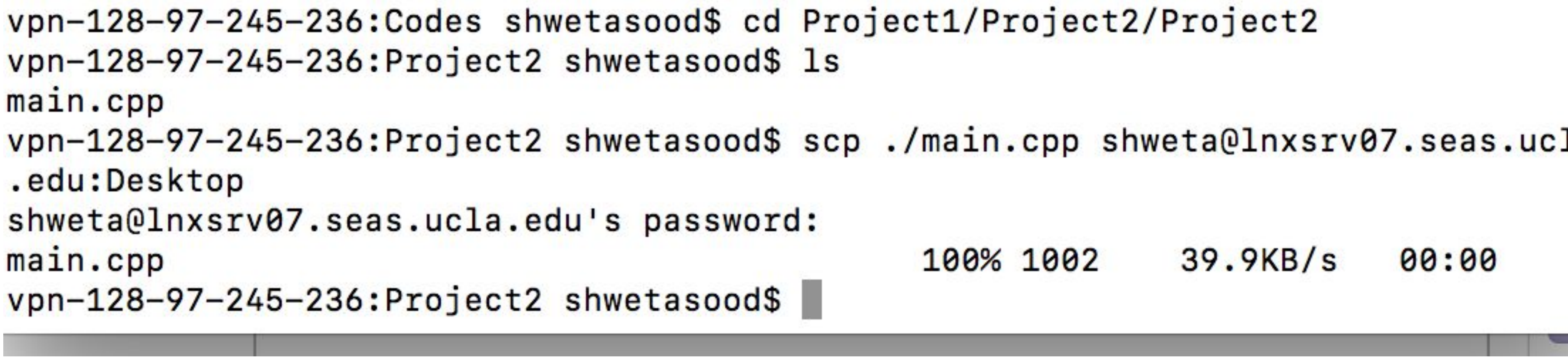
Connecting to Linux server on Windows Machine

Demo: Linux with g++



```
* early as a courtesy in case you are planning on taking a leave of absence *
* for the upcoming fall quarter.  Please make sure you check your email for *
* this important announcement.  Accounts that are not renewed or placed on  *
* hold by October 31, 2016 will be removed and will NOT be restored.        *
*                                                                           *
* Detailed information about the renewal process can also be found here:    *
*       http://www.seasnet.ucla.edu/seasnet-accounts                        *
* ========================================================================= *
* Starting winter quarter 2016, SEASnet will be restricting access to our   *
* instructional linux and eeapps servers to on campus IP addresses.  Users  *
* needing to use these machines will need to connect from:                  *
*       -a machine on the campus network                                    *
*       -machines off-campus, but connected to the campus VPN server        *
*       -SEASnet's terminal server                                          *
* ========================================================================= *
* SEASnet Computing Access                                                  *
*                                                                           *
* Priority is given both on the servers and in the student labs to those    *
* students doing coursework. Computing support for research is provided by  *
* each department.                                                          *
*****************************************************************************
* For assistance please contact help@seas.ucla.edu or call 206-6864.        *
*****************************************************************************
[sha@lnxsrv07 ~]$ 
```

Connecting to Linux server on Mac

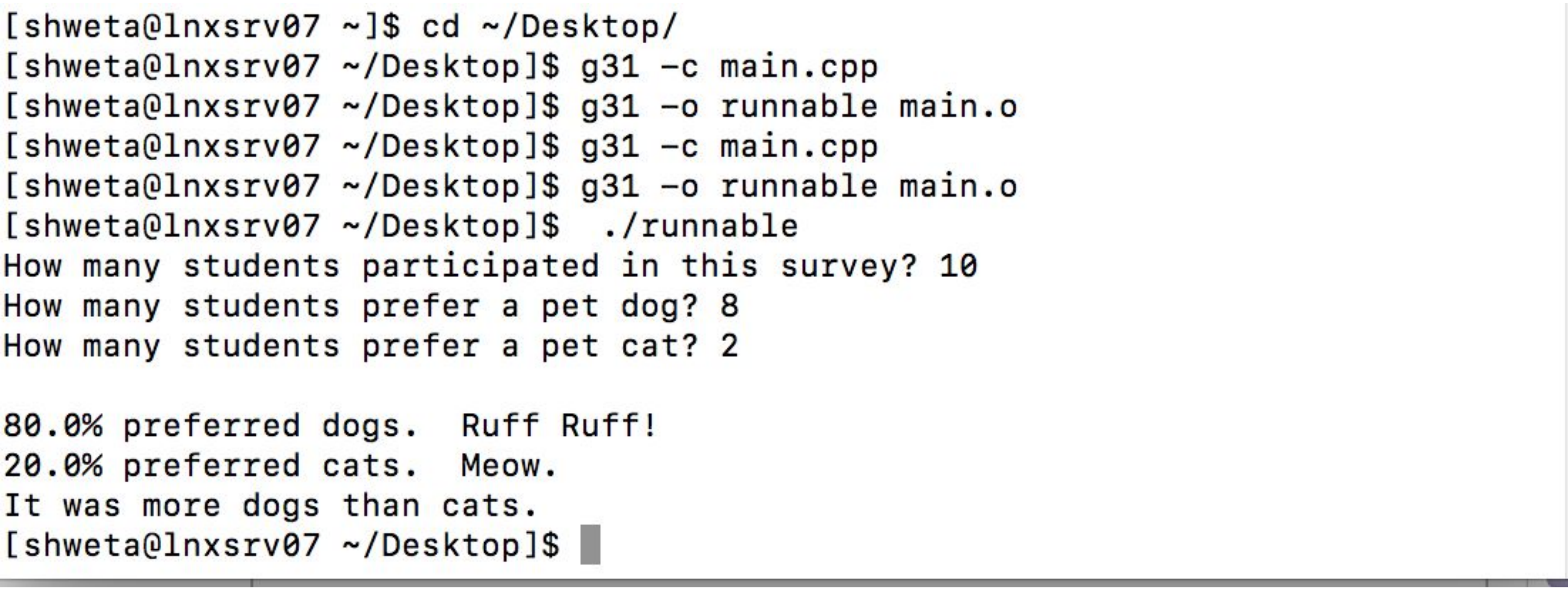


```
vpn-128-97-245-236:Codes shwetasood$ cd Project1/Project2/Project2
vpn-128-97-245-236:Project2 shwetasood$ ls
main.cpp
vpn-128-97-245-236:Project2 shwetasood$ scp ./main.cpp shweta@lnxsrv07.seas.ucl
.edu:Desktop
shweta@lnxsrv07.seas.ucla.edu's password:
main.cpp                                   100% 1002     39.9KB/s   00:00
vpn-128-97-245-236:Project2 shwetasood$ 
```

VPN: Cisco Client
Type these on terminal:

ssh yourSEASaccount@lnxsrv07.seas.ucla.edu
1 time set up command: curl -s -L http://cs.ucla.edu/classes/winter18/cs31/Utilities/setupg31 | bash
g31 -c main.cpp
g31 -o runnable main.o
./runnable

Connecting to Linux server on Mac



```
[shweta@lnxsrv07 ~]$ cd ~/Desktop/
[shweta@lnxsrv07 ~/Desktop]$ g31 -c main.cpp
[shweta@lnxsrv07 ~/Desktop]$ g31 -o runnable main.o
[shweta@lnxsrv07 ~/Desktop]$ g31 -c main.cpp
[shweta@lnxsrv07 ~/Desktop]$ g31 -o runnable main.o
[shweta@lnxsrv07 ~/Desktop]$  ./runnable
How many students participated in this survey? 10
How many students prefer a pet dog? 8
How many students prefer a pet cat? 2

80.0% preferred dogs.  Ruff Ruff!
20.0% preferred cats.  Meow.
It was more dogs than cats.
[shweta@lnxsrv07 ~/Desktop]$ 
```

Errors

- Compile / Syntax Error
  - Prevent your program to compile
  - e.g. missing a “;” at the end of line of statement
- Logic / Runtime / Semantic Error
  - Your program can compile (there might be warnings, but still it can compile successfully), but does not do what it suppose to do.

Thank You!

Project 1

```
cout.setf(ios::fixed)
```

makes cout print floats with a fixed number of decimals and

```
cout.precision(3)
```

sets this number to be three.

- **using**: You are going to use it
- **namespace**: To use what? A namespace
- **std**: The `std` namespace (where features of the C++ Standard Library, such as `string` or `vector`, are declared).

After you write this instruction, if the compiler sees `string` it will know that you may be referring to `std::string`, and if it sees `vector`, it will know that you may be referring to `std::vector`. (Provided that you have included in your compilation unit the header files where they are defined, of course.)

If you *don't* write it, when the compiler sees `string` or `vector` it will not know what you are refering to. You will need to explicitly tell it `std::string` or `std::vector`, and if you don't, you will get a compile error.