Version Control with 

Integration within common IDEs

Lecturer: Prof. Dr. Steffen ROTHKUGEL
Teaching Assistants: Dr. Jean BOTEV, Christian GRÉVISSE, Johannes KLEIN

UNIVERSITÉ DU LUXEMBOURG

Disclaimer: Scope of this document

These slides show the basic integration of the Git version control system in common IDEs, such as Eclipse, Xcode, Netbeans and Visual Studio. This guide only shows how to push a first version of a project to a remote repository, pull the current version of a project and clone an existing repository.

By no means shall this document be understood as a complete guide, as many fundamental topics such as branching and conflict resolution are not addressed here.

For further information, please refer to the official Git documentation, the respective IDE help sections and, of course, the Moodle section of your course.

Advantages of Version Control

Keeping track of modifications between project versions

Maintenance & Management

Sharing & Collaboration

Backup & Recovery

Use within our courses

BINFO

- Programming 2
- Algorithms 3
- Distributed Systems

MICS

- Mobile Computing

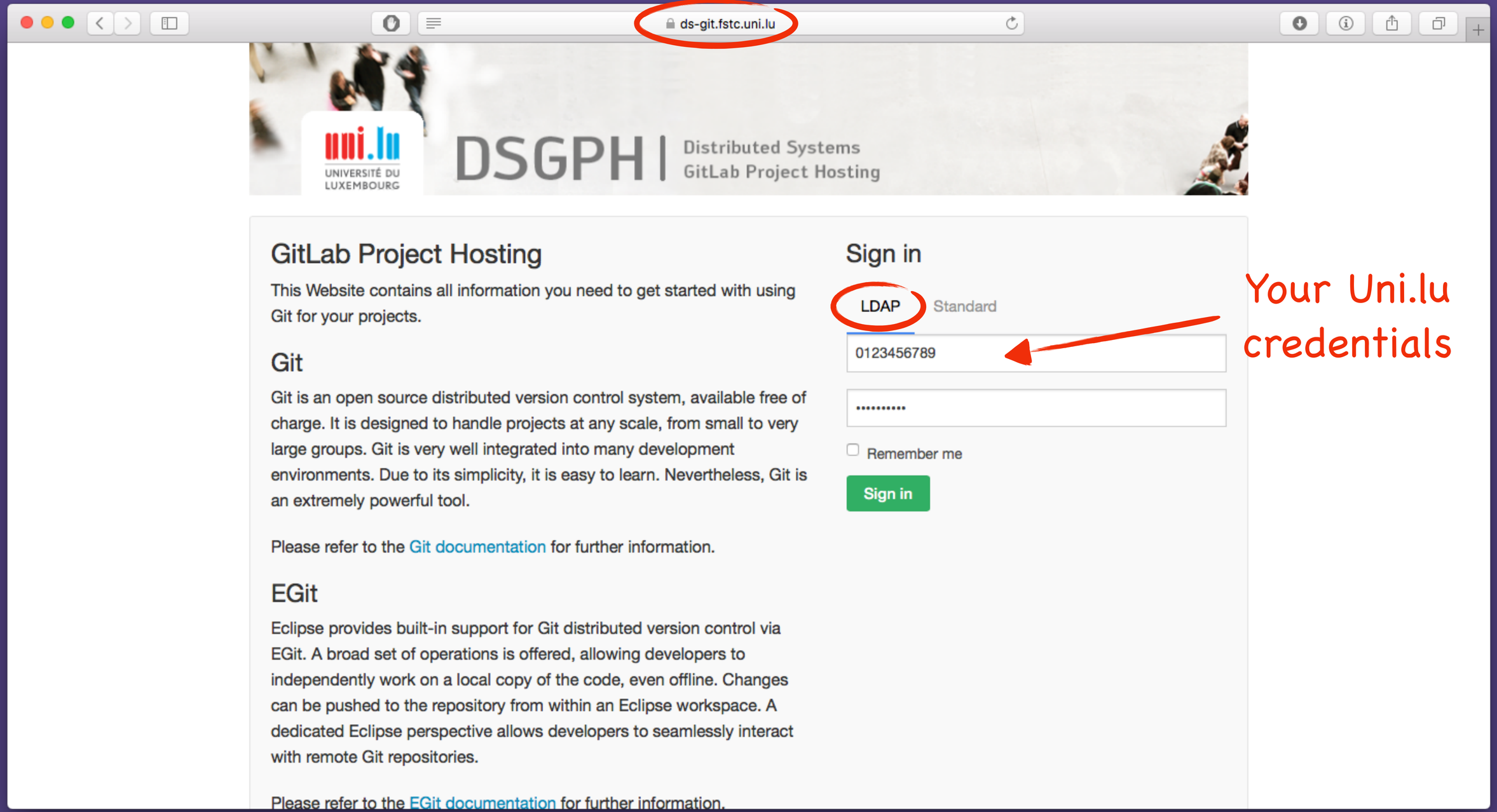
GitLab

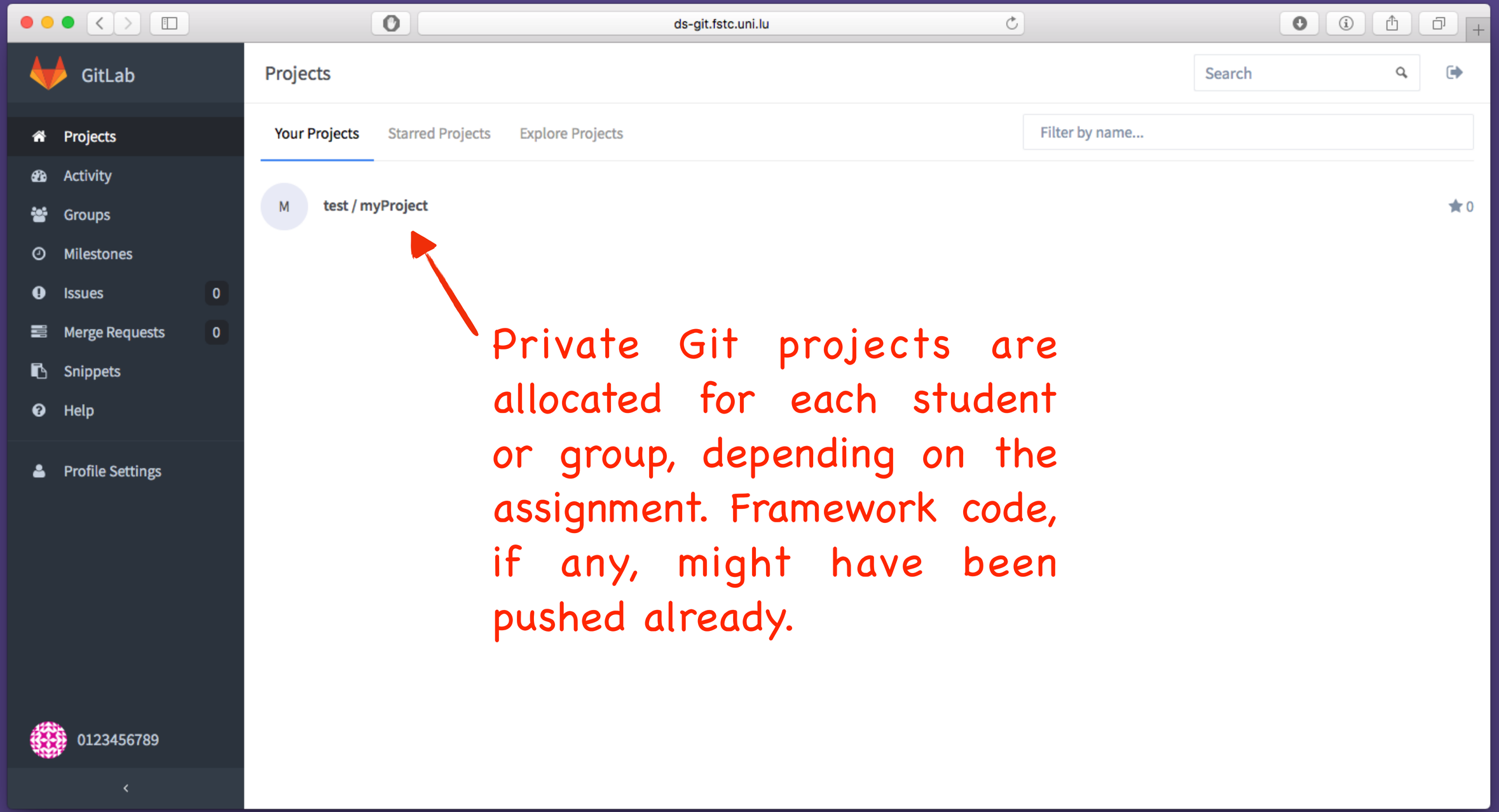
GitLab

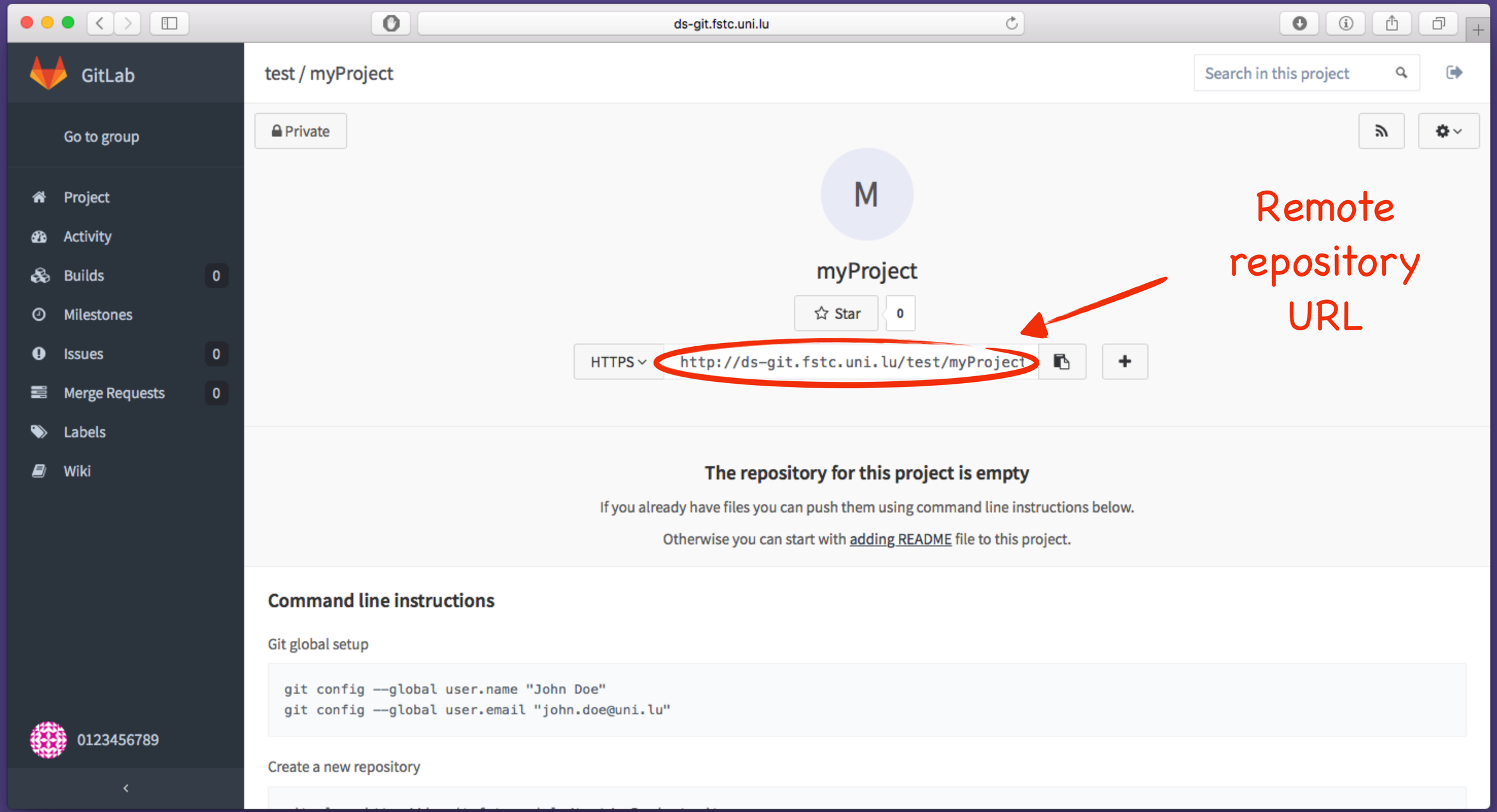
Web-based Git repository manager

Installed on uni.lu server (DS-GIT)

- https://ds-git.fstc.uni.lu


ds-git.fstc.uni.lu
UNIVERSITÉ DU LUXEMBOURG
DSGPH | Distributed Systems GitLab Project Hosting
GitLab Project Hosting
This Website contains all information you need to get started with using Git for your projects.
Git
Git is an open source distributed version control system, available free of charge. It is designed to handle projects at any scale, from small to very large groups. Git is very well integrated into many development environments. Due to its simplicity, it is easy to learn. Nevertheless, Git is an extremely powerful tool.
Please refer to the Git documentation for further information.
EGit
Eclipse provides built-in support for Git distributed version control via EGit. A broad set of operations is offered, allowing developers to independently work on a local copy of the code, even offline. Changes can be pushed to the repository from within an Eclipse workspace. A dedicated Eclipse perspective allows developers to seamlessly interact with remote Git repositories.
Please refer to the EGit documentation for further information.
Sign in
LDAP
Standard
0123456789
Remember me
Sign in
Your Uni.lu credentials

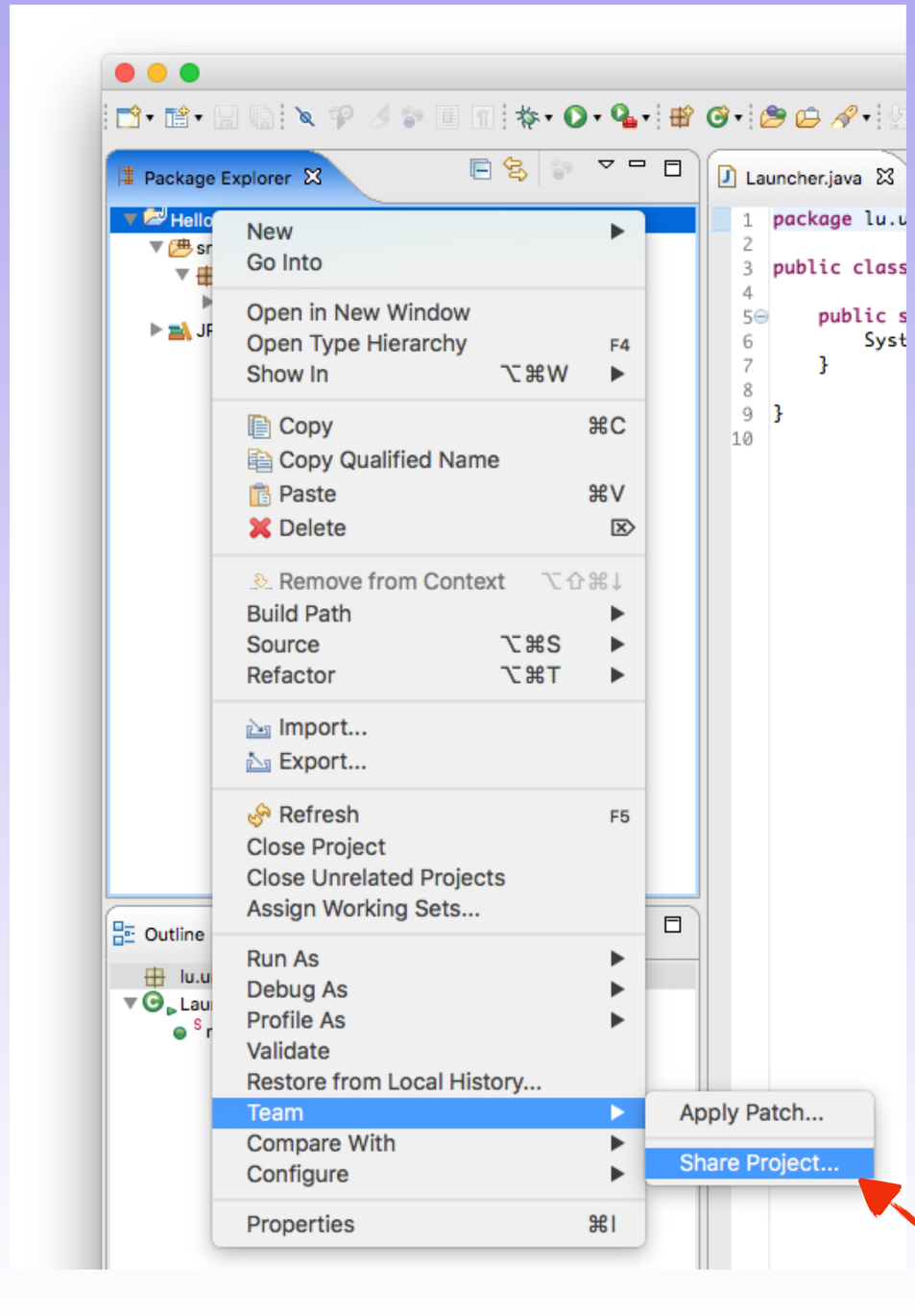

ds-git.fstc.uni.lu
GitLab
Projects
Activity
Groups
Milestones
Issues 0
Merge Requests 0
Snippets
Help
Profile Settings
0123456789
Projects
Search
Your Projects
Starred Projects
Explore Projects
Filter by name...
M
test / myProject
0
Private Git projects are allocated for each student or group, depending on the assignment. Framework code, if any, might have been pushed already.

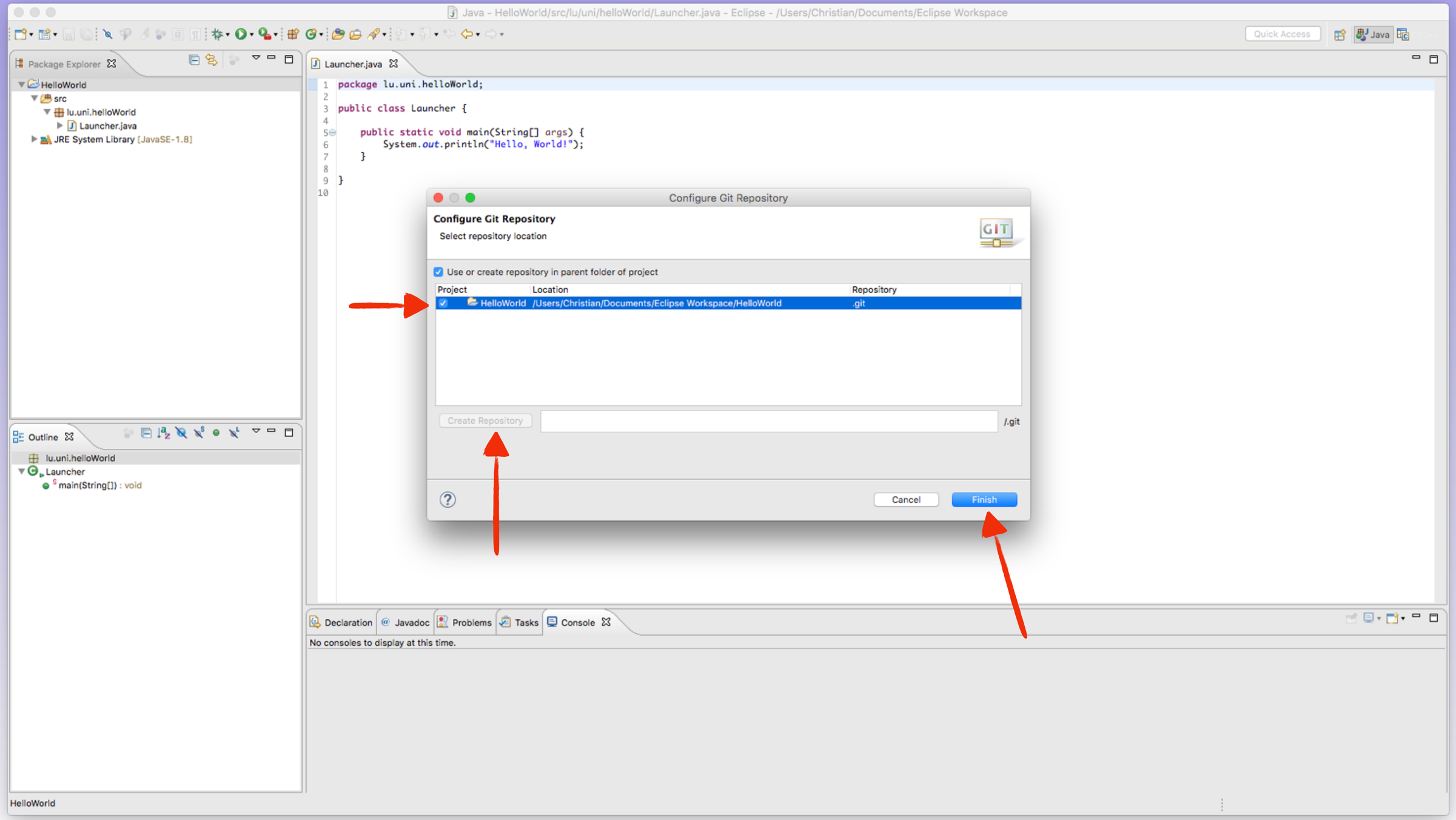

ds-git.fstc.uni.lu
GitLab
test / myProject
Search in this project
Go to group
Private
Project
Activity
Builds 0
Milestones
Issues 0
Merge Requests 0
Labels
Wiki
M
myProject
Star 0
HTTPS
http://ds-git.fstc.uni.lu/test/myProject
Remote repository URL
The repository for this project is empty
If you already have files you can push them using command line instructions below.
Otherwise you can start with adding README file to this project.
Command line instructions
Git global setup
git config --global user.name "John Doe"
git config --global user.email "john.doe@uni.lu"
Create a new repository
0123456789

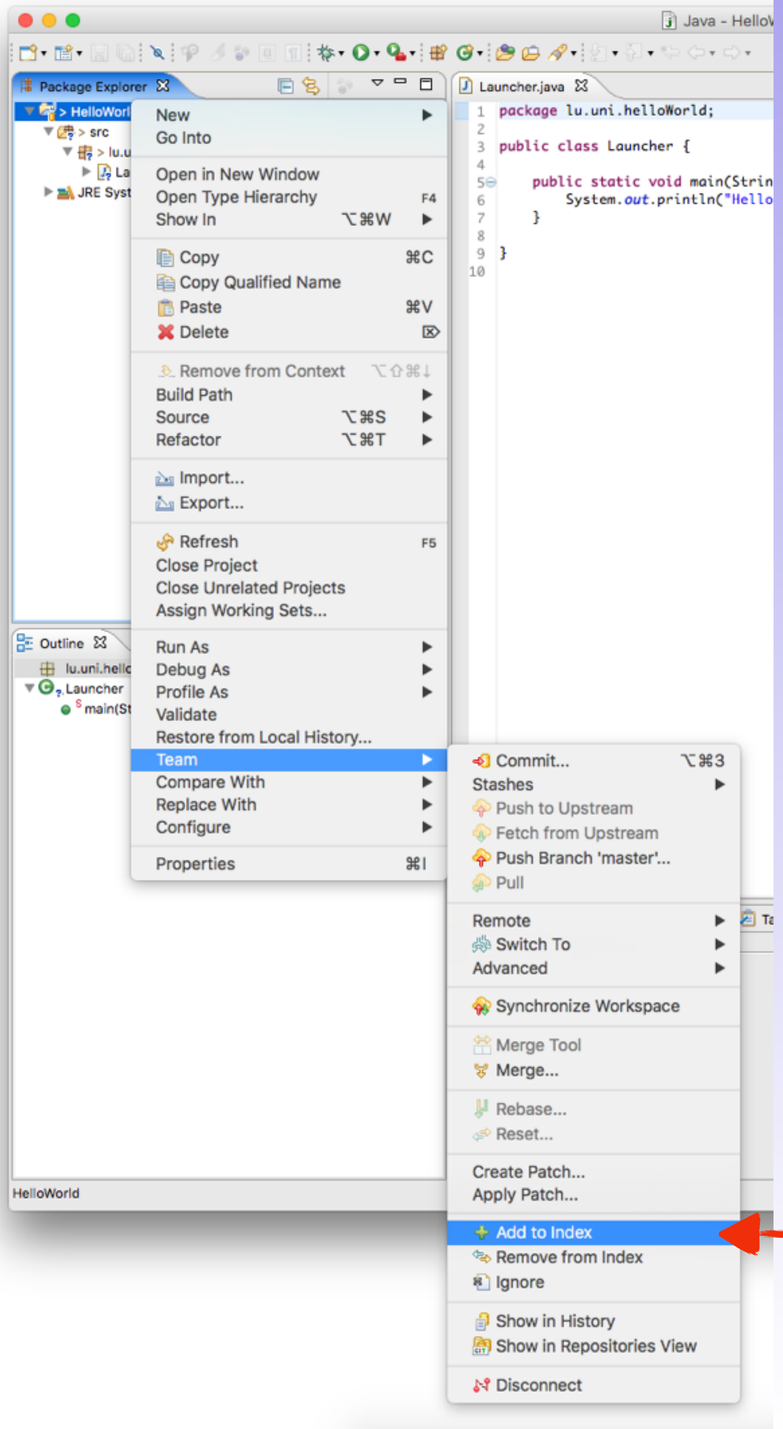
IDE-specific Git Integration

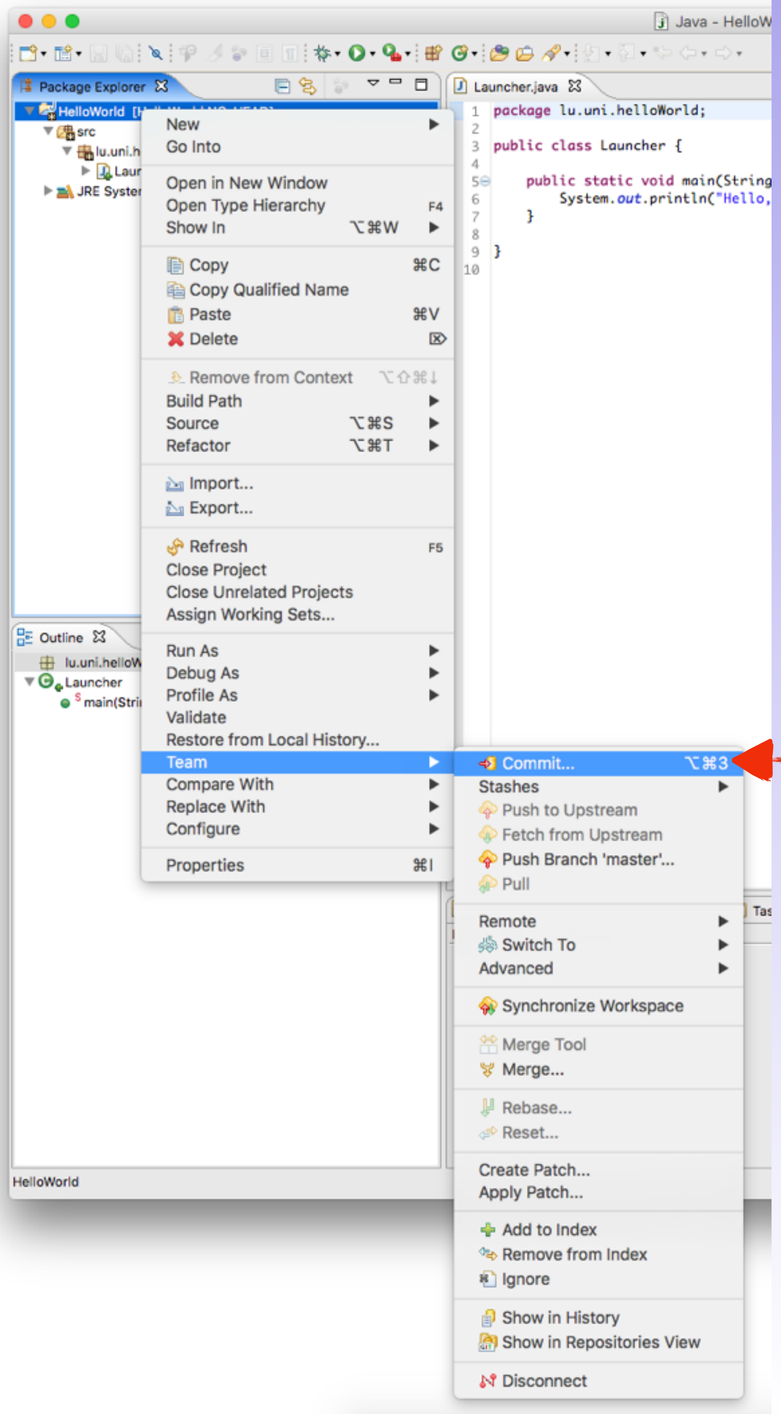
Eclipse

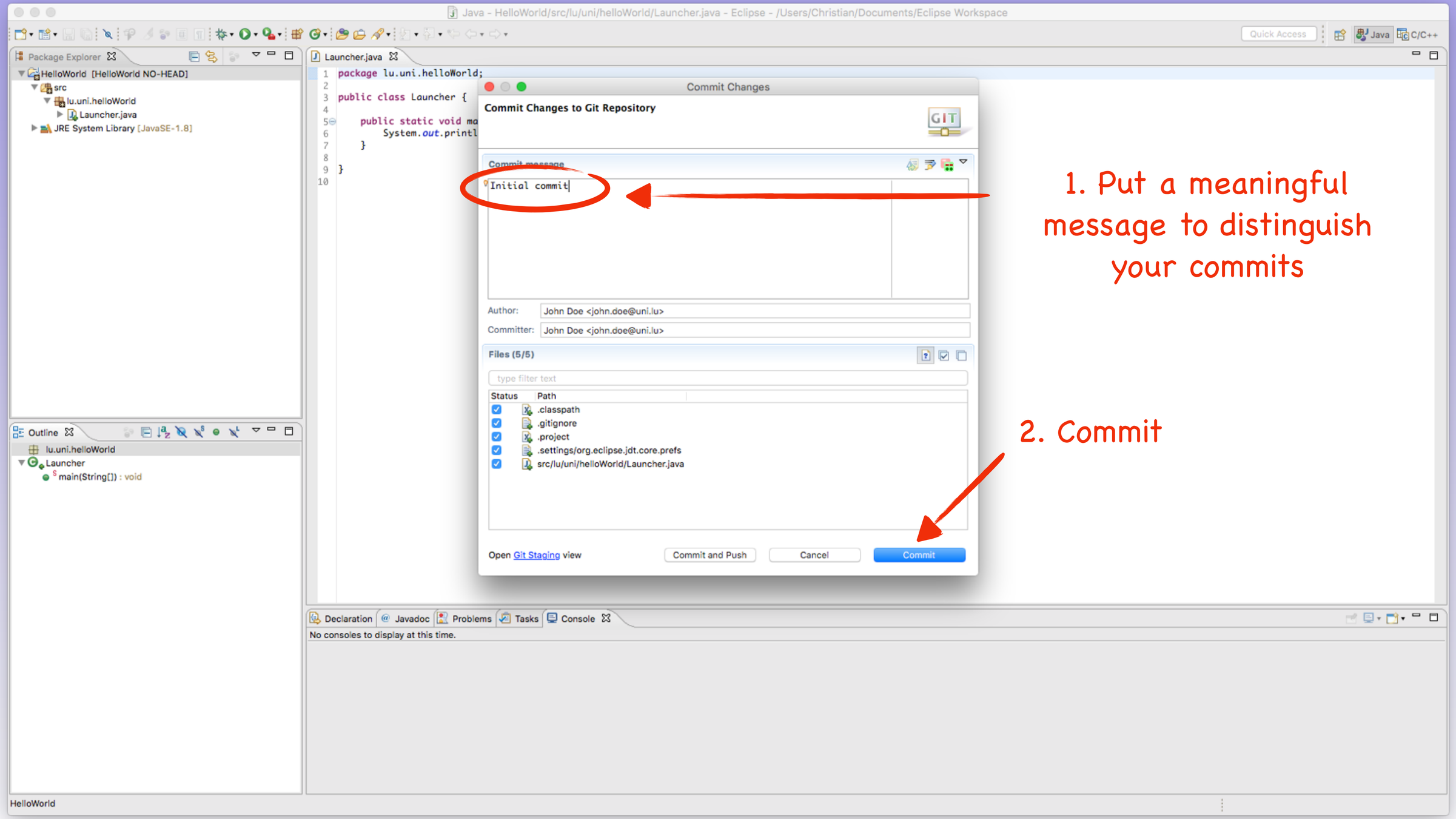
Initial repository setup

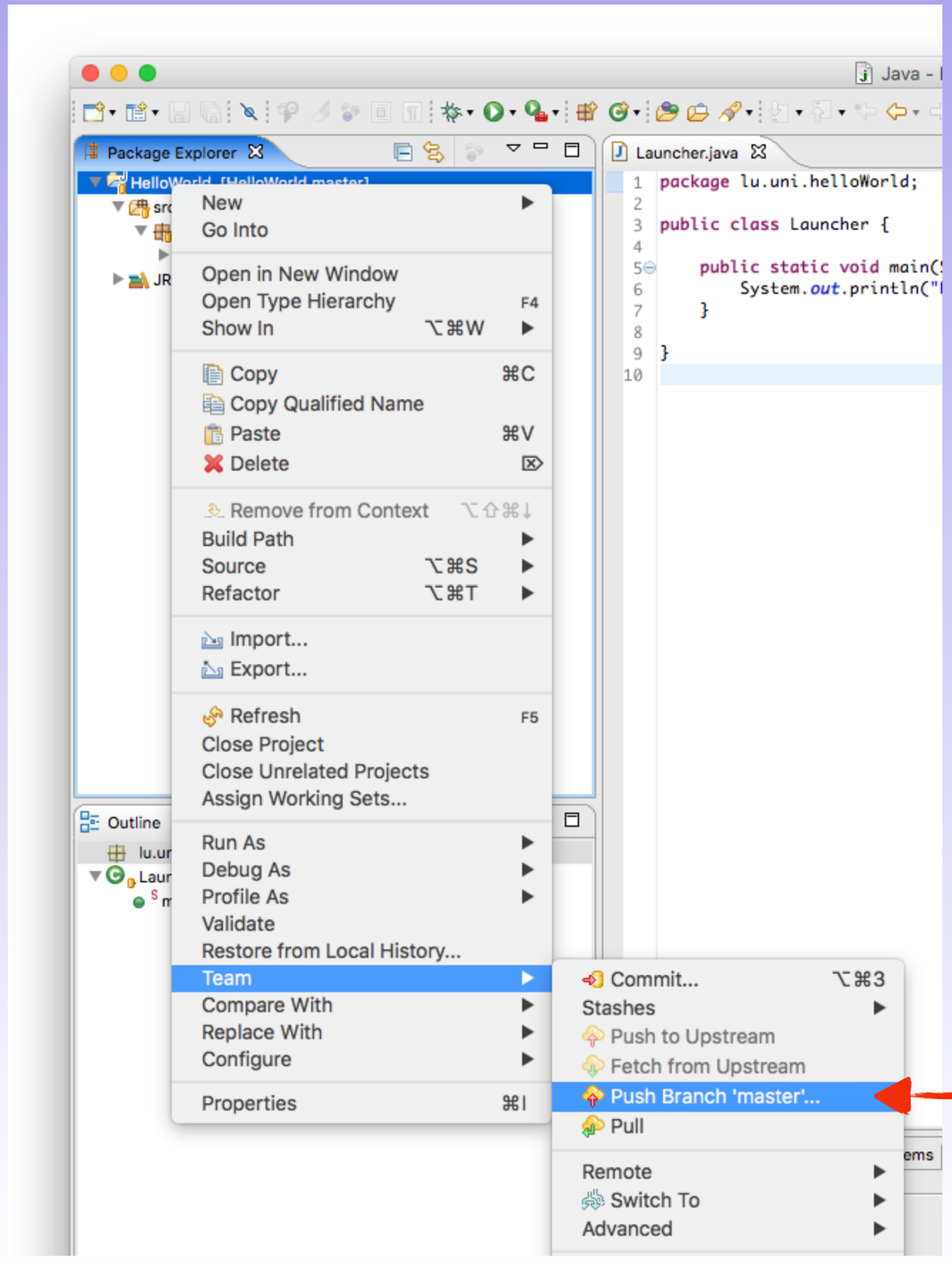


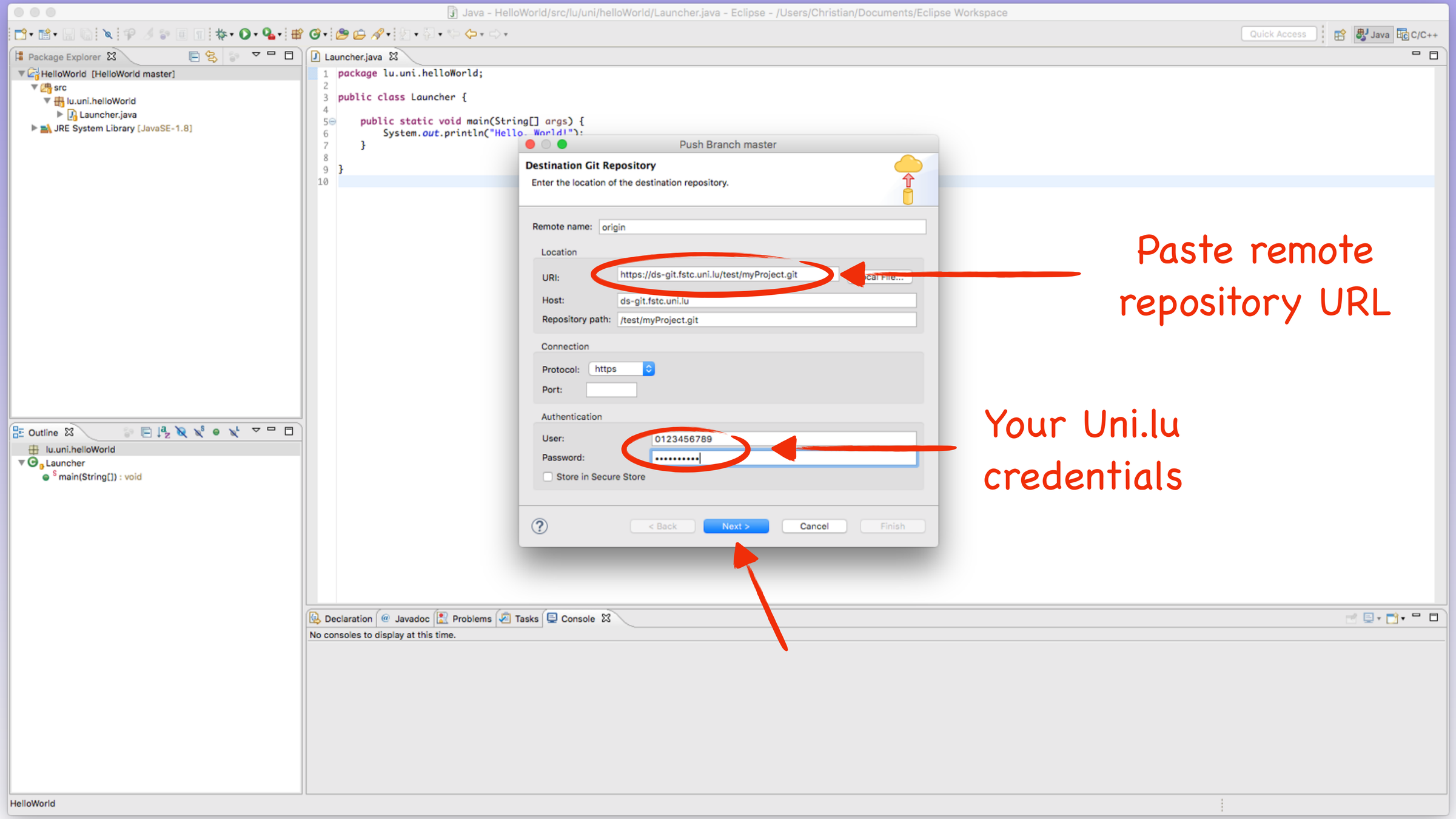

Java - HelloWorld/src/lu/uni/helloWorld/Launcher.java - Eclipse - /Users/Christian/Documents/Eclipse Workspace
Quick Access
Java
Package Explorer
HelloWorld
src
lu.uni.helloWorld
Launcher.java
JRE System Library [JavaSE-1.8]
Launcher.java
1 package lu.uni.helloWorld;
2
3 public class Launcher {
4
5 public static void main(String[] args) {
6 System.out.println("Hello, World!");
7 }
8
9 }
10
Configure Git Repository
Configure Git Repository
Select repository location
GIT
Use or create repository in parent folder of project
Project
Location
Repository
HelloWorld
/Users/Christian/Documents/Eclipse Workspace/HelloWorld
.git
Create Repository
/.git
Cancel
Finish
Outline
lu.uni.helloWorld
Launcher
main(String[]) : void
Declaration
Javadoc
Problems
Tasks
Console
No consoles to display at this time.
HelloWorld

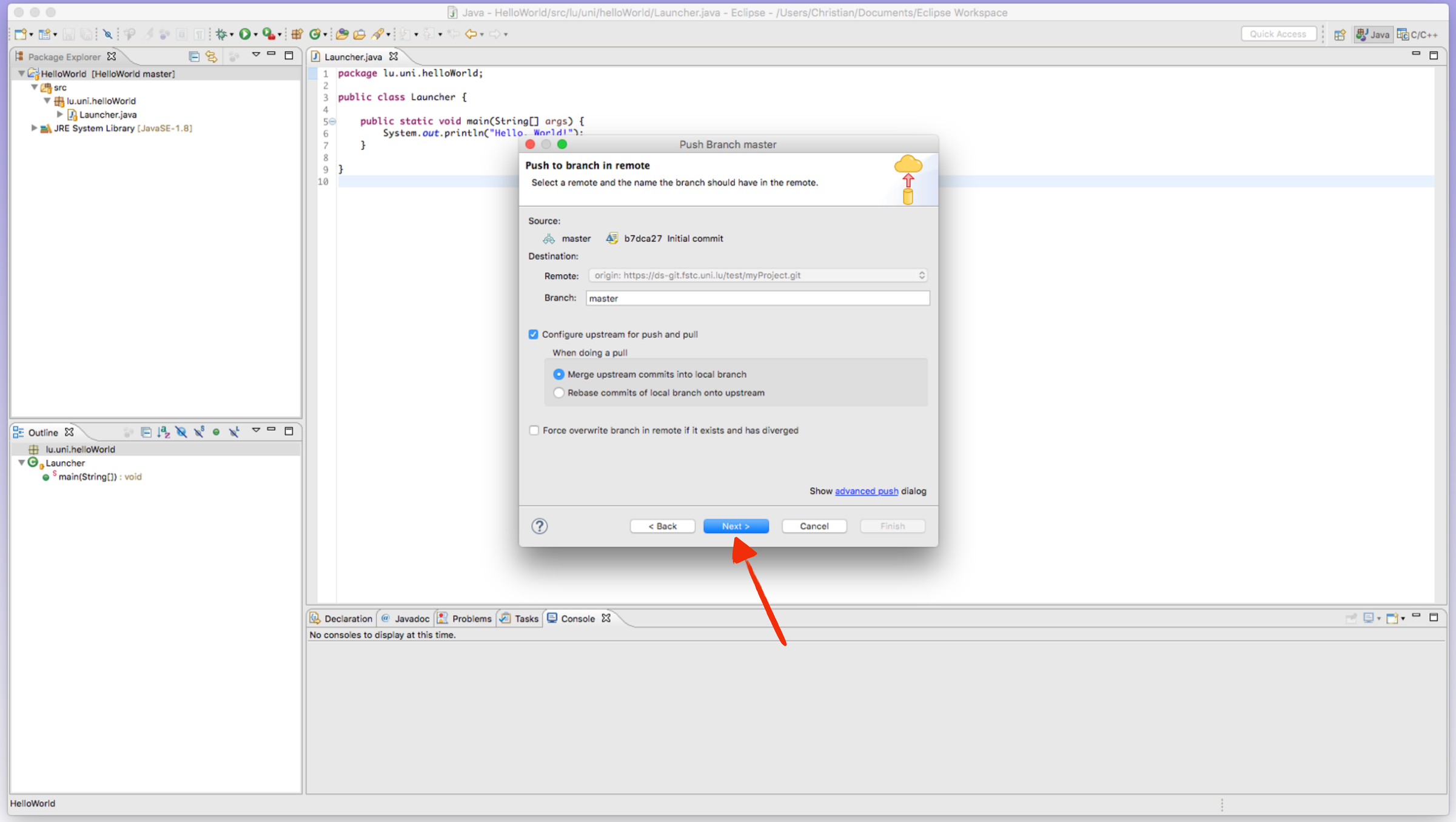

Java - HelloW
Package Explorer
HelloWorl
src
lu.u
La
JRE Syst
Launcher.java
1 package lu.uni.helloWorld;
2
3 public class Launcher {
4
5 public static void main(Strin
6 System.out.println("Hello
7 }
8
9 }
10
Outline
lu.uni.hello
Launcher
main(St
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In ⌥⌘W
Copy ⌘C
Copy Qualified Name
Paste ⌘V
Delete
Remove from Context ⌥⇧⌘↓
Build Path
Source ⌥⌘S
Refactor ⌥⌘T
Import...
Export...
Refresh F5
Close Project
Close Unrelated Projects
Assign Working Sets...
Run As
Debug As
Profile As
Validate
Restore from Local History...
Team
Compare With
Replace With
Configure
Properties ⌘I
Commit... ⌥⌘3
Stashes
Push to Upstream
Fetch from Upstream
Push Branch 'master'...
Pull
Remote
Switch To
Advanced
Synchronize Workspace
Merge Tool
Merge...
Rebase...
Reset...
Create Patch...
Apply Patch...
Add to Index
Remove from Index
Ignore
Show in History
Show in Repositories View
Disconnect
HelloWorld

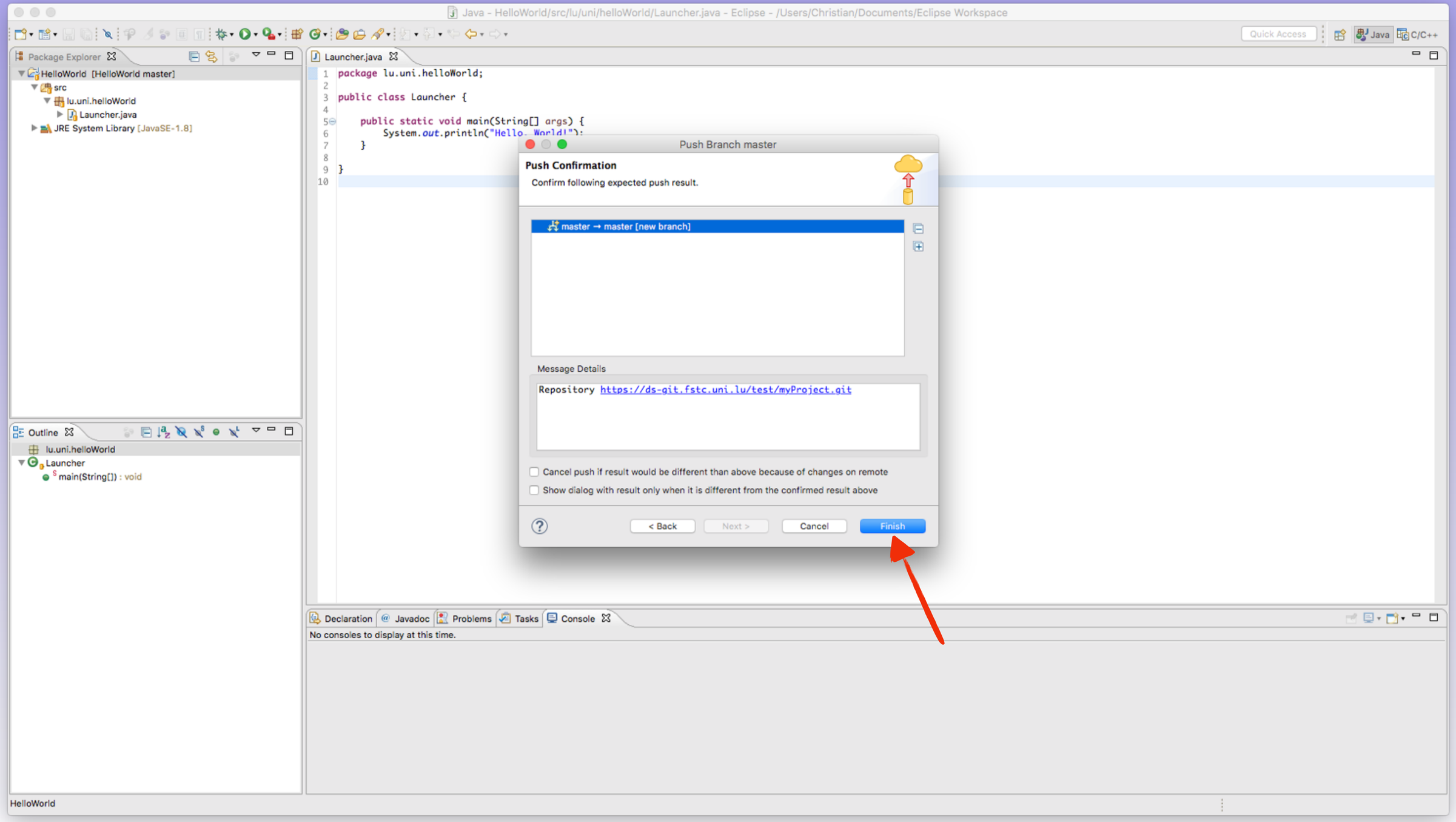

Java - HelloW
Package Explorer
HelloWorld
src
JRE Syste
Launcher.java
1 package lu.uni.helloWorld;
2
3 public class Launcher {
4
5 public static void main(String
6 System.out.println("Hello,
7 }
8
9 }
10
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In ⌥⌘W
Copy ⌘C
Copy Qualified Name
Paste ⌘V
Delete
Remove from Context ⌥⇧⌘↓
Build Path
Source ⌥⌘S
Refactor ⌥⌘T
Import...
Export...
Refresh F5
Close Project
Close Unrelated Projects
Assign Working Sets...
Run As
Debug As
Profile As
Validate
Restore from Local History...
Team
Compare With
Replace With
Configure
Properties ⌘I
Outline
lu.uni.helloW
Launcher
main(Stri
Commit... ⌥⌘3
Stashes
Push to Upstream
Fetch from Upstream
Push Branch 'master'...
Pull
Remote
Switch To
Advanced
Synchronize Workspace
Merge Tool
Merge...
Rebase...
Reset...
Create Patch...
Apply Patch...
Add to Index
Remove from Index
Ignore
Show in History
Show in Repositories View
Disconnect
HelloWorld

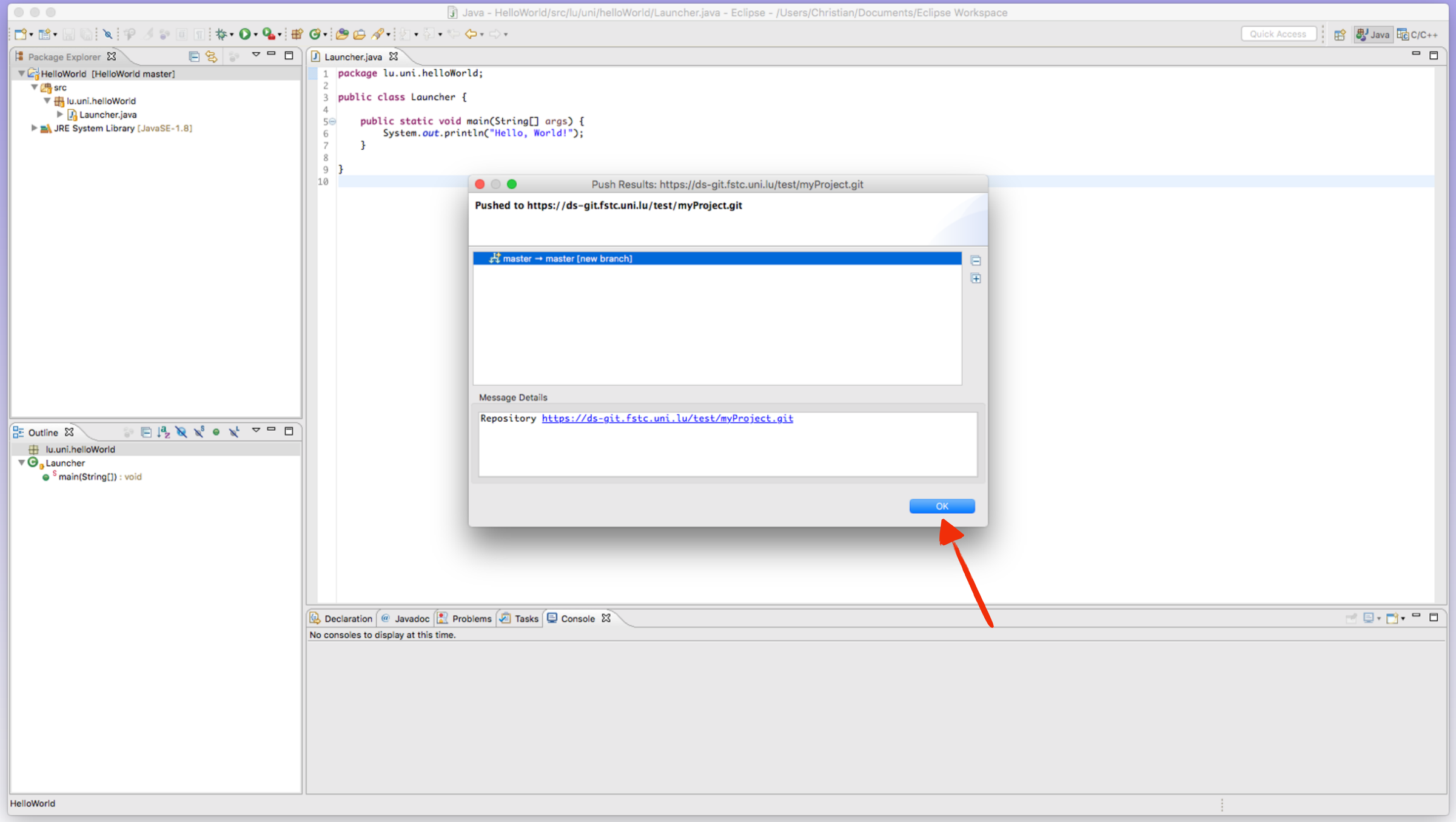

Java - HelloWorld/src/lu/uni/helloWorld/Launcher.java - Eclipse - /Users/Christian/Documents/Eclipse Workspace
Package Explorer
HelloWorld [HelloWorld NO-HEAD]
src
lu.uni.helloWorld
Launcher.java
JRE System Library [JavaSE-1.8]
Launcher.java
package lu.uni.helloWorld;
public class Launcher {
public static void ma
System.out.printl
}
}
Outline
lu.uni.helloWorld
Launcher
main(String[]) : void
Commit Changes
Commit Changes to Git Repository
Commit message
Initial commit
Author: John Doe <john.doe@uni.lu>
Committer: John Doe <john.doe@uni.lu>
Files (5/5)
type filter text
Status Path
.classpath
.gitignore
.project
.settings/org.eclipse.jdt.core.prefs
src/lu/uni/helloWorld/Launcher.java
Open Git Staging view
Commit and Push
Cancel
Commit
1. Put a meaningful message to distinguish your commits
2. Commit
Declaration
Javadoc
Problems
Tasks
Console
No consoles to display at this time.
HelloWorld


Java -
Package Explorer
Launcher.java
1 package lu.uni.helloWorld;
2
3 public class Launcher {
4
5 public static void main(
6 System.out.println("
7 }
8
9 }
10
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In ⌥⌘W
Copy ⌘C
Copy Qualified Name
Paste ⌘V
Delete
Remove from Context ⌥⇧⌘↓
Build Path
Source ⌥⌘S
Refactor ⌥⌘T
Import...
Export...
Refresh F5
Close Project
Close Unrelated Projects
Assign Working Sets...
Run As
Debug As
Profile As
Validate
Restore from Local History...
Team
Compare With
Replace With
Configure
Properties ⌘I
Commit... ⌥⌘3
Stashes
Push to Upstream
Fetch from Upstream
Push Branch 'master'...
Pull
Remote
Switch To
Advanced
Outline

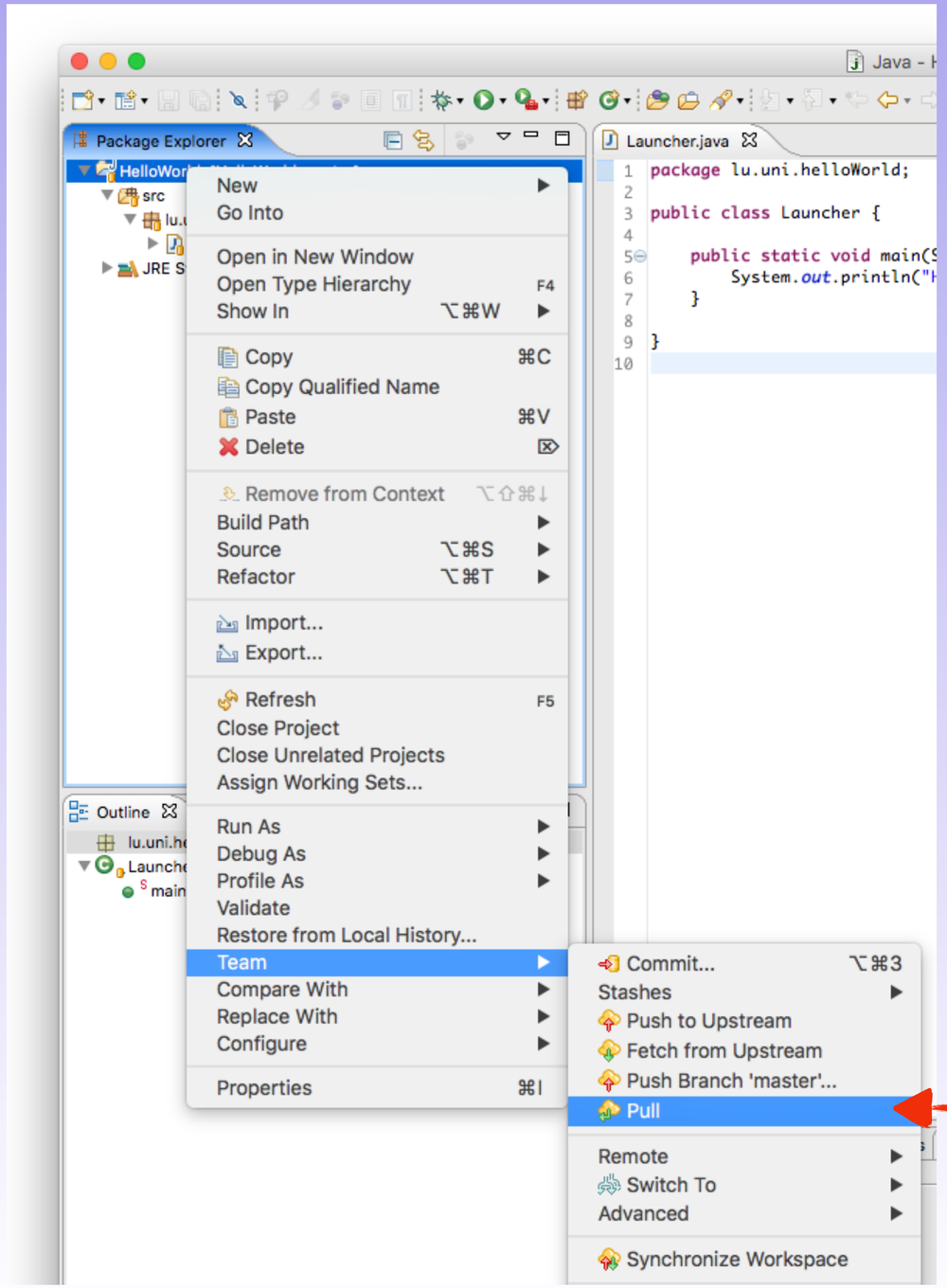

Java - HelloWorld/src/lu/uni/helloWorld/Launcher.java - Eclipse - /Users/Christian/Documents/Eclipse Workspace
Package Explorer
HelloWorld [HelloWorld master]
src
lu.uni.helloWorld
Launcher.java
JRE System Library [JavaSE-1.8]
Launcher.java
package lu.uni.helloWorld;
public class Launcher {
public static void main(String[] args) {
System.out.println("Hello, World!");
}
}
Outline
lu.uni.helloWorld
Launcher
main(String[]) : void
Push Branch master
Destination Git Repository
Enter the location of the destination repository.
Remote name: origin
Location
URI: https://ds-git.fstc.uni.lu/test/myProject.git
Host: ds-git.fstc.uni.lu
Repository path: /test/myProject.git
Connection
Protocol: https
Port:
Authentication
User: 0123456789
Password:
Store in Secure Store
< Back
Next >
Cancel
Finish
Declaration
Javadoc
Problems
Tasks
Console
No consoles to display at this time.
HelloWorld
Paste remote repository URL
Your Uni.lu credentials


Java - HelloWorld/src/lu/uni/helloWorld/Launcher.java - Eclipse - /Users/Christian/Documents/Eclipse Workspace
Quick Access
Java
C/C++
Package Explorer
HelloWorld [HelloWorld master]
src
lu.uni.helloWorld
Launcher.java
JRE System Library [JavaSE-1.8]
Launcher.java
1 package lu.uni.helloWorld;
2
3 public class Launcher {
4
5 public static void main(String[] args) {
6 System.out.println("Hello, World!");
7 }
8
9 }
10
Push Branch master
Push to branch in remote
Select a remote and the name the branch should have in the remote.
Source:
master
b7dca27 Initial commit
Destination:
Remote:
origin: https://ds-git.fstc.uni.lu/test/myProject.git
Branch:
master
Configure upstream for push and pull
When doing a pull
Merge upstream commits into local branch
Rebase commits of local branch onto upstream
Force overwrite branch in remote if it exists and has diverged
Show advanced push dialog
< Back
Next >
Cancel
Finish
Outline
lu.uni.helloWorld
Launcher
main(String[]) : void
Declaration
Javadoc
Problems
Tasks
Console
No consoles to display at this time.
HelloWorld

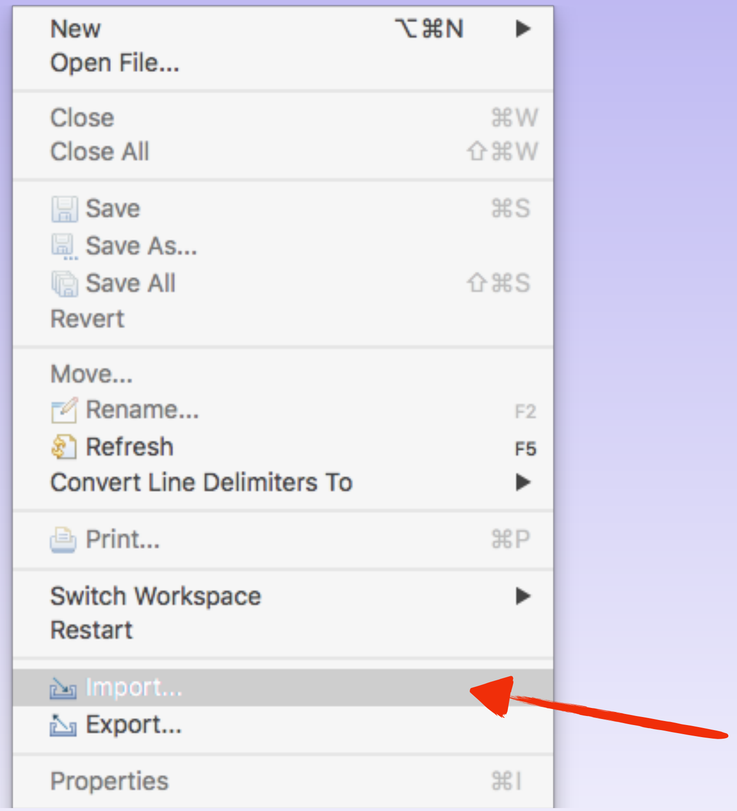

Java - HelloWorld/src/lu/uni/helloWorld/Launcher.java - Eclipse - /Users/Christian/Documents/Eclipse Workspace
Quick Access
Java
C/C++
Package Explorer
HelloWorld [HelloWorld master]
src
lu.uni.helloWorld
Launcher.java
JRE System Library [JavaSE-1.8]
Launcher.java
1 package lu.uni.helloWorld;
2
3 public class Launcher {
4
5 public static void main(String[] args) {
6 System.out.println("Hello, World!");
7 }
8
9 }
10
Push Branch master
Push Confirmation
Confirm following expected push result.
master → master [new branch]
Message Details
Repository https://ds-git.fstc.uni.lu/test/myProject.git
Cancel push if result would be different than above because of changes on remote
Show dialog with result only when it is different from the confirmed result above
< Back
Next >
Cancel
Finish
Outline
lu.uni.helloWorld
Launcher
main(String[]) : void
Declaration
Javadoc
Problems
Tasks
Console
No consoles to display at this time.
HelloWorld

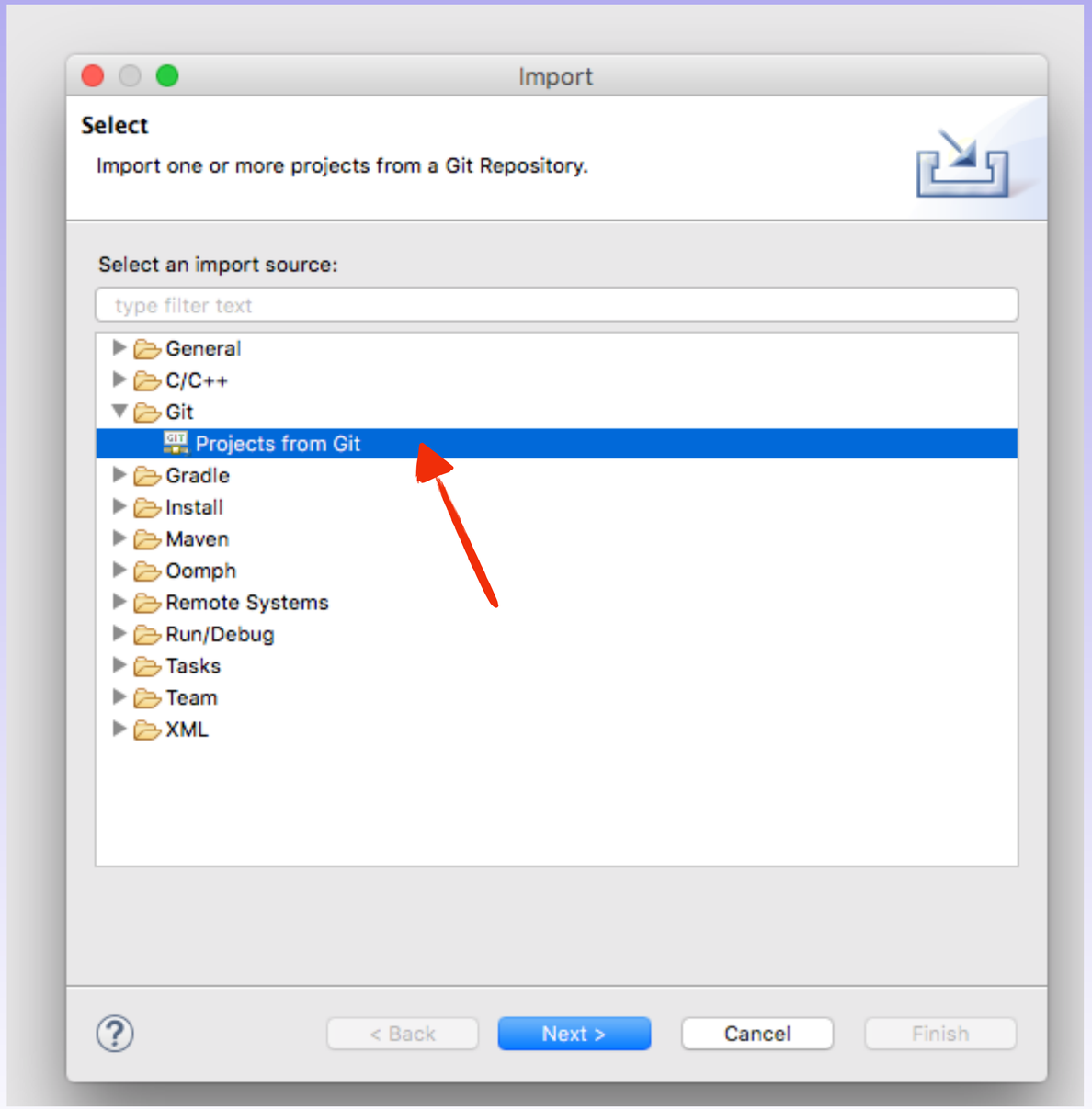

Java - HelloWorld/src/lu/uni/helloWorld/Launcher.java - Eclipse - /Users/Christian/Documents/Eclipse Workspace
Quick Access
Java
C/C++
Package Explorer
HelloWorld [HelloWorld master]
src
lu.uni.helloWorld
Launcher.java
JRE System Library [JavaSE-1.8]
Launcher.java
1 package lu.uni.helloWorld;
2
3 public class Launcher {
4
5 public static void main(String[] args) {
6 System.out.println("Hello, World!");
7 }
8
9 }
10
Push Results: https://ds-git.fstc.uni.lu/test/myProject.git
Pushed to https://ds-git.fstc.uni.lu/test/myProject.git
master → master [new branch]
Message Details
Repository https://ds-git.fstc.uni.lu/test/myProject.git
OK
Outline
lu.uni.helloWorld
Launcher
main(String[]) : void
Declaration
Javadoc
Problems
Tasks
Console
No consoles to display at this time.
HelloWorld

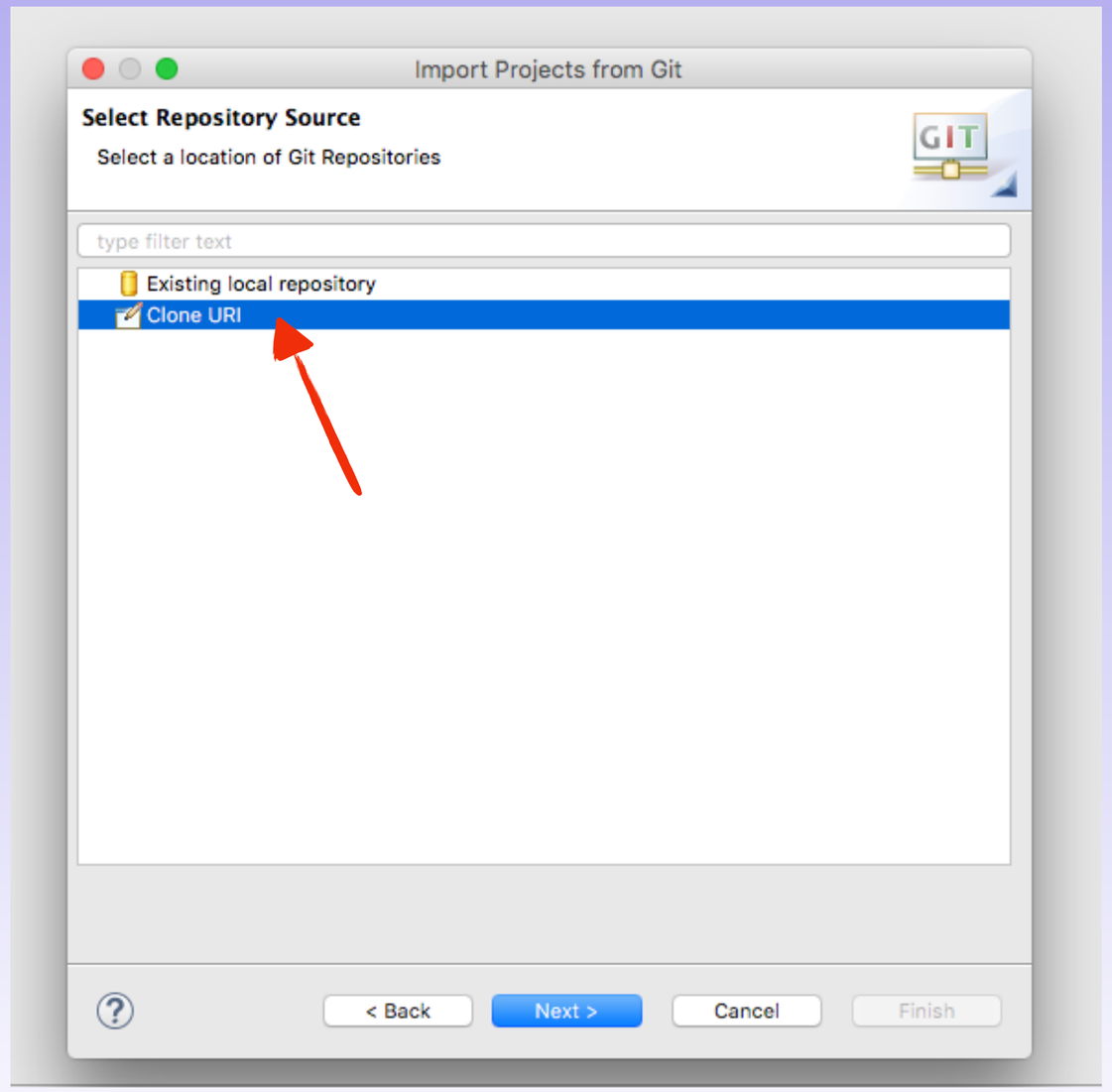
Pull current version



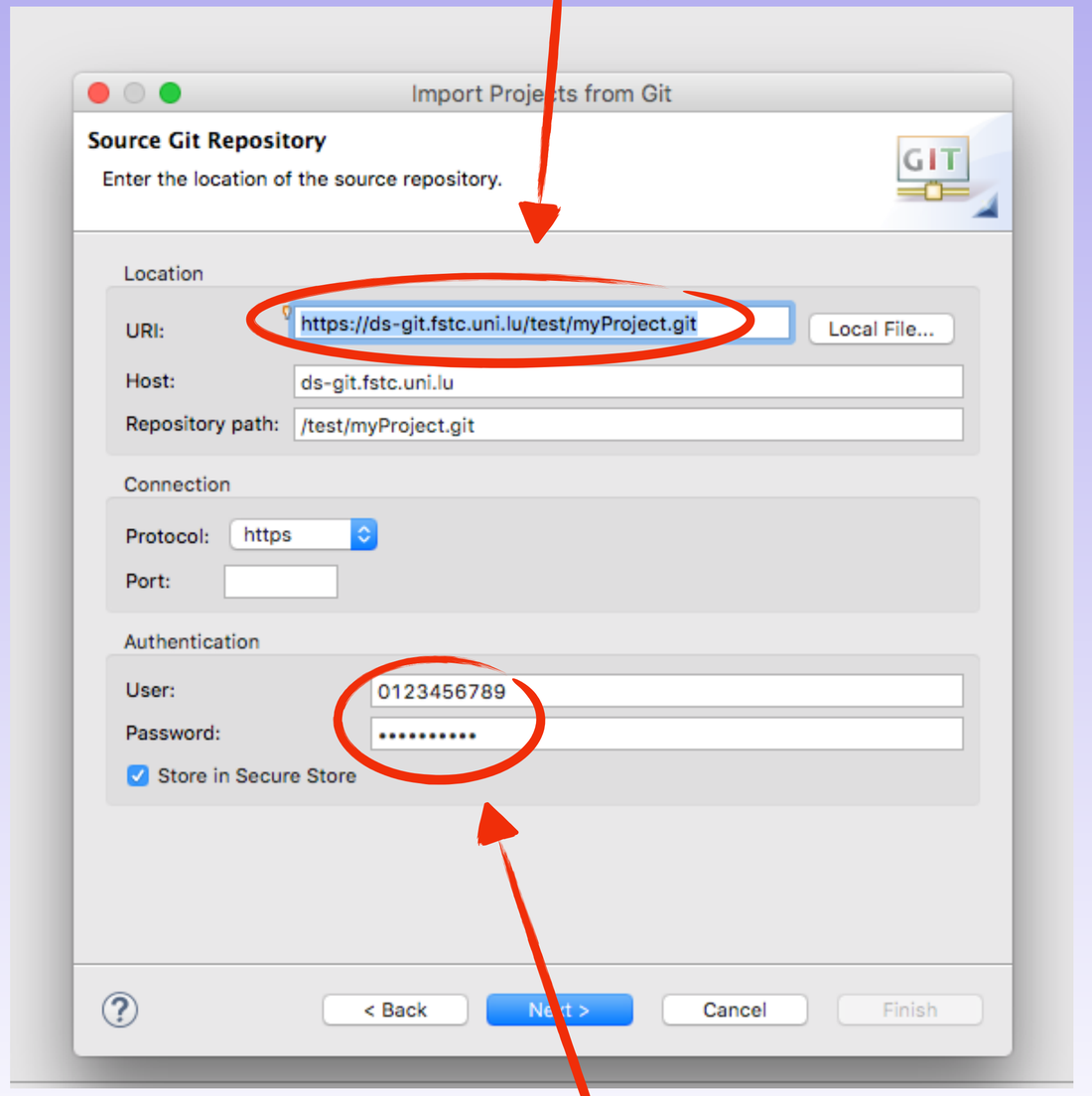
Clone existing project



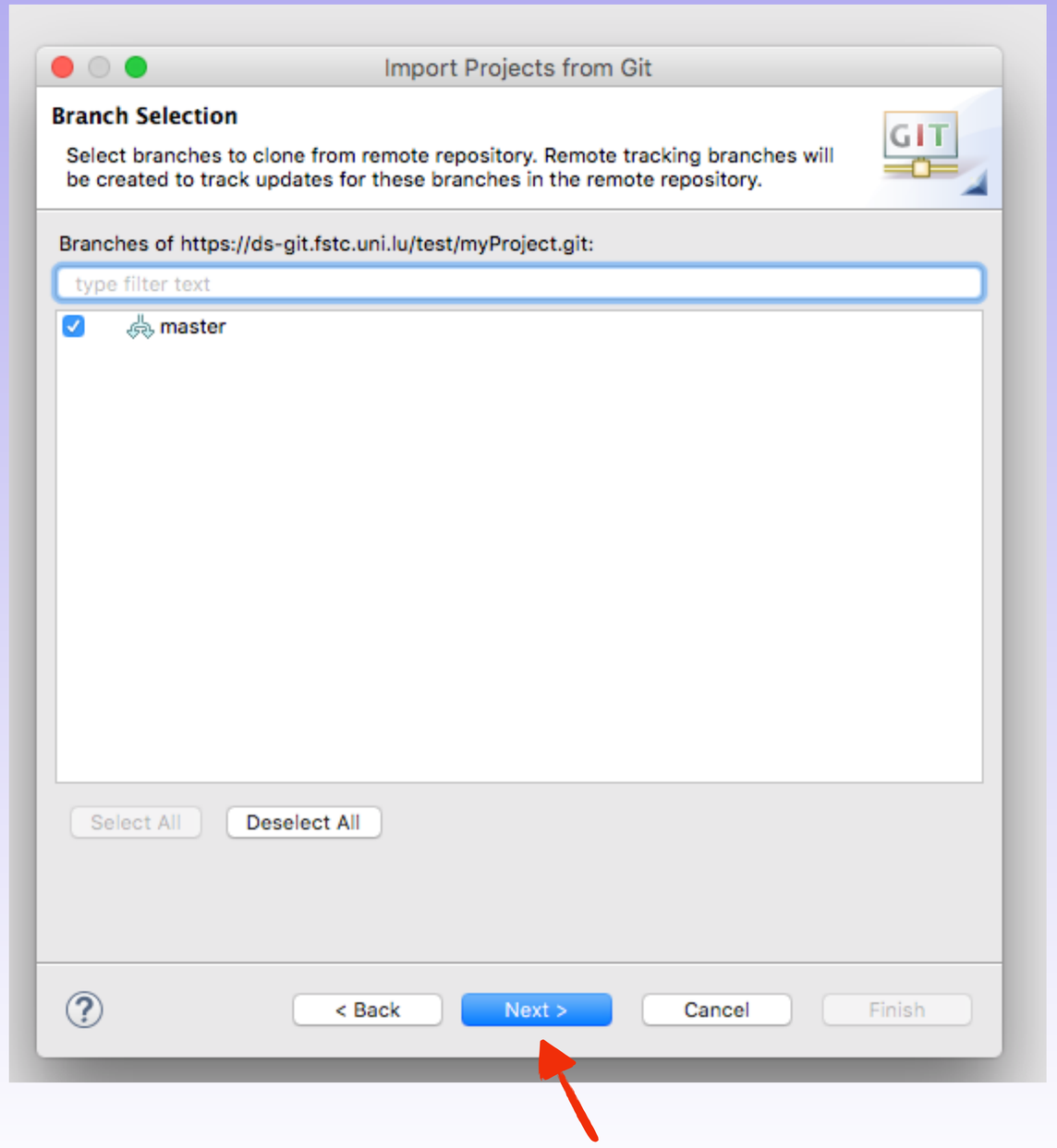


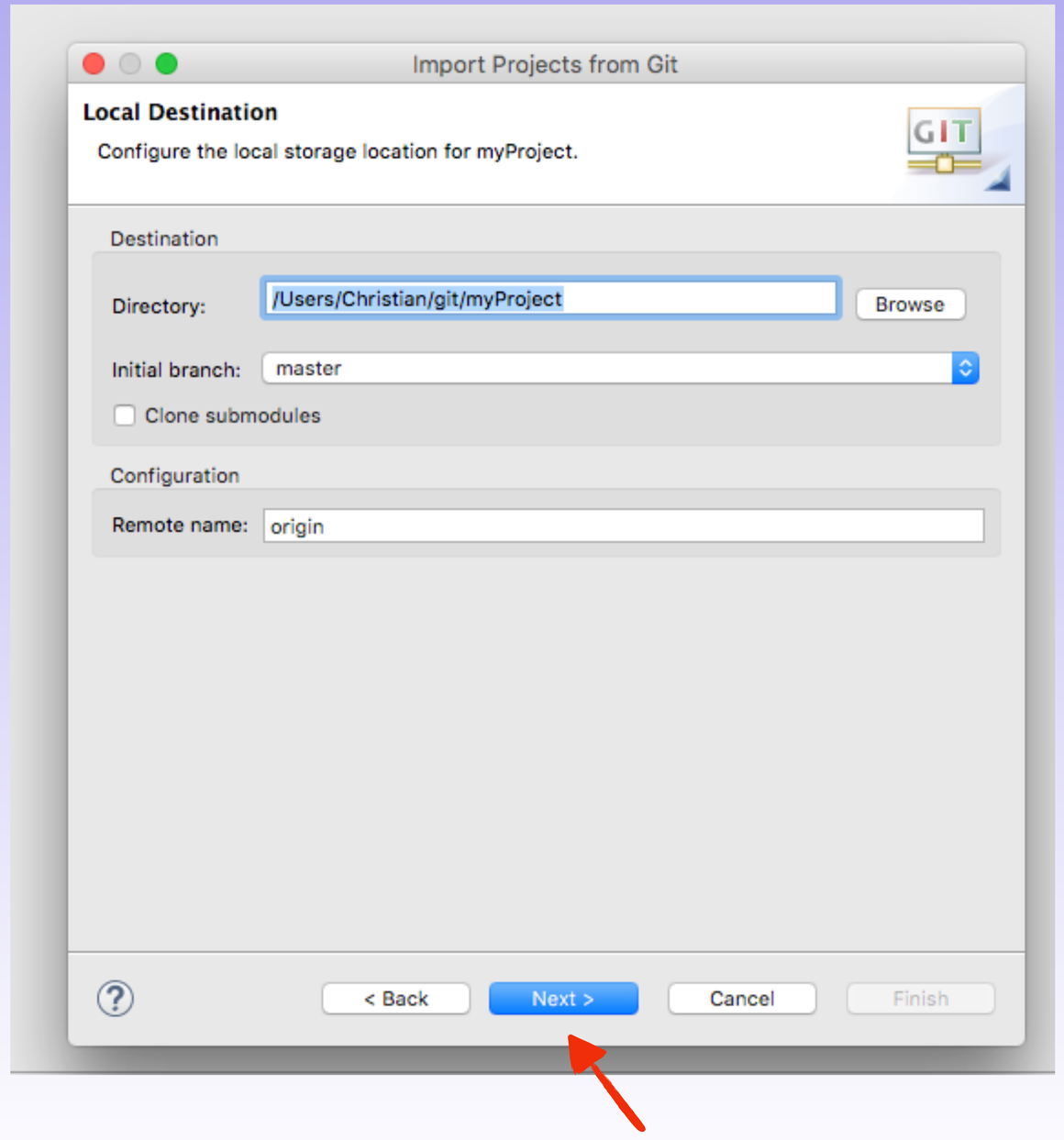


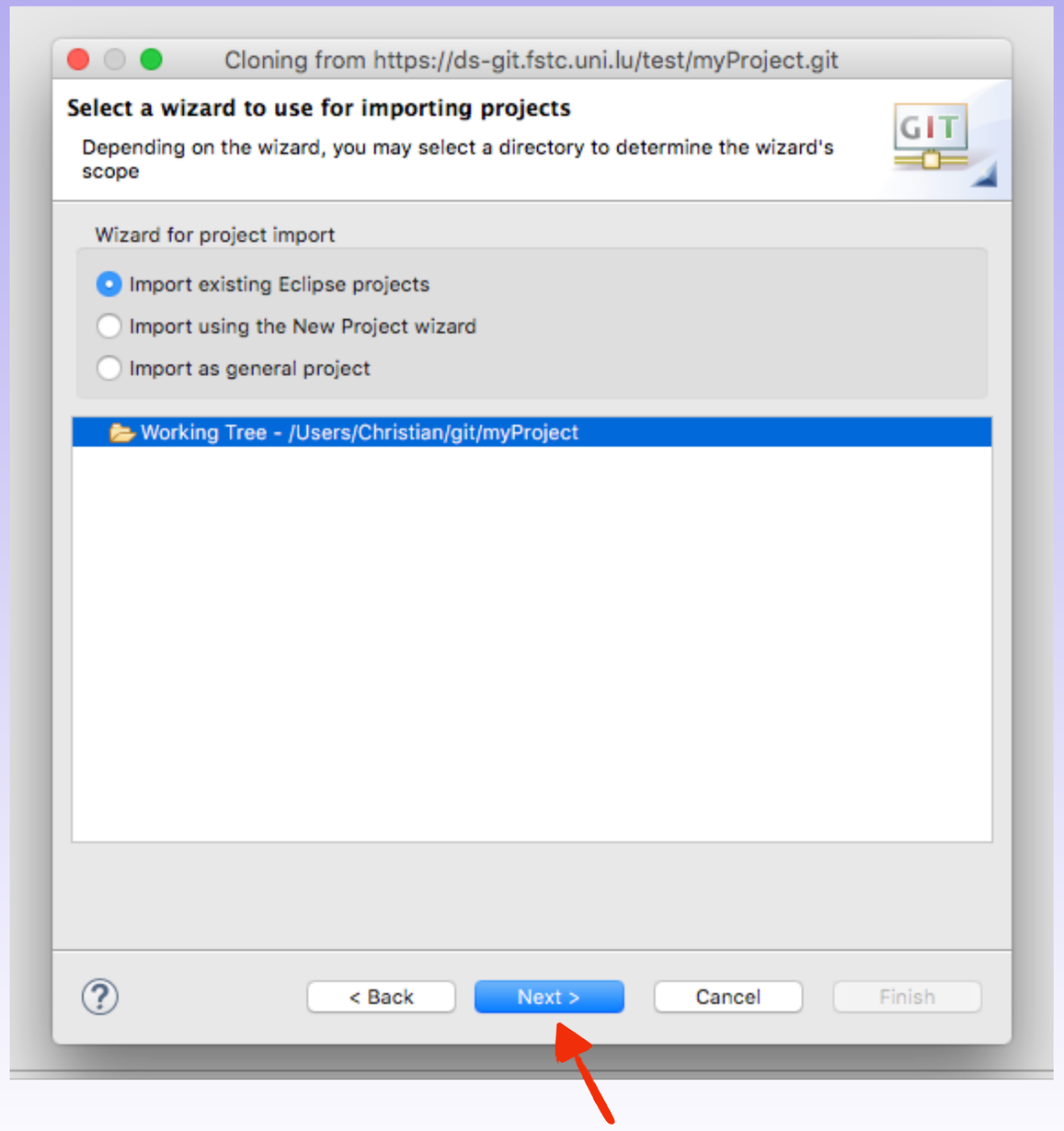
Paste remote repository URL

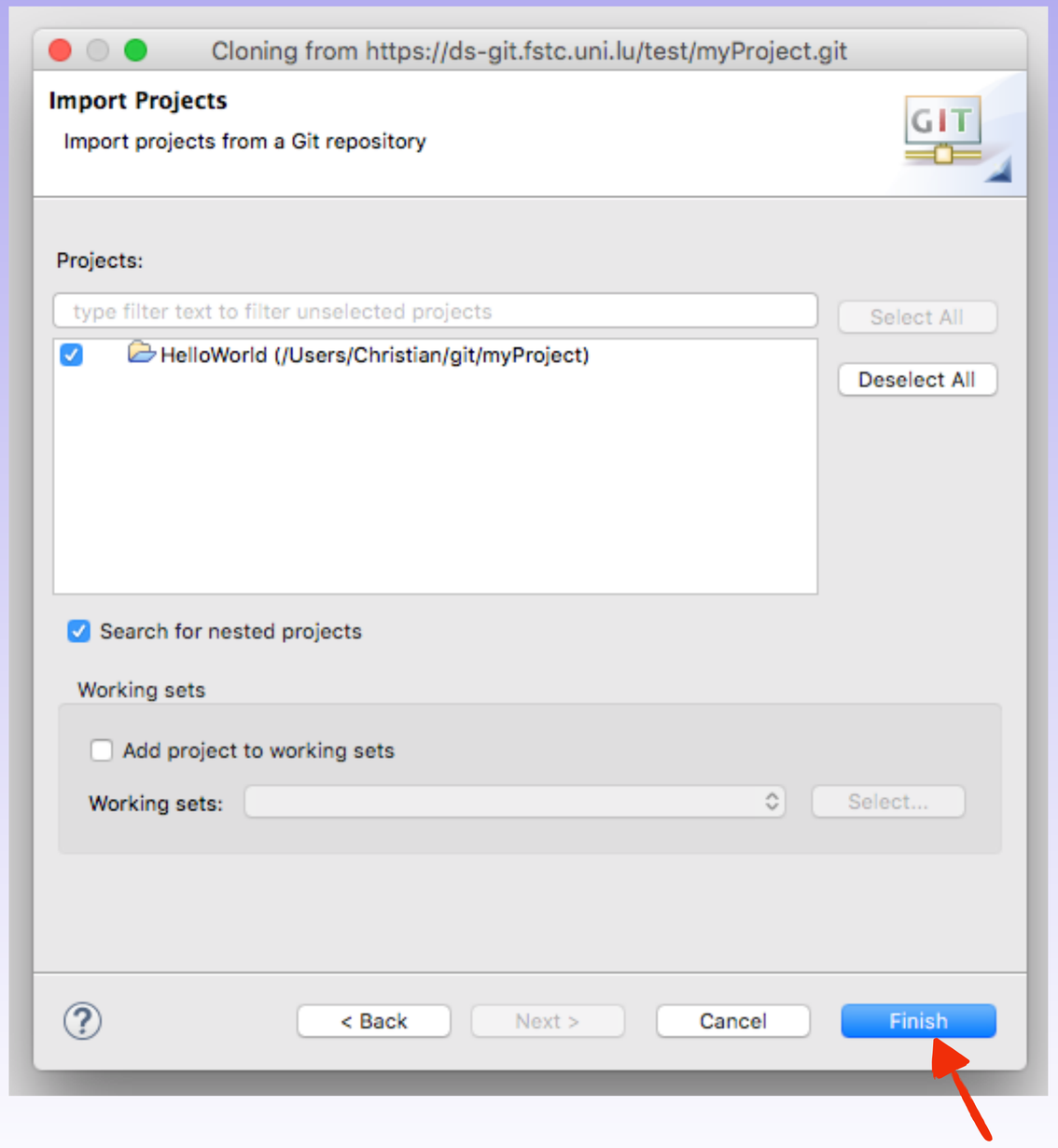


Your Uni.lu credentials


Import Projects from Git
Branch Selection
Select branches to clone from remote repository. Remote tracking branches will be created to track updates for these branches in the remote repository.
GIT
Branches of https://ds-git.fstc.uni.lu/test/myProject.git:
type filter text
master
Select All
Deselect All
< Back
Next >
Cancel
Finish

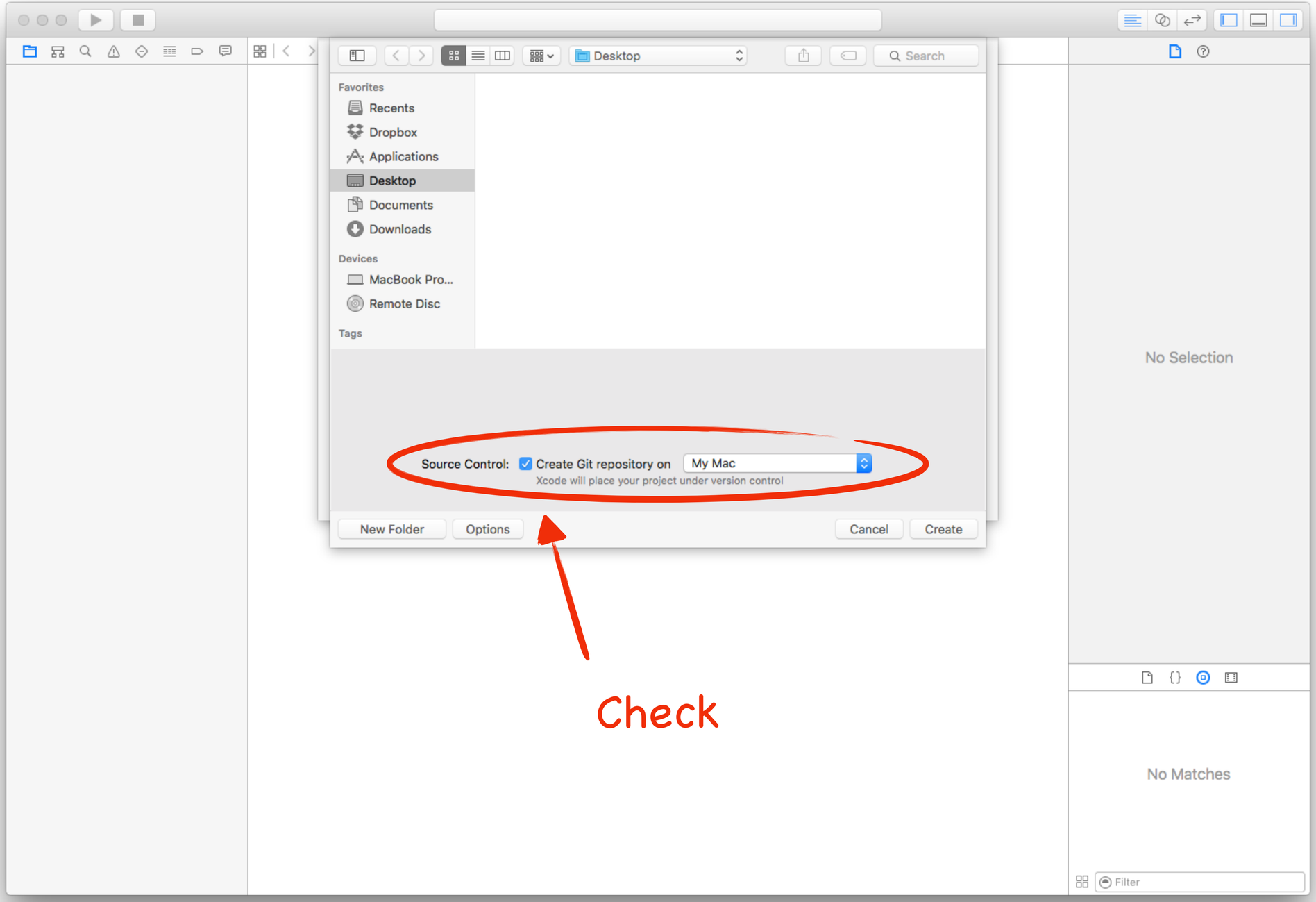

Import Projects from Git
Local Destination
Configure the local storage location for myProject.
GIT
Destination
Directory:
/Users/Christian/git/myProject
Browse
Initial branch:
master
Clone submodules
Configuration
Remote name:
origin
< Back
Next >
Cancel
Finish

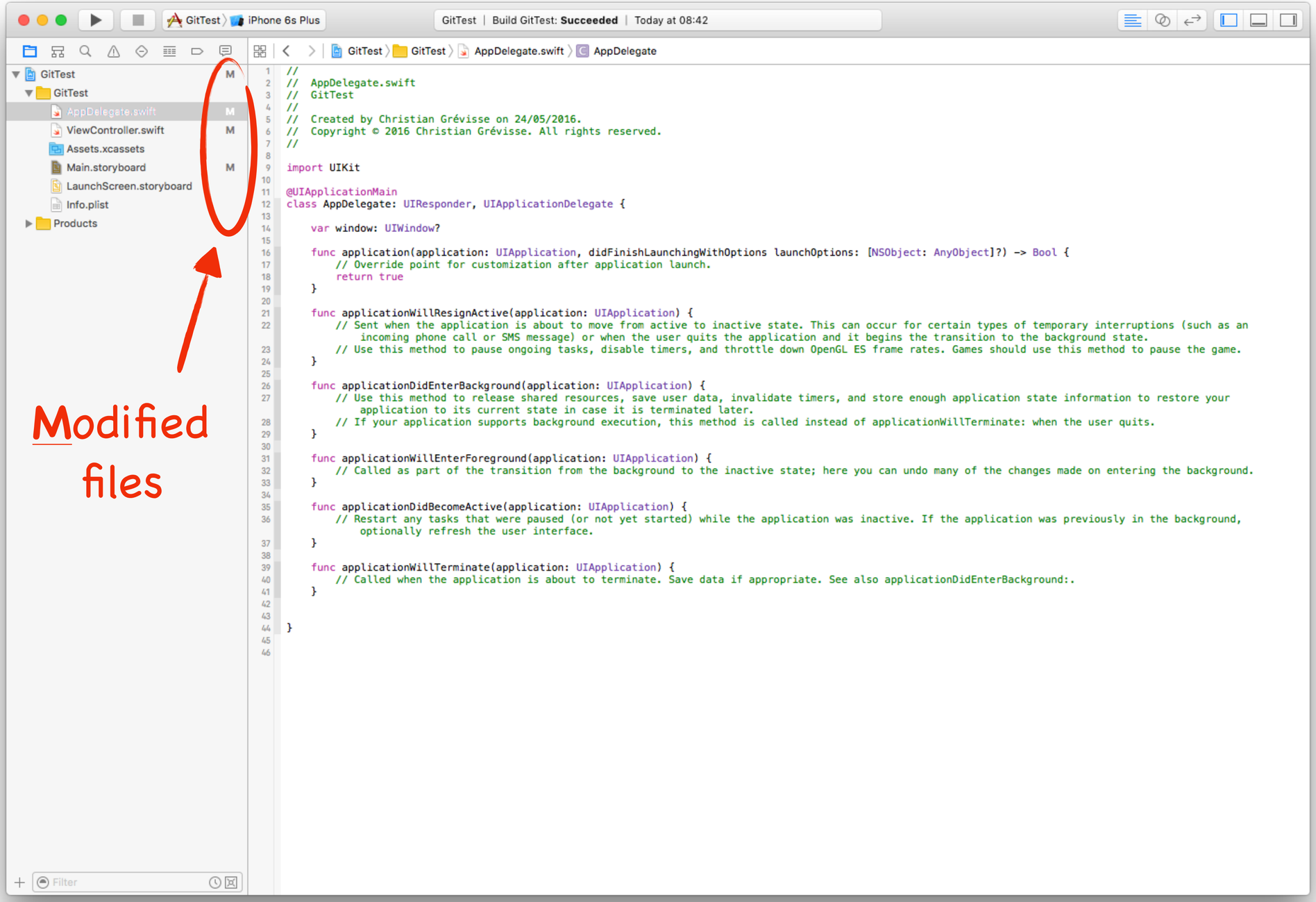

Cloning from https://ds-git.fstc.uni.lu/test/myProject.git
Select a wizard to use for importing projects
Depending on the wizard, you may select a directory to determine the wizard's scope
GIT
Wizard for project import
Import existing Eclipse projects
Import using the New Project wizard
Import as general project
Working Tree - /Users/Christian/git/myProject
< Back
Next >
Cancel
Finish

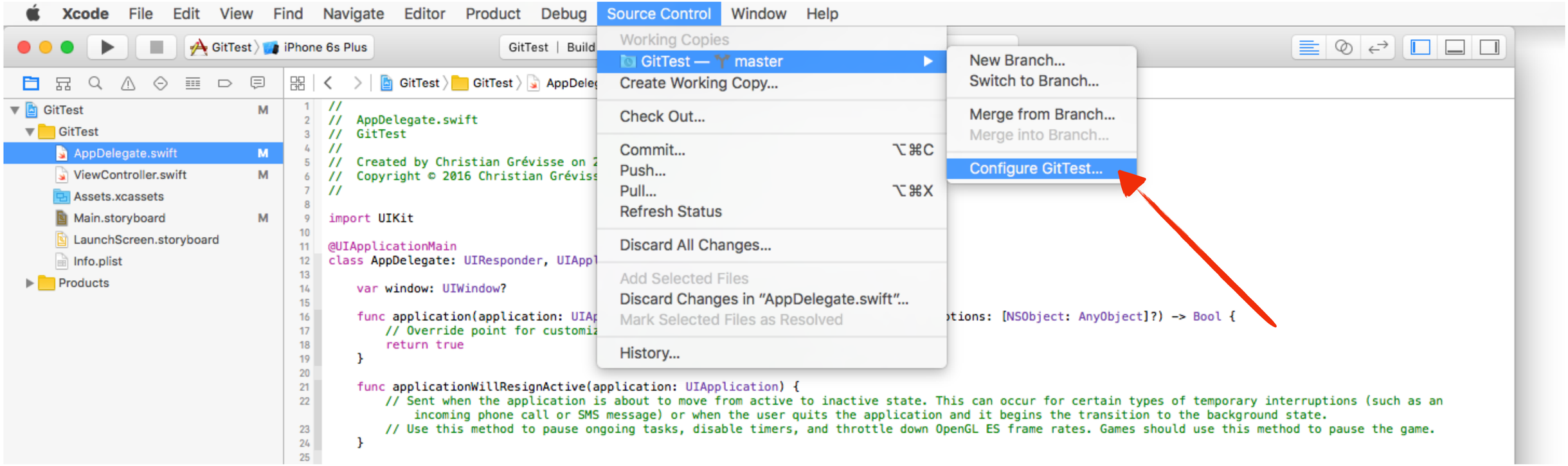

Cloning from https://ds-git.fstc.uni.lu/test/myProject.git
Import Projects
Import projects from a Git repository
GIT
Projects:
type filter text to filter unselected projects
Select All
HelloWorld (/Users/Christian/git/myProject)
Deselect All
Search for nested projects
Working sets
Add project to working sets
Working sets:
Select...
< Back
Next >
Cancel
Finish

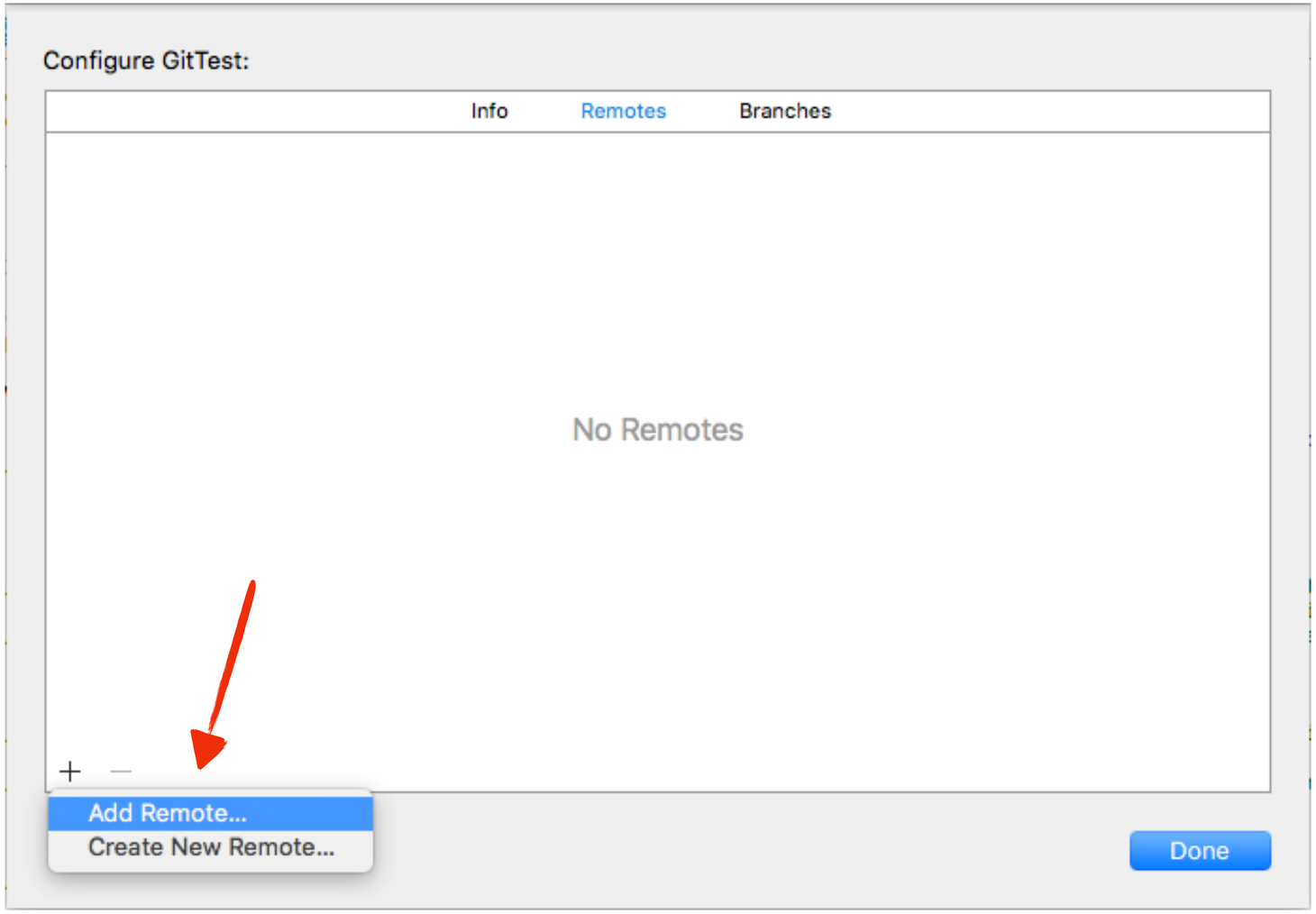
Xcode

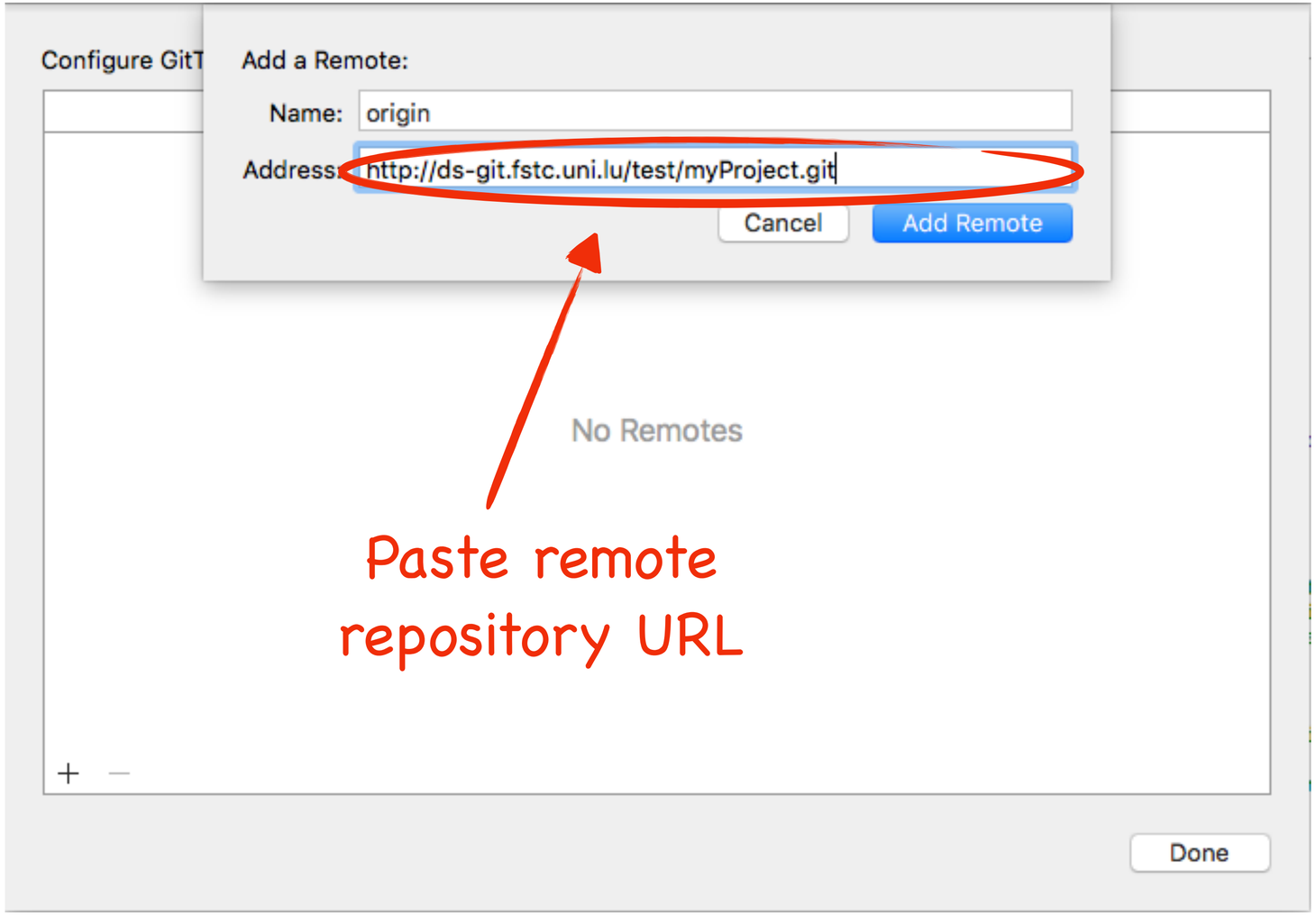
Initial repository setup

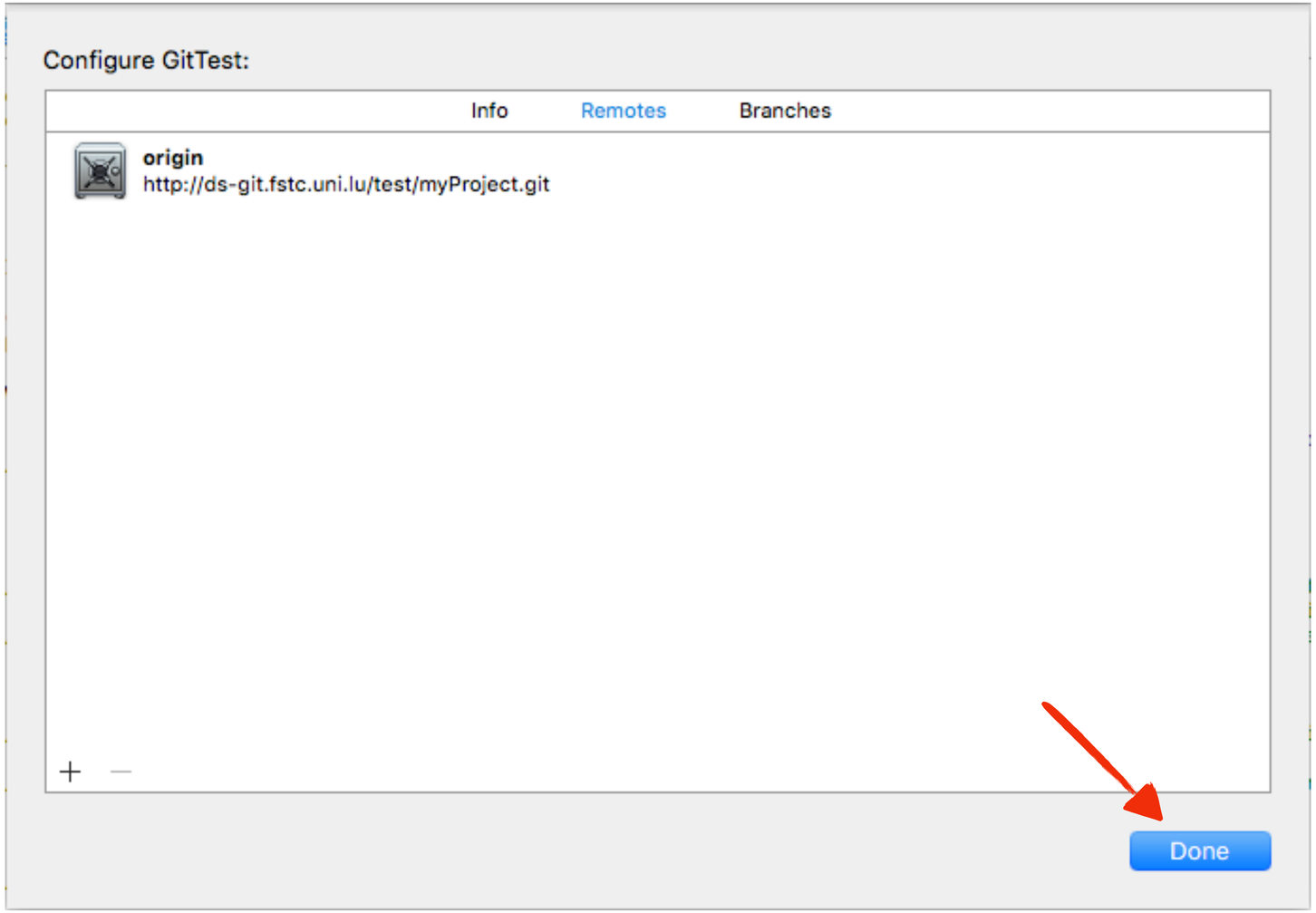


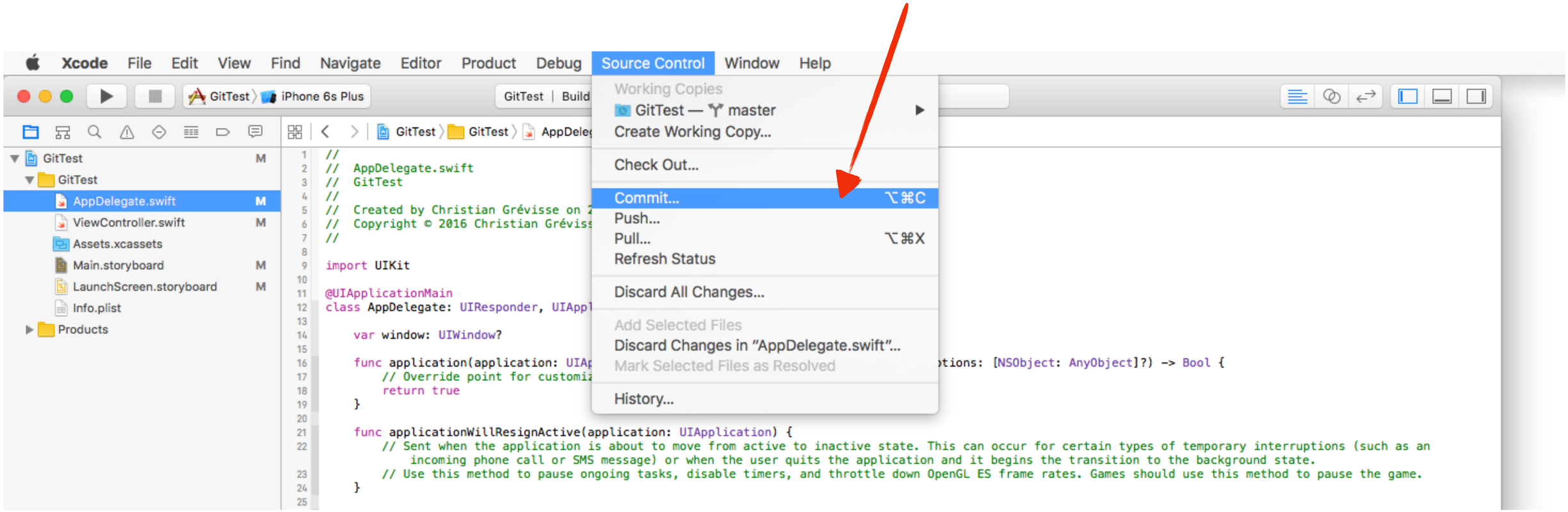

GitTest 〉 iPhone 6s Plus
GitTest | Build GitTest: Succeeded | Today at 08:42
GitTest 〉 GitTest 〉 AppDelegate.swift 〉 AppDelegate
GitTest M
GitTest
AppDelegate.swift M
ViewController.swift M
Assets.xcassets
Main.storyboard M
LaunchScreen.storyboard
Info.plist
Products
Filter
Modified files
//
// AppDelegate.swift
// GitTest
//
// Created by Christian Grévisse on 24/05/2016.
// Copyright © 2016 Christian Grévisse. All rights reserved.
//
import UIKit
@UIApplicationMain
class AppDelegate: UIResponder, UIApplicationDelegate {
var window: UIWindow?
func application(application: UIApplication, didFinishLaunchingWithOptions launchOptions: [NSObject: AnyObject]?) -> Bool {
// Override point for customization after application launch.
return true
}
func applicationWillResignActive(application: UIApplication) {
// Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
// Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
}
func applicationDidEnterBackground(application: UIApplication) {
// Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
// If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
}
func applicationWillEnterForeground(application: UIApplication) {
// Called as part of the transition from the background to the inactive state; here you can undo many of the changes made on entering the background.
}
func applicationDidBecomeActive(application: UIApplication) {
// Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
}
func applicationWillTerminate(application: UIApplication) {
// Called when the application is about to terminate. Save data if appropriate. See also applicationDidEnterBackground:.
}
}

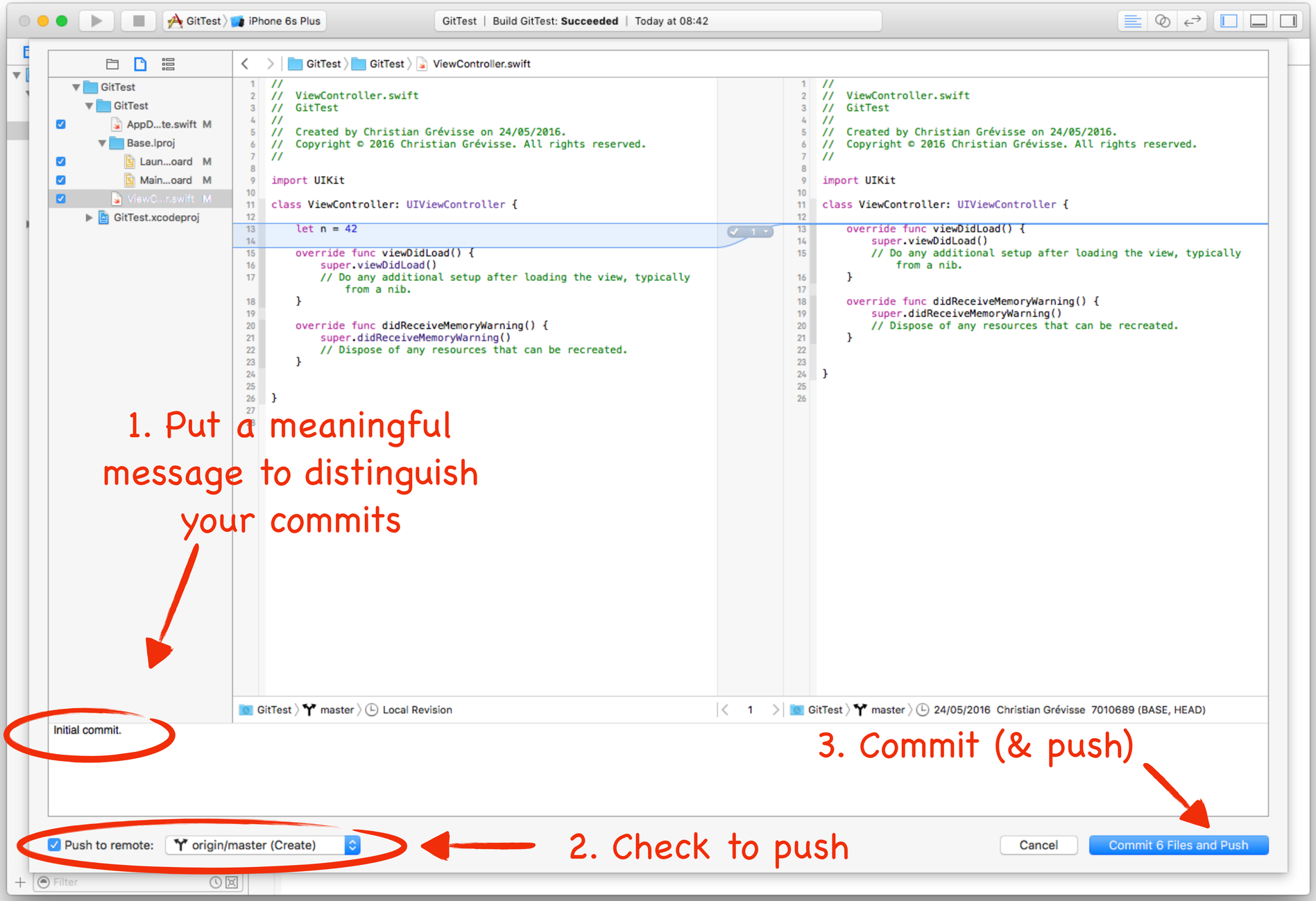

Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help
GitTest iPhone 6s Plus
GitTest | Build
Working Copies
GitTest — master
Create Working Copy...
Check Out...
Commit... ⌥⌘C
Push...
Pull... ⌥⌘X
Refresh Status
Discard All Changes...
Add Selected Files
Discard Changes in "AppDelegate.swift"...
Mark Selected Files as Resolved
History...
New Branch...
Switch to Branch...
Merge from Branch...
Merge into Branch...
Configure GitTest...
GitTest
GitTest
AppDelegate.swift M
ViewController.swift M
Assets.xcassets
Main.storyboard M
LaunchScreen.storyboard
Info.plist
Products
// AppDelegate.swift // GitTest //
// Created by Christian Grévisse on
// Copyright © 2016 Christian Grévis
import UIKit
@UIApplicationMain
class AppDelegate: UIResponder, UIAppl
var window: UIWindow?
func application(application: UIAp
// Override point for customiz
return true
func applicationWillResignActive(application: UIApplication) {
// Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
// Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.

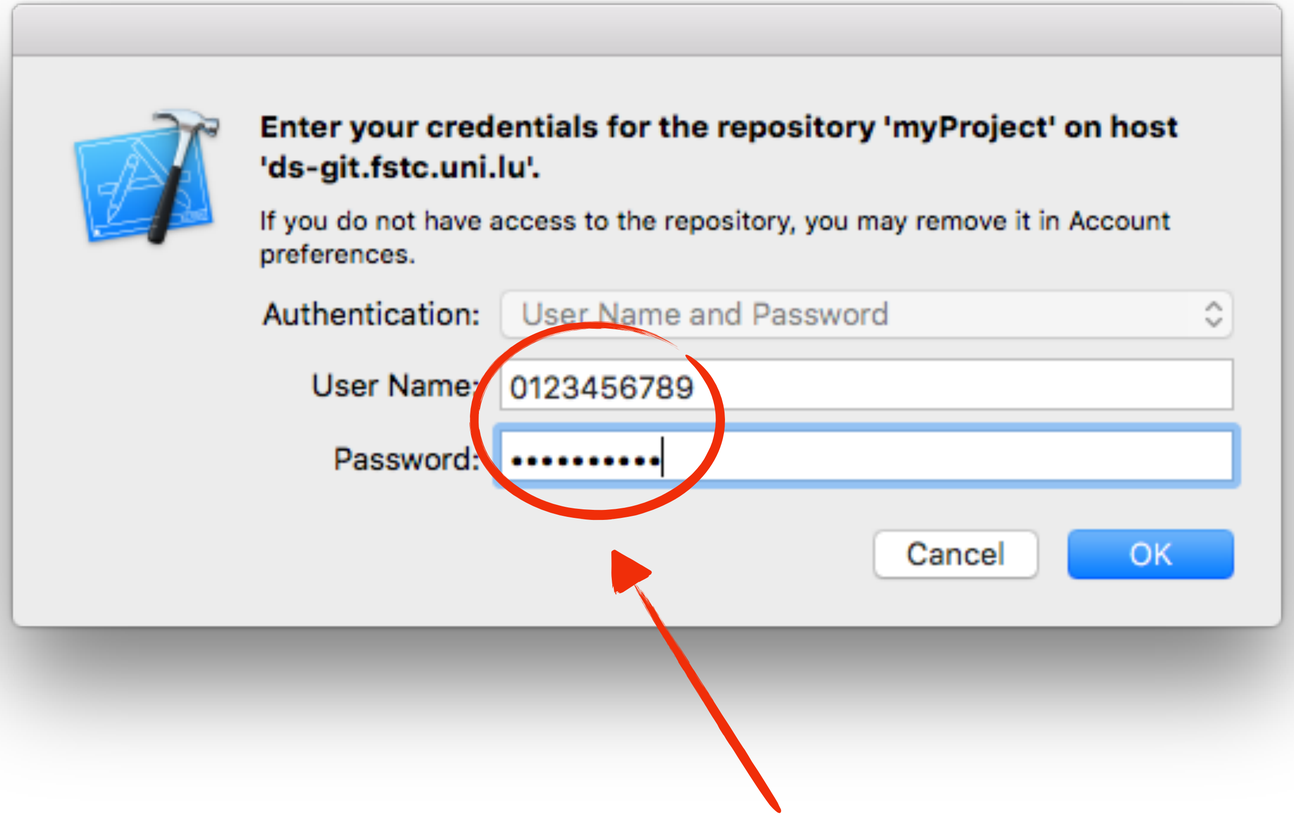

Configure GitTest:
Info
Remotes
Branches
No Remotes
+
—
Add Remote...
Create New Remote...
Done


Configure GitT
Add a Remote:
Name:
origin
Address:
http://ds-git.fstc.uni.lu/test/myProject.git
Cancel
Add Remote
No Remotes
Paste remote repository URL
Done

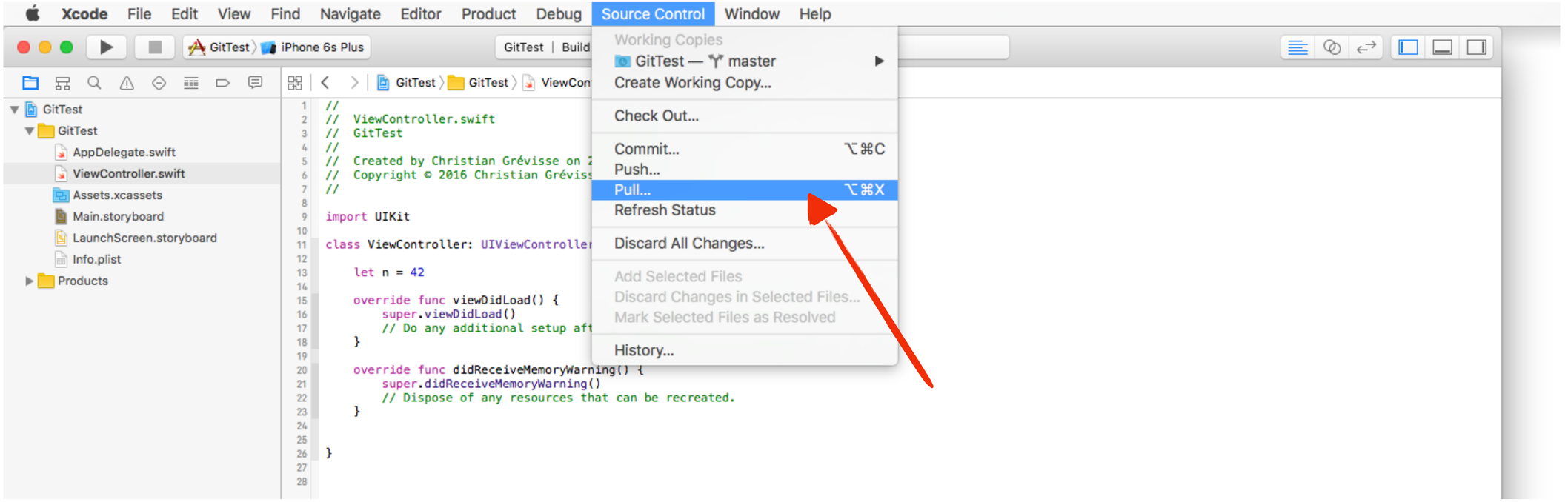

Configure GitTest:
Info
Remotes
Branches
origin
http://ds-git.fstc.uni.lu/test/myProject.git
Done


Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help
GitTest iPhone 6s Plus
GitTest | Build
GitTest
GitTest
AppDeleg
GitTest M
GitTest
AppDelegate.swift M
ViewController.swift M
Assets.xcassets
Main.storyboard M
LaunchScreen.storyboard M
Info.plist
Products
Working Copies
GitTest — master
Create Working Copy...
Check Out...
Commit... ⌥⌘C
Push...
Pull... ⌥⌘X
Refresh Status
Discard All Changes...
Add Selected Files
Discard Changes in "AppDelegate.swift"...
Mark Selected Files as Resolved
History...
//
// AppDelegate.swift
// GitTest
//
// Created by Christian Grévisse on
// Copyright © 2016 Christian Grévis
//
import UIKit
@UIApplicationMain
class AppDelegate: UIResponder, UIAppl
var window: UIWindow?
func application(application: UIAp
// Override point for customiz
return true
}
func applicationWillResignActive(application: UIApplication) {
// Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an
incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
// Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should use this method to pause the game.
}
tions: [NSObject: AnyObject]?) -> Bool {

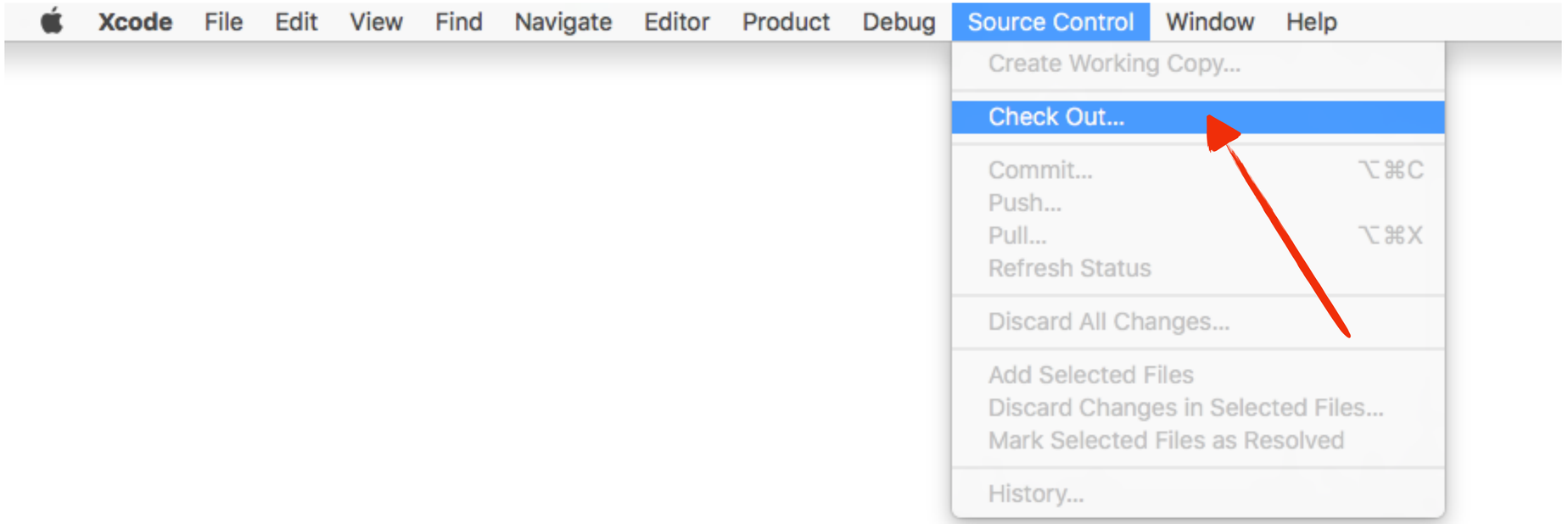

GitTest | Build GitTest: Succeeded | Today at 08:42
GitTest
GitTest
AppD...te.swift M
Base.lproj
Laun...oard M
Main...oard M
ViewC...r.swift M
GitTest.xcodeproj
GitTest › GitTest › ViewController.swift
1. Put a meaningful message to distinguish your commits
Initial commit.
GitTest › master › Local Revision
GitTest › master › 24/05/2016 Christian Grévisse 7010689 (BASE, HEAD)
3. Commit (& push)
Push to remote: origin/master (Create)
2. Check to push
Cancel
Commit 6 Files and Push

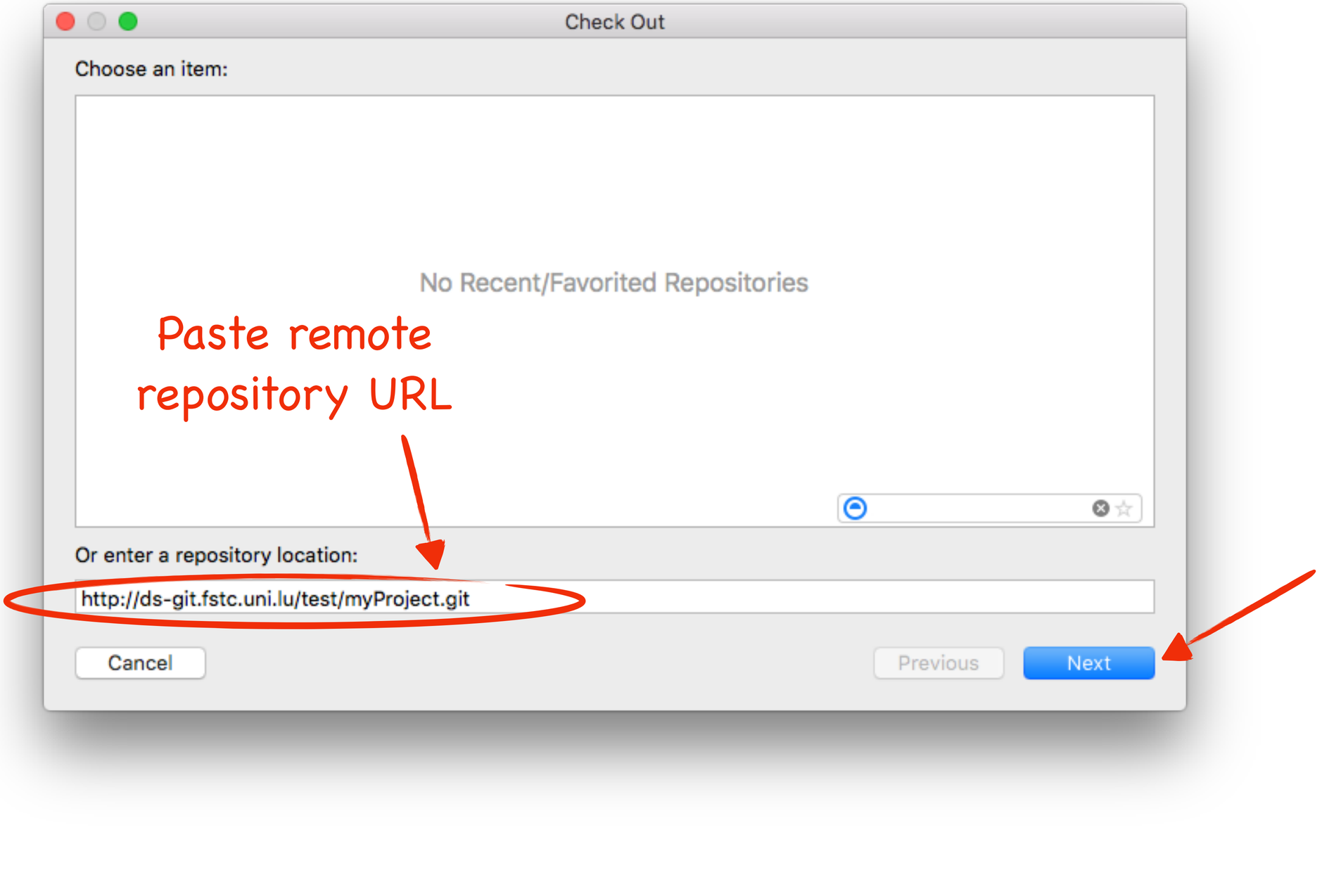

Enter your credentials for the repository 'myProject' on host 'ds-git.fstc.uni.lu'.
If you do not have access to the repository, you may remove it in Account preferences.
Authentication:
User Name and Password
User Name:
0123456789
Password:
Cancel
OK
Your Uni.lu credentials

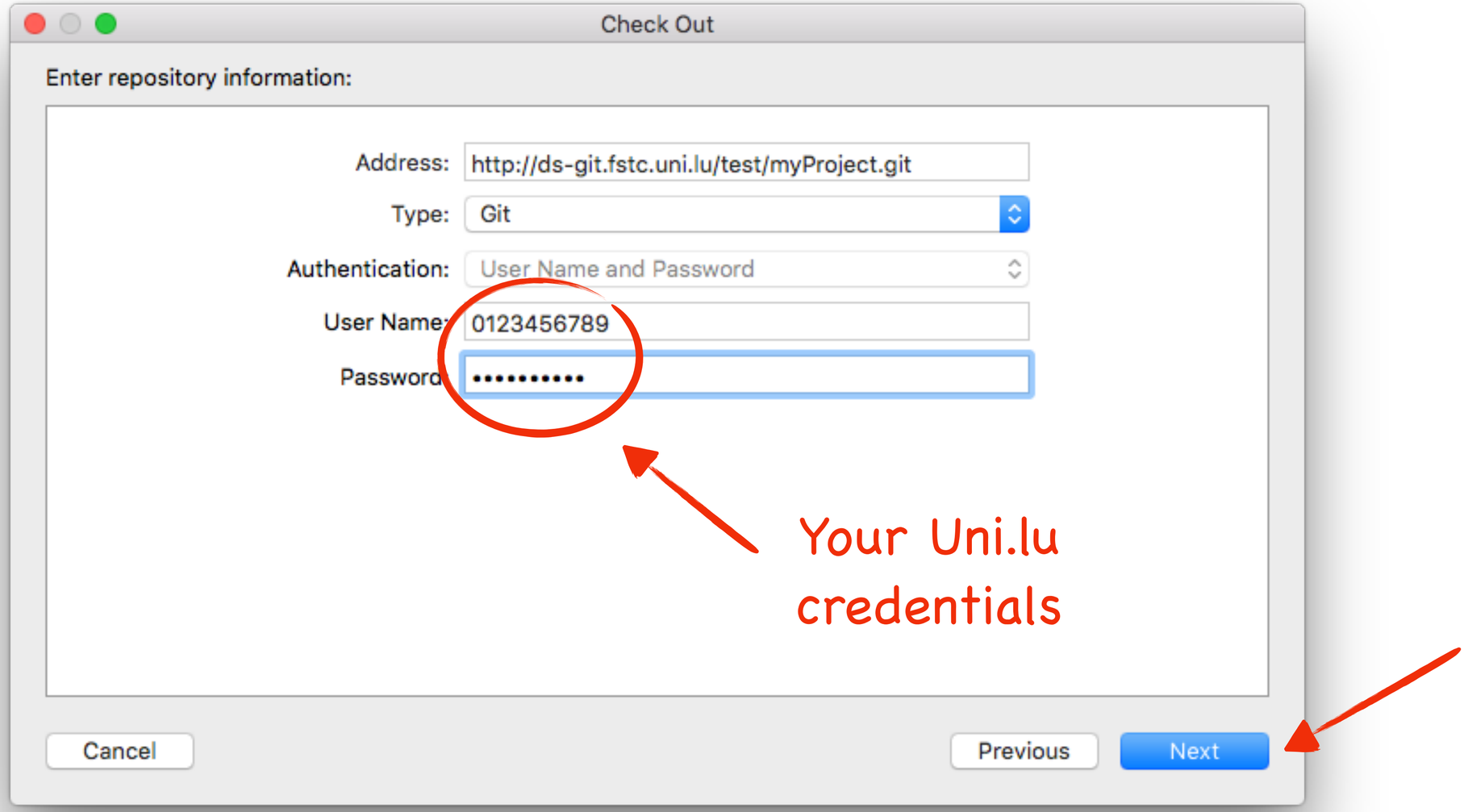
Pull current version

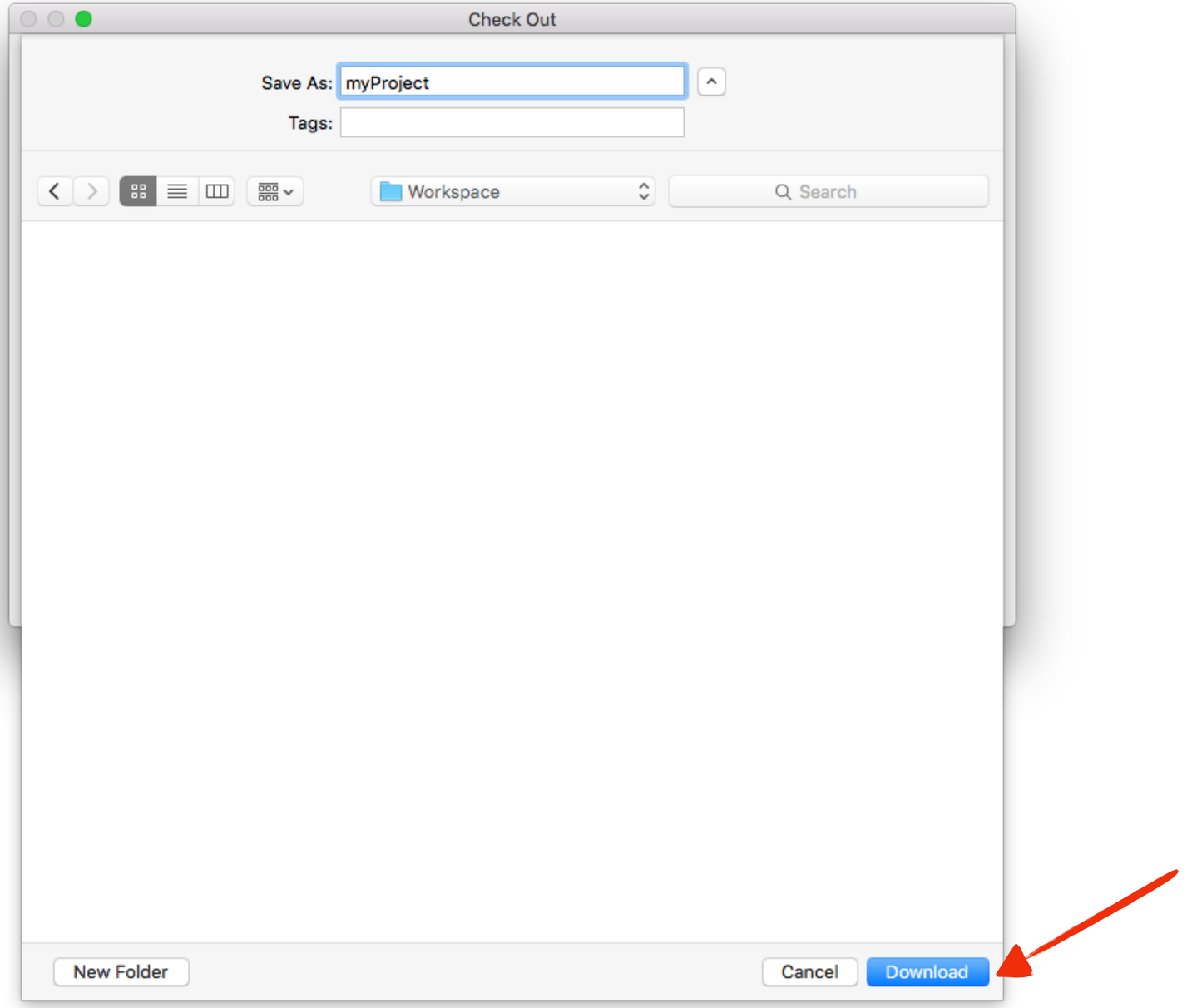


Clone existing project



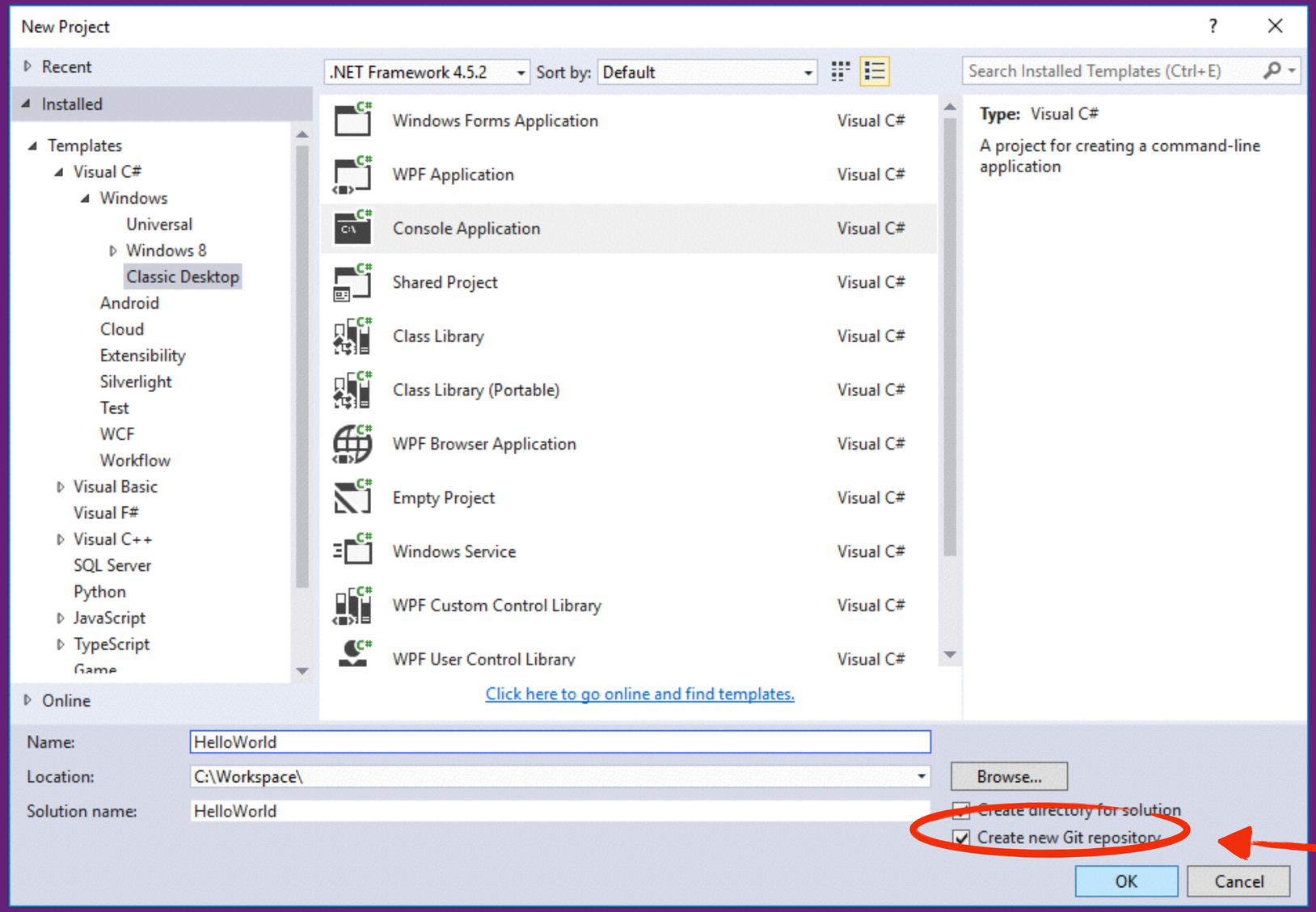

Check Out
Choose an item:
No Recent/Favorited Repositories
Paste remote repository URL
Or enter a repository location:
http://ds-git.fstc.uni.lu/test/myProject.git
Cancel
Previous
Next

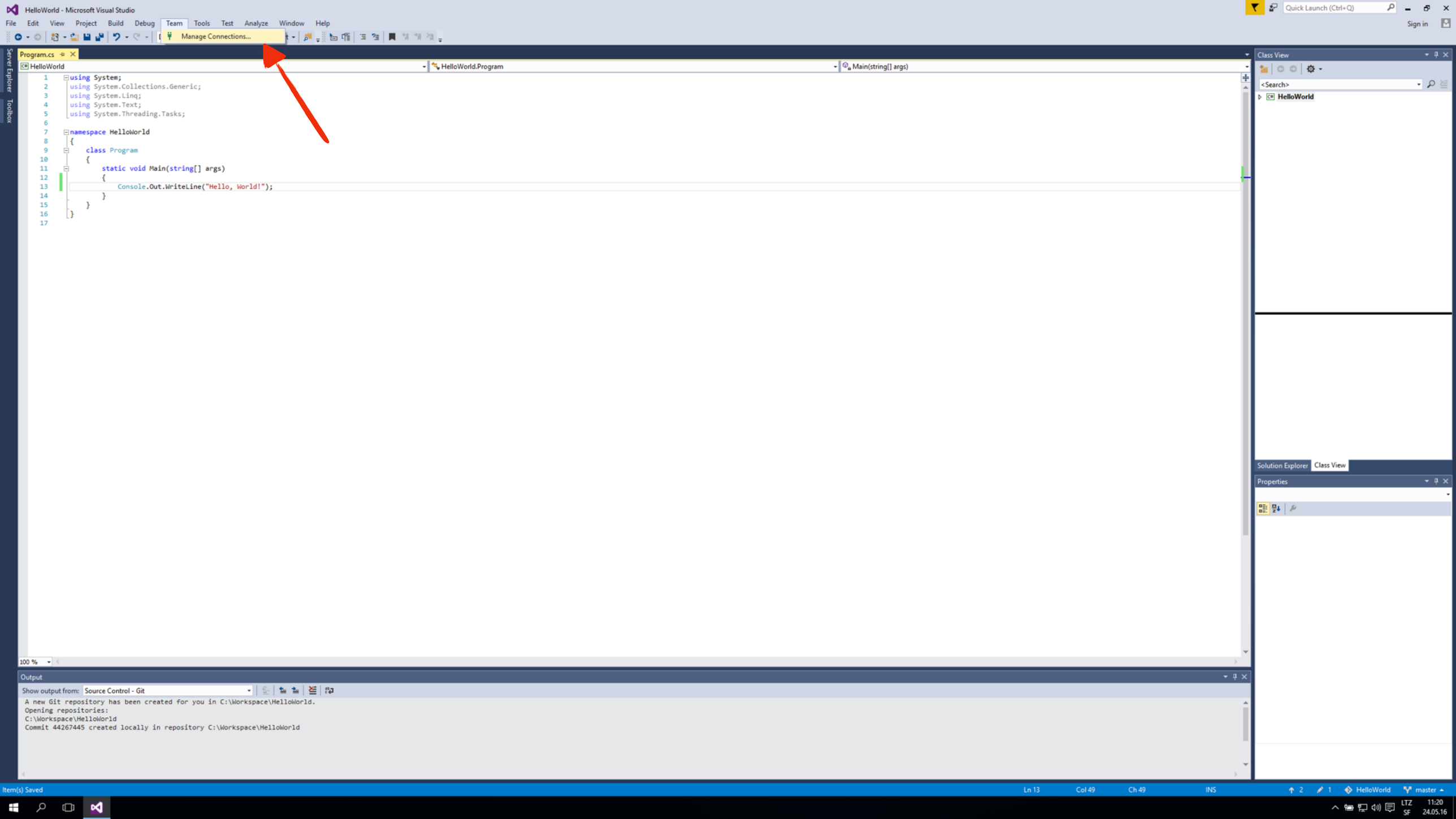

Check Out
Enter repository information:
Address:
http://ds-git.fstc.uni.lu/test/myProject.git
Type:
Git
Authentication:
User Name and Password
User Name:
0123456789
Password:
Your Uni.lu credentials
Cancel
Previous
Next

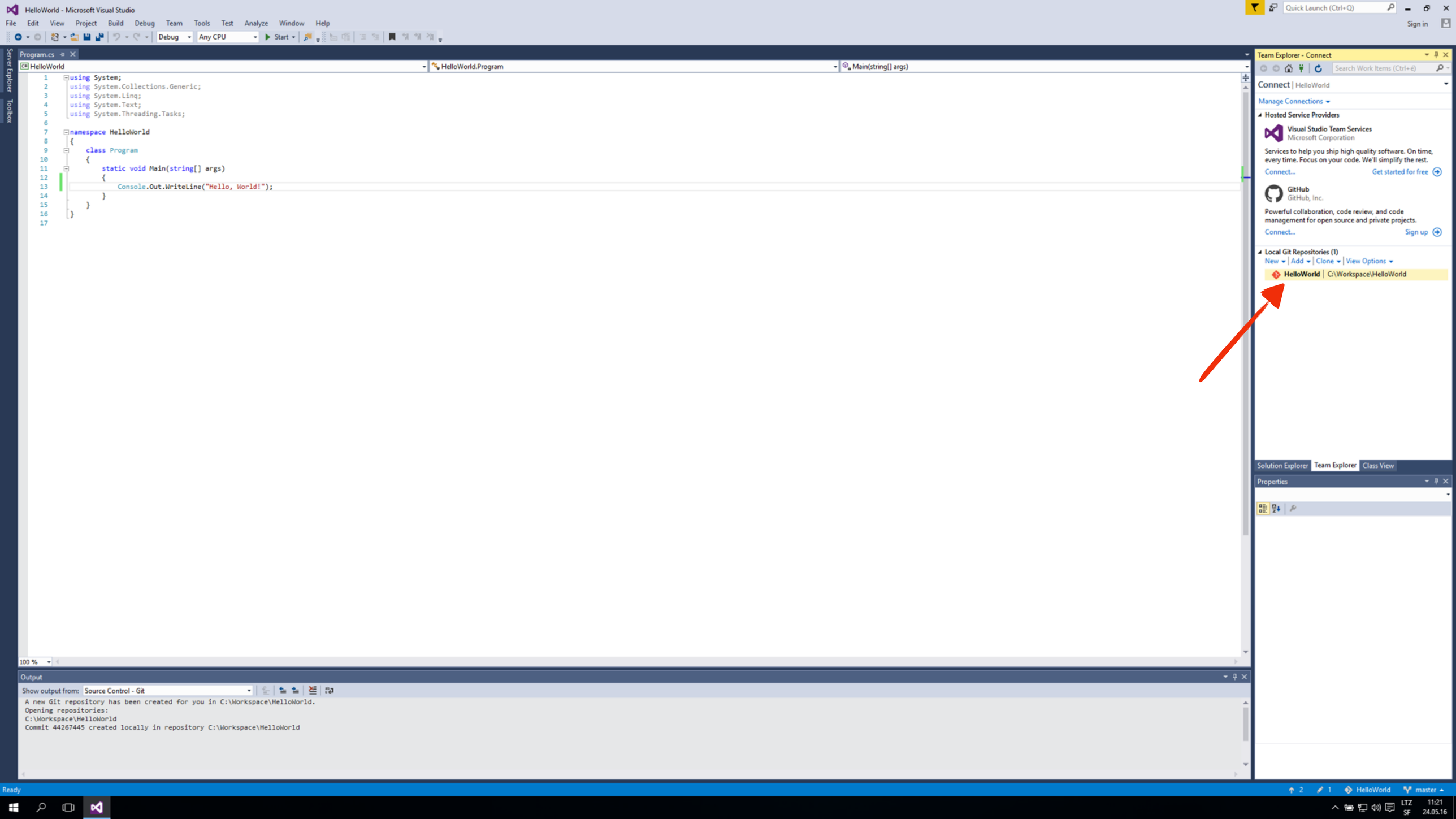

Check Out
Save As: myProject
Tags:
Workspace
Search
New Folder
Cancel
Download

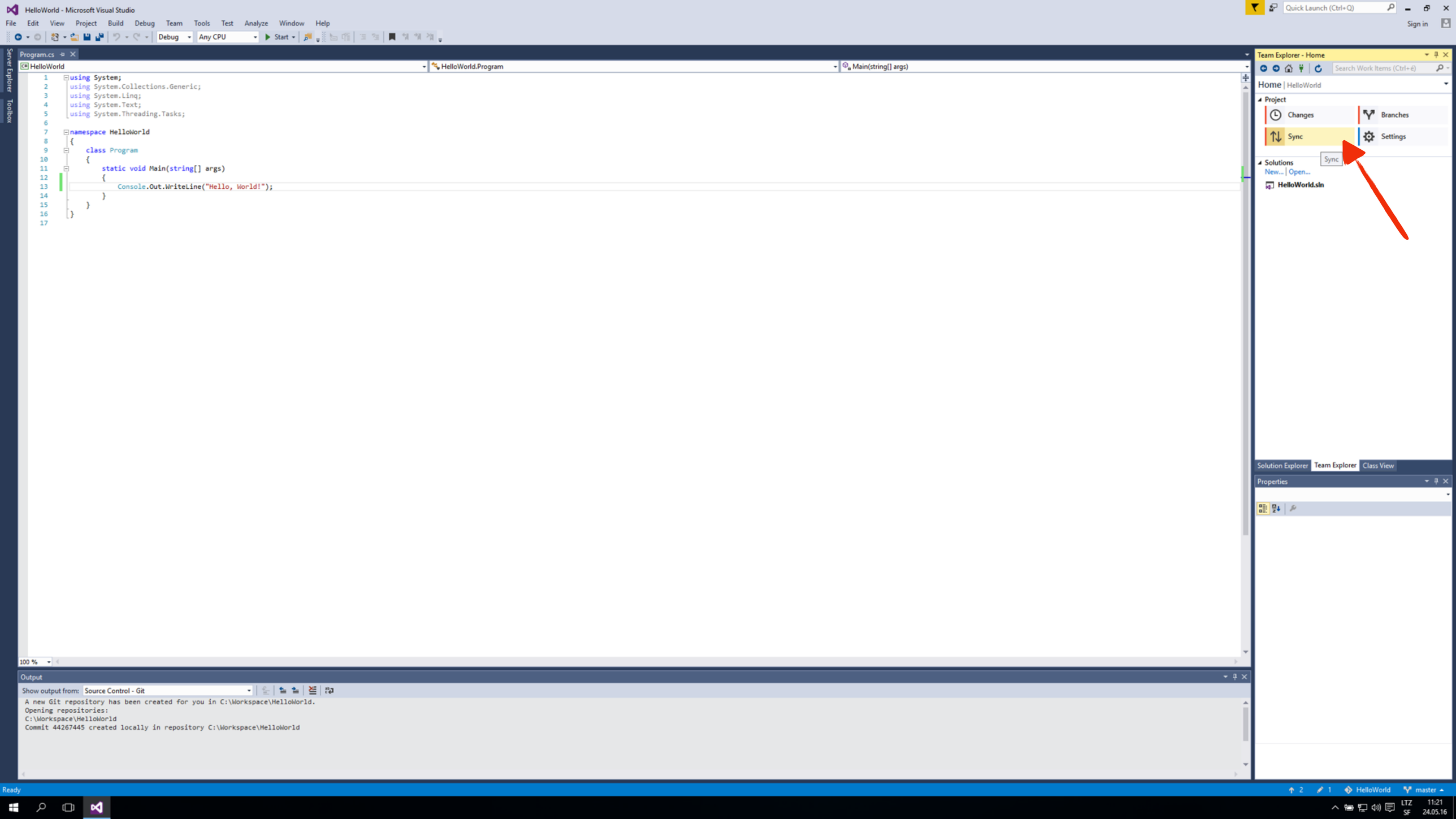
Visual Studio

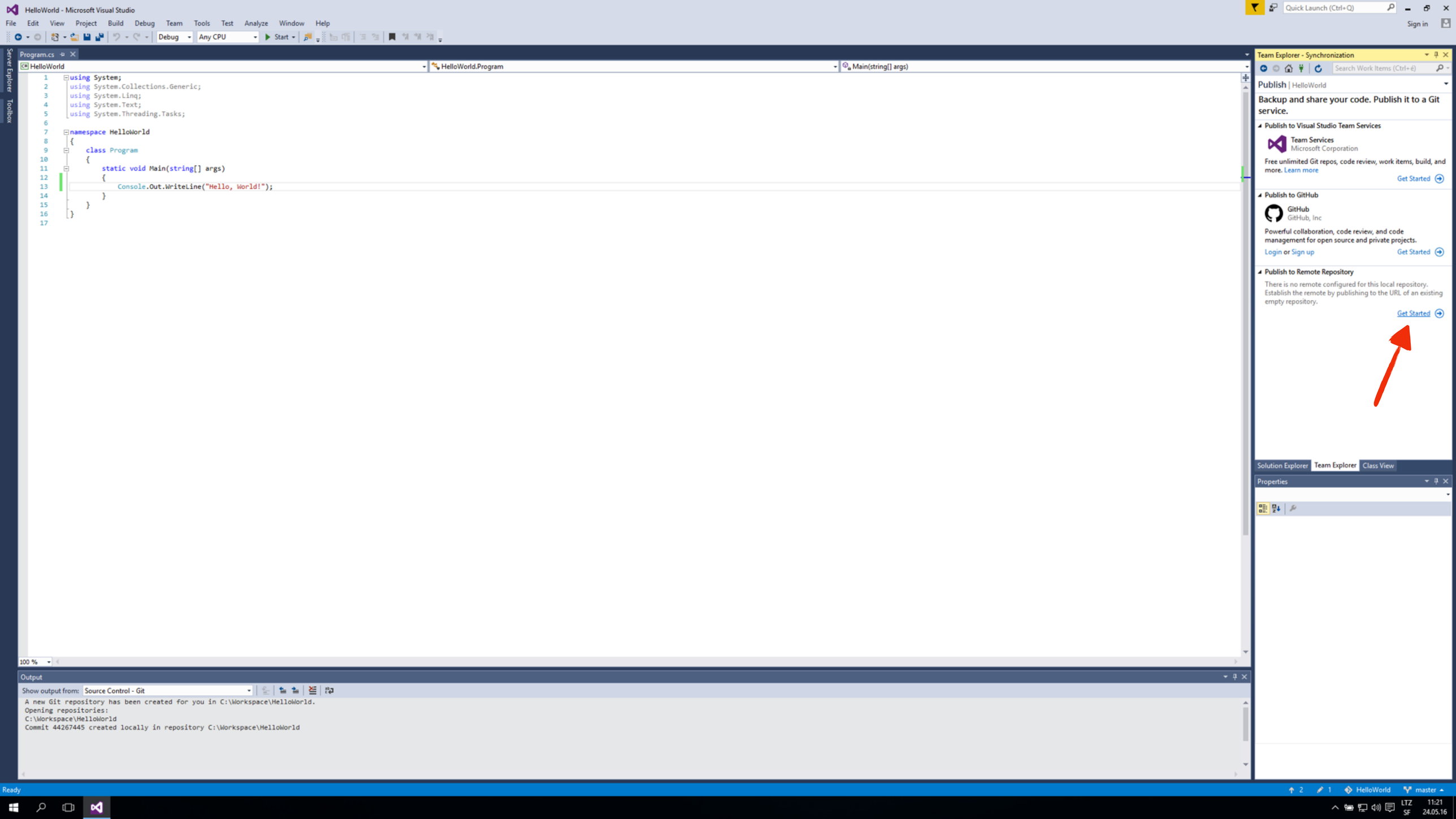
Initial repository setup

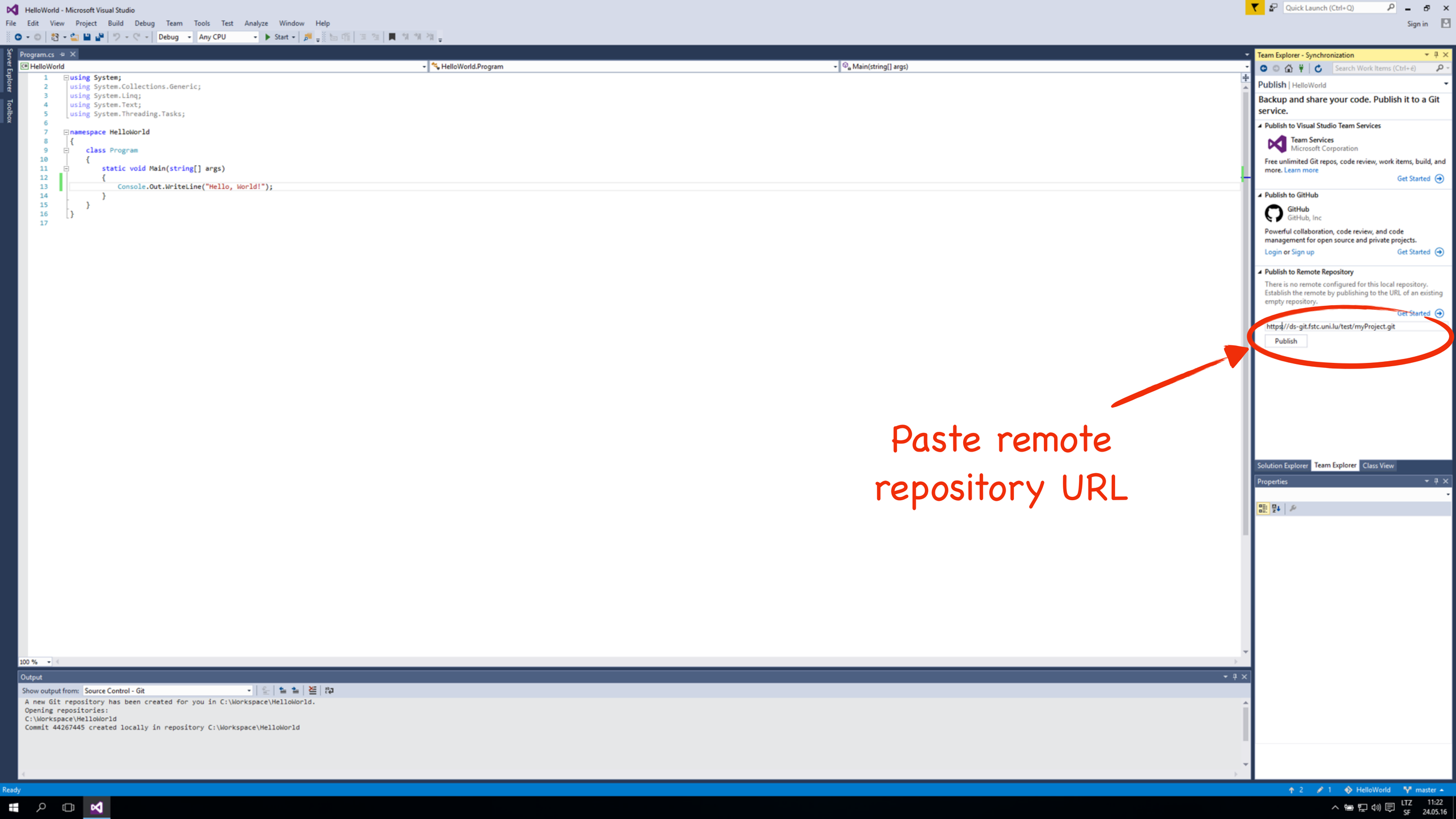


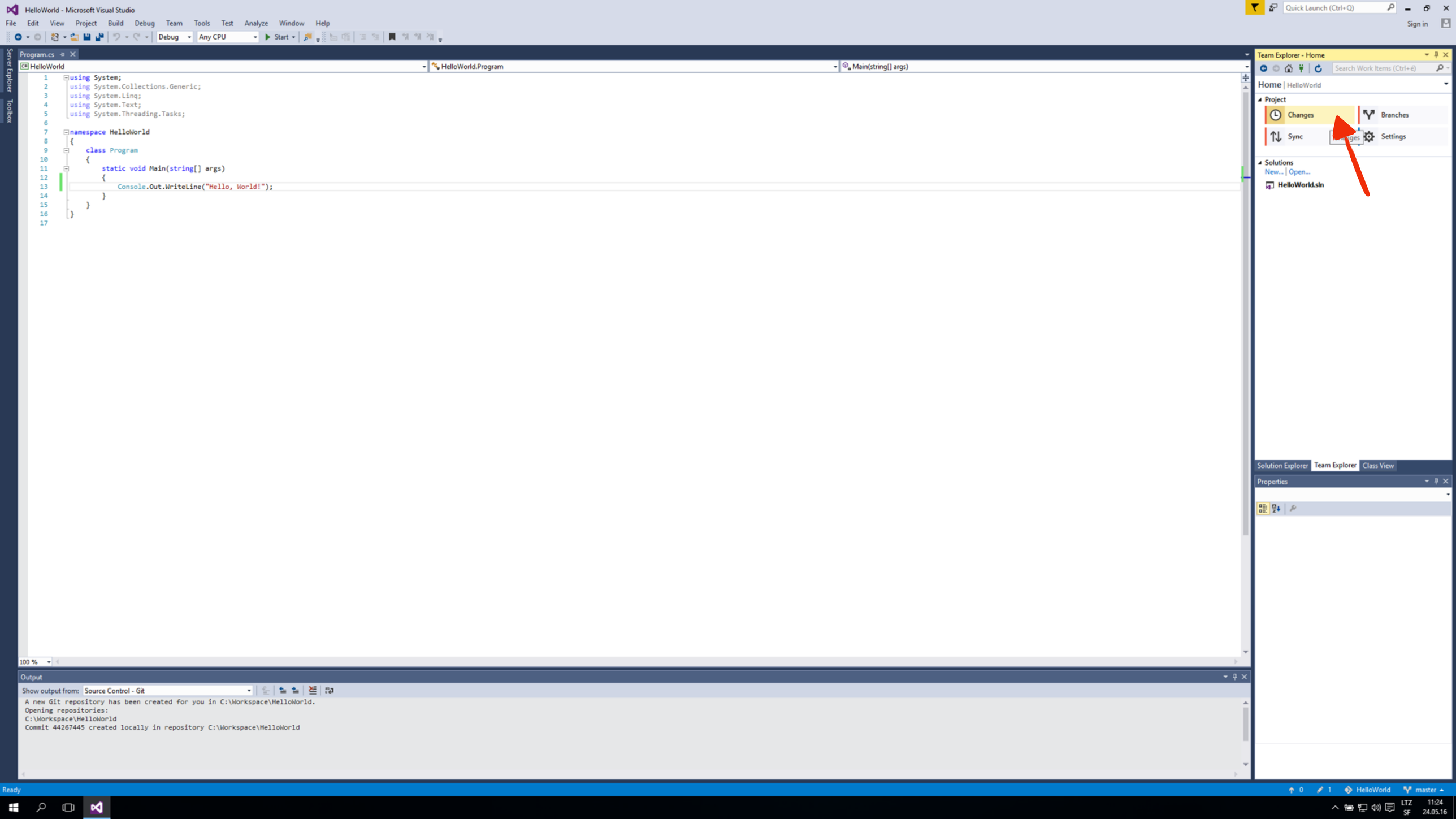

HelloWorld - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Manage Connections...
Program.cs
HelloWorld
HelloWorld.Program
Main(string[] args)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HelloWorld
{
class Program
{
static void Main(string[] args)
{
Console.Out.WriteLine("Hello, World!");
}
}
}
Class View
<Search>
HelloWorld
Solution Explorer
Class View
Properties
Output
Show output from: Source Control - Git
A new Git repository has been created for you in C:\Workspace\HelloWorld.
Opening repositories:
C:\Workspace\HelloWorld
Commit 44267445 created locally in repository C:\Workspace\HelloWorld
Item(s) Saved
Ln 13
Col 49
Ch 49
INS
HelloWorld
master

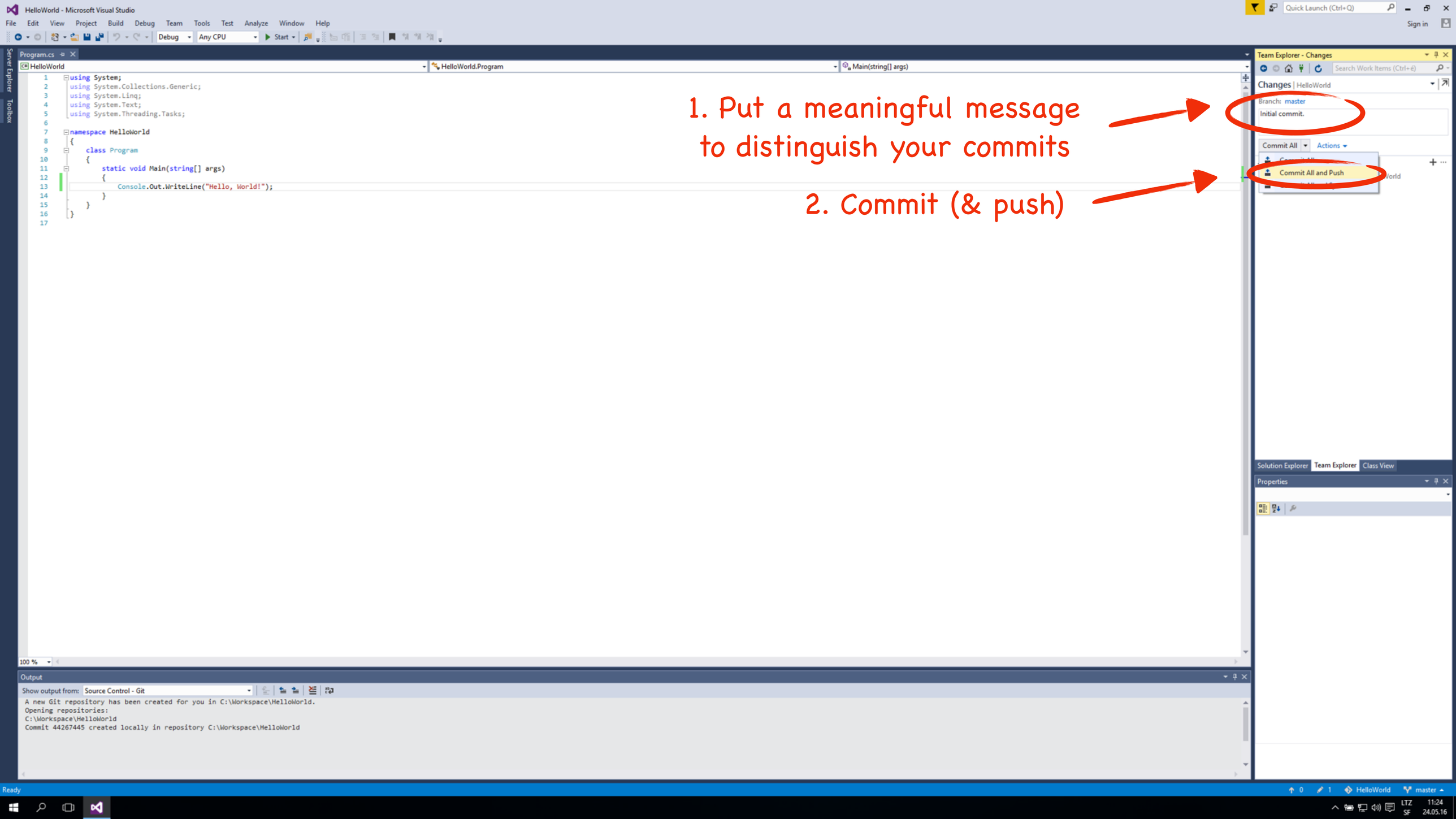

HelloWorld - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Quick Launch (Ctrl+Q)
Sign in
Debug
Any CPU
Start
Program.cs
HelloWorld
HelloWorld.Program
Main(string[] args)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HelloWorld
{
class Program
{
static void Main(string[] args)
{
Console.Out.WriteLine("Hello, World!");
}
}
}
Team Explorer - Connect
Search Work Items (Ctrl+è)
Connect | HelloWorld
Manage Connections
Hosted Service Providers
Visual Studio Team Services
Microsoft Corporation
Services to help you ship high quality software. On time, every time. Focus on your code. We'll simplify the rest.
Connect...
Get started for free
GitHub
GitHub, Inc.
Powerful collaboration, code review, and code management for open source and private projects.
Connect...
Sign up
Local Git Repositories (1)
New | Add | Clone | View Options
HelloWorld | C:\Workspace\HelloWorld
Solution Explorer
Team Explorer
Class View
Properties
100 %
Output
Show output from: Source Control - Git
A new Git repository has been created for you in C:\Workspace\HelloWorld.
Opening repositories:
C:\Workspace\HelloWorld
Commit 44267445 created locally in repository C:\Workspace\HelloWorld
Ready
HelloWorld
master
LTZ
SF
11:21
24.05.16

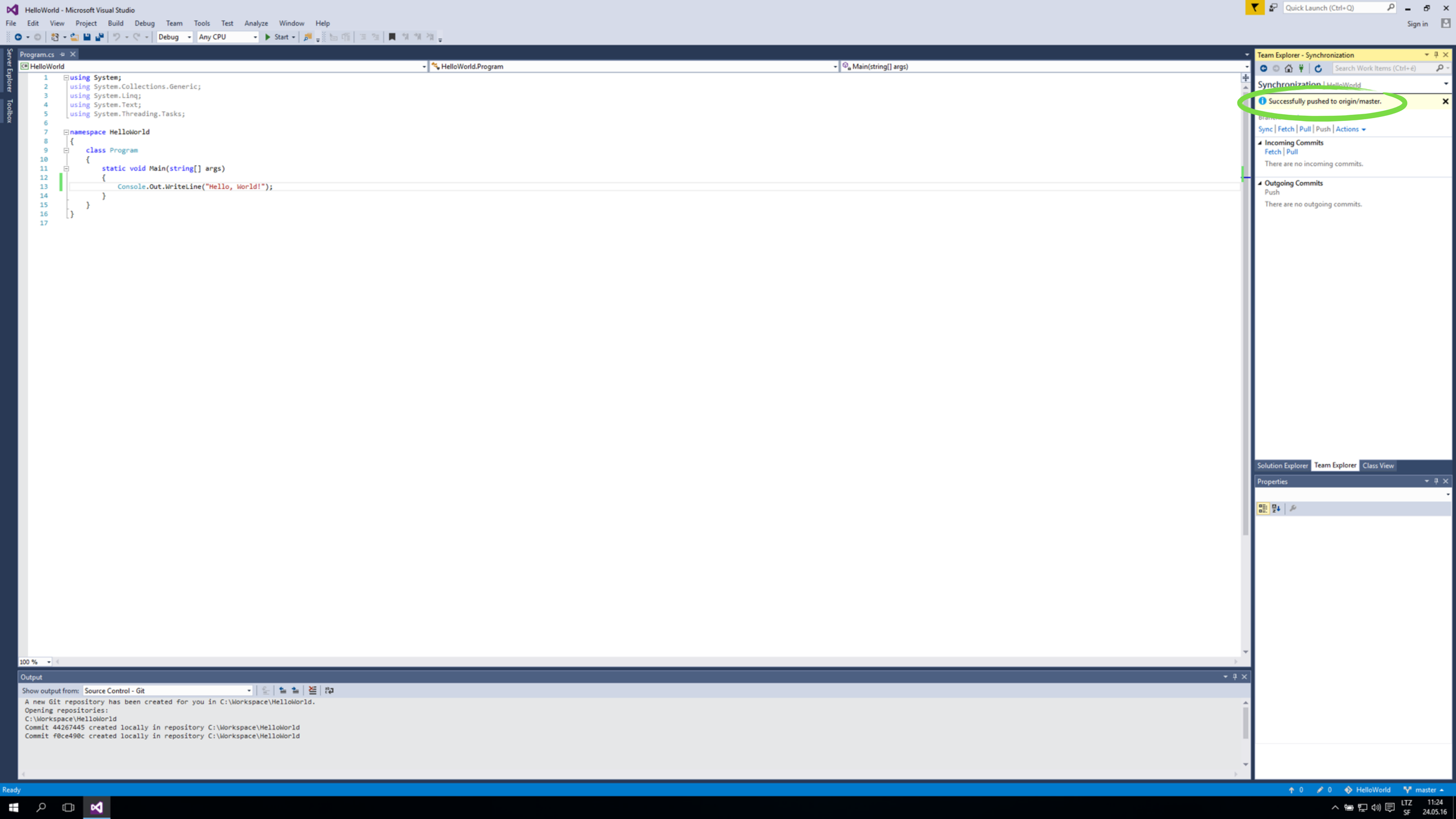

HelloWorld - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Quick Launch (Ctrl+Q)
Sign in
Debug
Any CPU
Start
Program.cs
HelloWorld
HelloWorld.Program
Main(string[] args)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HelloWorld
{
class Program
{
static void Main(string[] args)
{
Console.Out.WriteLine("Hello, World!");
}
}
}
Team Explorer - Home
Search Work Items (Ctrl+è)
Home | HelloWorld
Project
Changes
Branches
Sync
Settings
Sync
Solutions
New... | Open...
HelloWorld.sln
Solution Explorer
Team Explorer
Class View
Properties
100 %
Output
Show output from: Source Control - Git
A new Git repository has been created for you in C:\Workspace\HelloWorld.
Opening repositories:
C:\Workspace\HelloWorld
Commit 44267445 created locally in repository C:\Workspace\HelloWorld
Ready
HelloWorld
master
LTZ
SF
11:21
24.05.16


HelloWorld - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Quick Launch (Ctrl+Q)
Sign in
Debug
Any CPU
Start
Program.cs
HelloWorld
HelloWorld.Program
Main(string[] args)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HelloWorld
{
class Program
{
static void Main(string[] args)
{
Console.Out.WriteLine("Hello, World!");
}
}
}
Team Explorer - Synchronization
Search Work Items (Ctrl+è)
Publish | HelloWorld
Backup and share your code. Publish it to a Git service.
Publish to Visual Studio Team Services
Team Services
Microsoft Corporation
Free unlimited Git repos, code review, work items, build, and more. Learn more
Get Started
Publish to GitHub
GitHub
GitHub, Inc
Powerful collaboration, code review, and code management for open source and private projects.
Login or Sign up
Get Started
Publish to Remote Repository
There is no remote configured for this local repository. Establish the remote by publishing to the URL of an existing empty repository.
Get Started
Solution Explorer
Team Explorer
Class View
Properties
100 %
Output
Show output from: Source Control - Git
A new Git repository has been created for you in C:\Workspace\HelloWorld.
Opening repositories:
C:\Workspace\HelloWorld
Commit 44267445 created locally in repository C:\Workspace\HelloWorld
Ready
HelloWorld
master
LTZ
SF
11:21
24.05.16

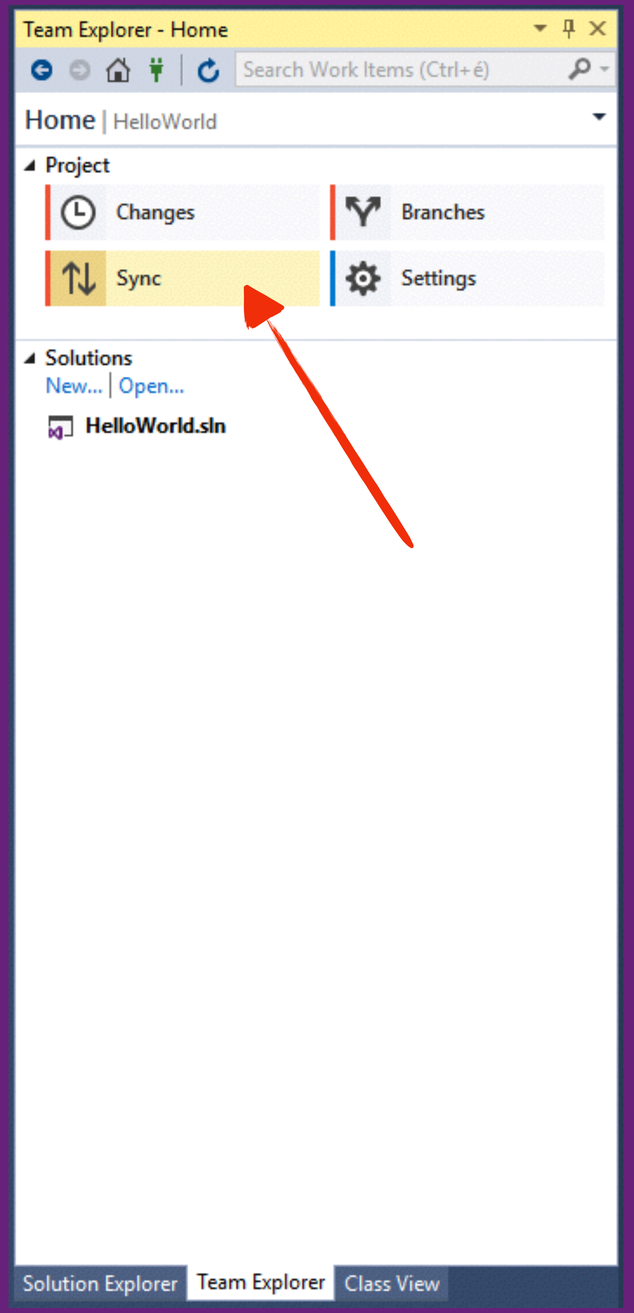

HelloWorld - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
Start
Program.cs
HelloWorld
HelloWorld.Program
Main(string[] args)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HelloWorld
{
class Program
{
static void Main(string[] args)
{
Console.Out.WriteLine("Hello, World!");
}
}
}
Team Explorer - Synchronization
Publish | HelloWorld
Backup and share your code. Publish it to a Git service.
Publish to Visual Studio Team Services
Team Services
Microsoft Corporation
Free unlimited Git repos, code review, work items, build, and more. Learn more
Get Started
Publish to GitHub
GitHub
GitHub, Inc
Powerful collaboration, code review, and code management for open source and private projects.
Login or Sign up
Get Started
Publish to Remote Repository
There is no remote configured for this local repository. Establish the remote by publishing to the URL of an existing empty repository.
Get Started
https://ds-git.fstc.uni.lu/test/myProject.git
Publish
Solution Explorer
Team Explorer
Class View
Properties
Output
Show output from: Source Control - Git
A new Git repository has been created for you in C:\Workspace\HelloWorld.
Opening repositories:
C:\Workspace\HelloWorld
Commit 44267445 created locally in repository C:\Workspace\HelloWorld
Ready
HelloWorld
master
Paste remote repository URL

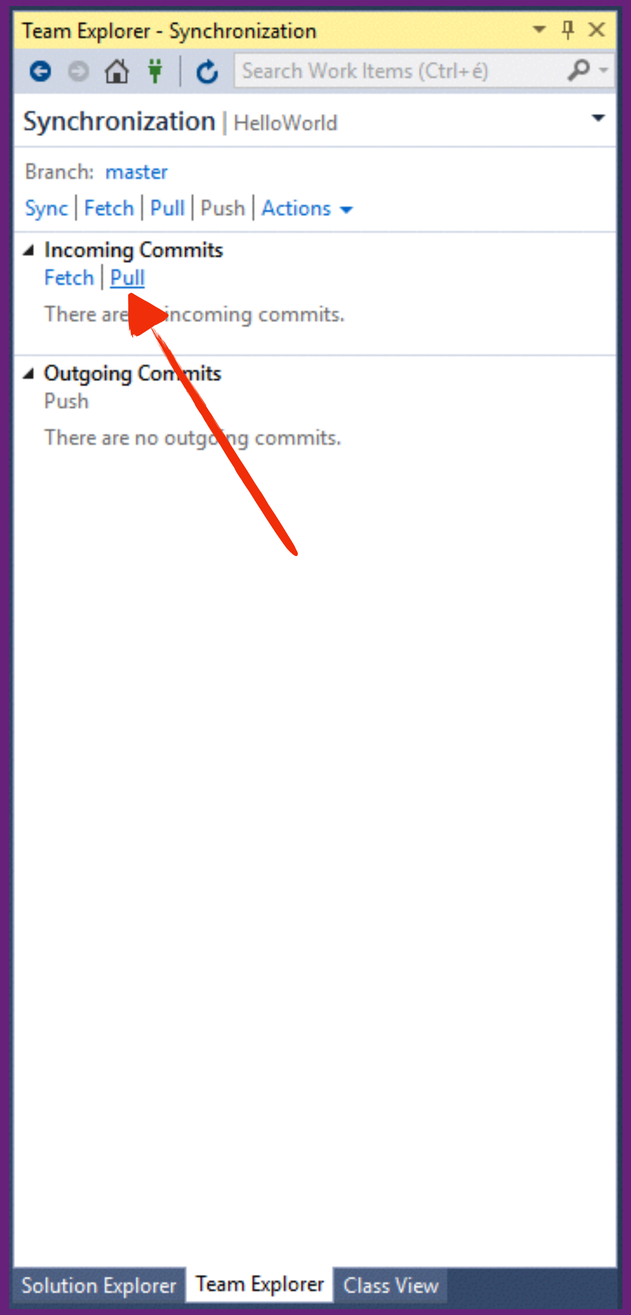

HelloWorld - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
Start
Program.cs
HelloWorld
HelloWorld.Program
Main(string[] args)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HelloWorld
{
class Program
{
static void Main(string[] args)
{
Console.Out.WriteLine("Hello, World!");
}
}
}
100 %
Output
Show output from: Source Control - Git
A new Git repository has been created for you in C:\Workspace\HelloWorld.
Opening repositories:
C:\Workspace\HelloWorld
Commit 44267445 created locally in repository C:\Workspace\HelloWorld
Team Explorer - Home
Search Work Items (Ctrl+è)
Home | HelloWorld
Project
Changes
Branches
Sync
Settings
Solutions
New... | Open...
HelloWorld.sln
Solution Explorer
Team Explorer
Class View
Properties
Ready
HelloWorld
master
24.05.16


HelloWorld - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug
Any CPU
Start
Program.cs
HelloWorld
HelloWorld.Program
Main(string[] args)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HelloWorld
{
class Program
{
static void Main(string[] args)
{
Console.Out.WriteLine("Hello, World!");
}
}
}
1. Put a meaningful message to distinguish your commits
2. Commit (& push)
Team Explorer - Changes
Changes | HelloWorld
Branch: master
Initial commit.
Commit All
Actions
Commit All and Push
Solution Explorer
Team Explorer
Class View
Properties
Output
Show output from: Source Control - Git
A new Git repository has been created for you in C:\Workspace\HelloWorld.
Opening repositories:
C:\Workspace\HelloWorld
Commit 44267445 created locally in repository C:\Workspace\HelloWorld
Ready
HelloWorld
master

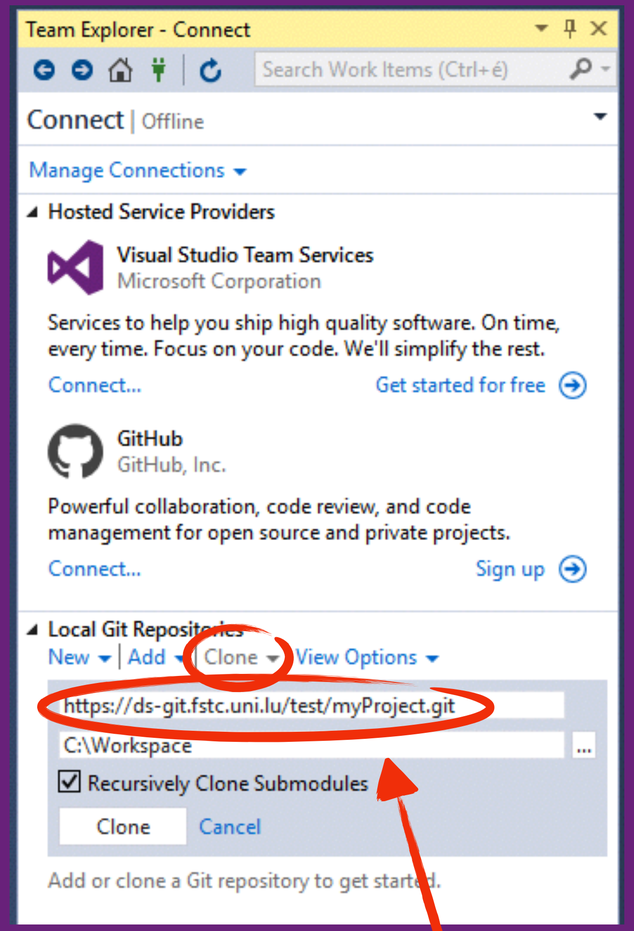

HelloWorld - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Quick Launch (Ctrl+Q)
Sign in
Debug
Any CPU
Start
Program.cs
HelloWorld
HelloWorld.Program
Main(string[] args)
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HelloWorld
{
class Program
{
static void Main(string[] args)
{
Console.Out.WriteLine("Hello, World!");
}
}
}
Team Explorer - Synchronization
Search Work Items (Ctrl+è)
Synchronization
Successfully pushed to origin/master.
Sync | Fetch | Pull | Push | Actions
Incoming Commits
Fetch | Pull
There are no incoming commits.
Outgoing Commits
Push
There are no outgoing commits.
Solution Explorer
Team Explorer
Class View
Properties
100 %
Output
Show output from: Source Control - Git
A new Git repository has been created for you in C:\Workspace\HelloWorld.
Opening repositories:
C:\Workspace\HelloWorld
Commit 44267445 created locally in repository C:\Workspace\HelloWorld
Commit f0ce490c created locally in repository C:\Workspace\HelloWorld
Ready
HelloWorld
master
LTZ
SF
11:24
24.05.16

Pull current version

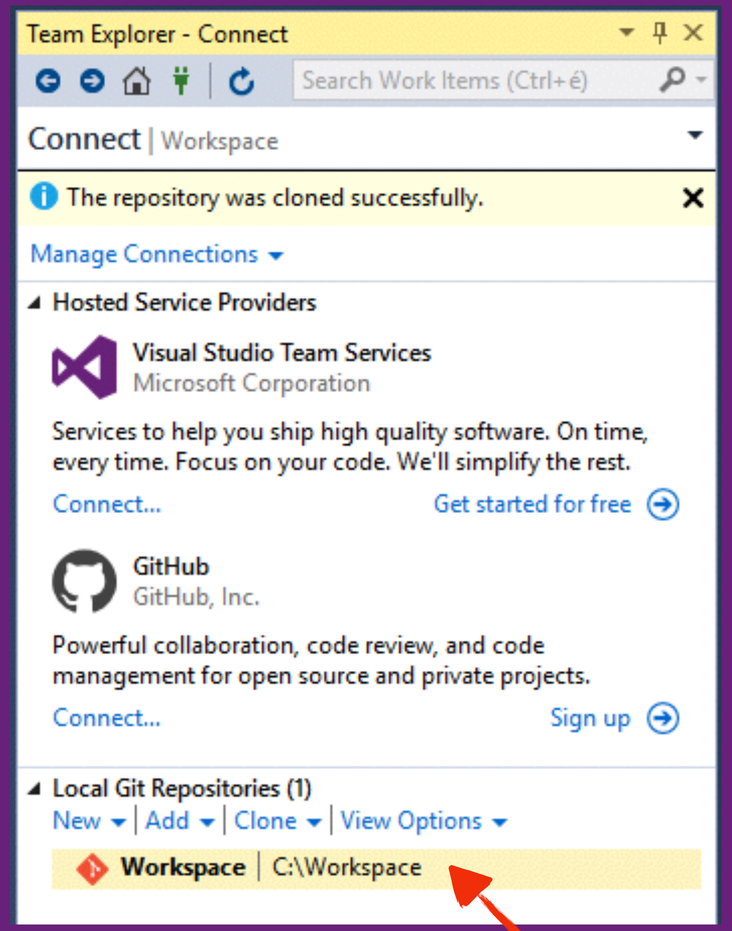


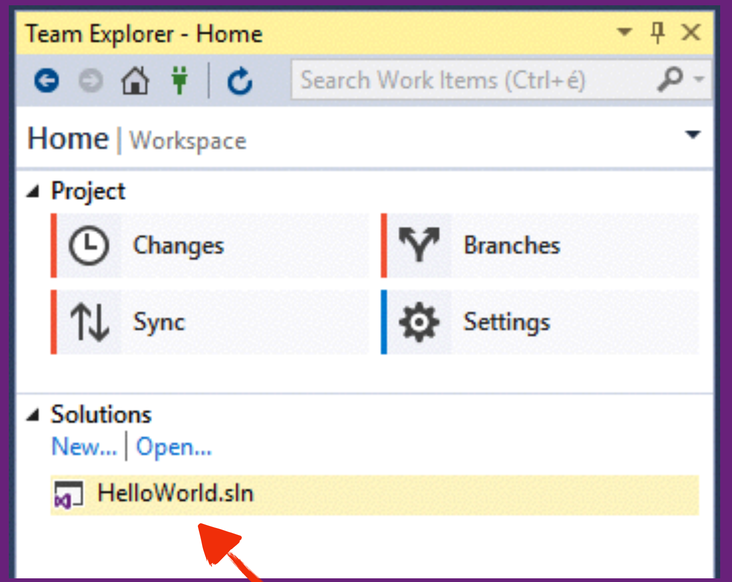


Clone existing project



Paste remote repository URL

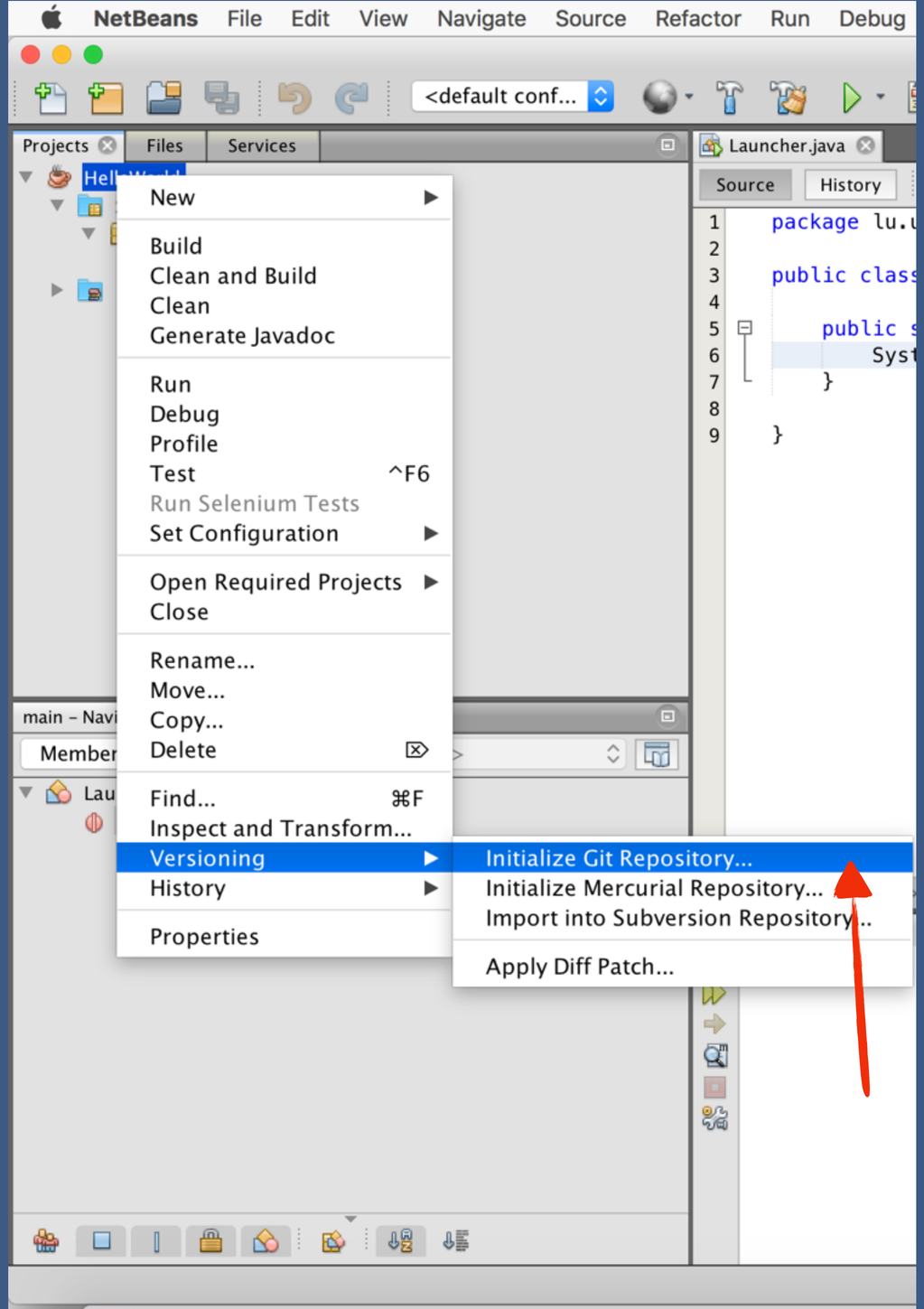


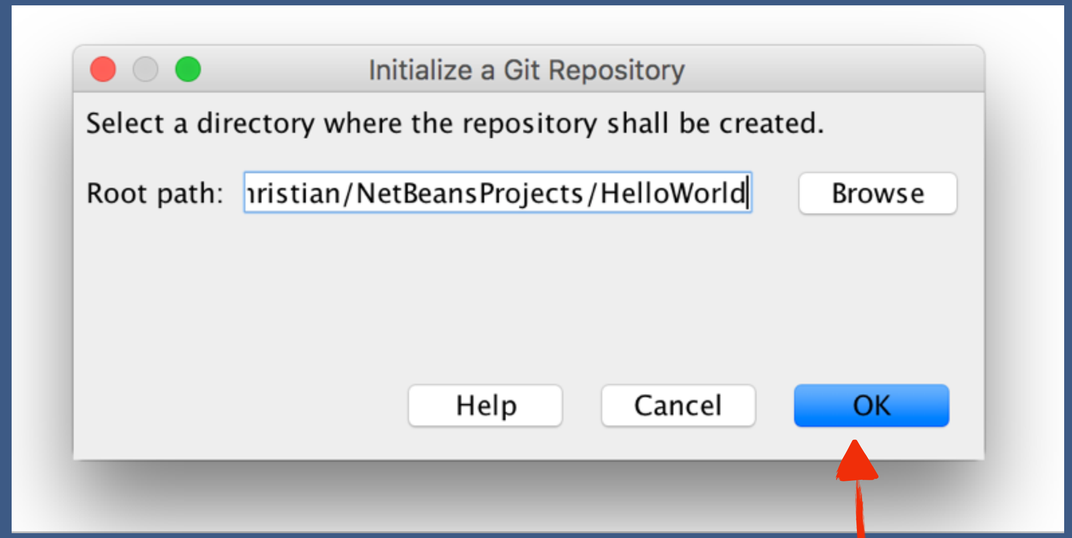


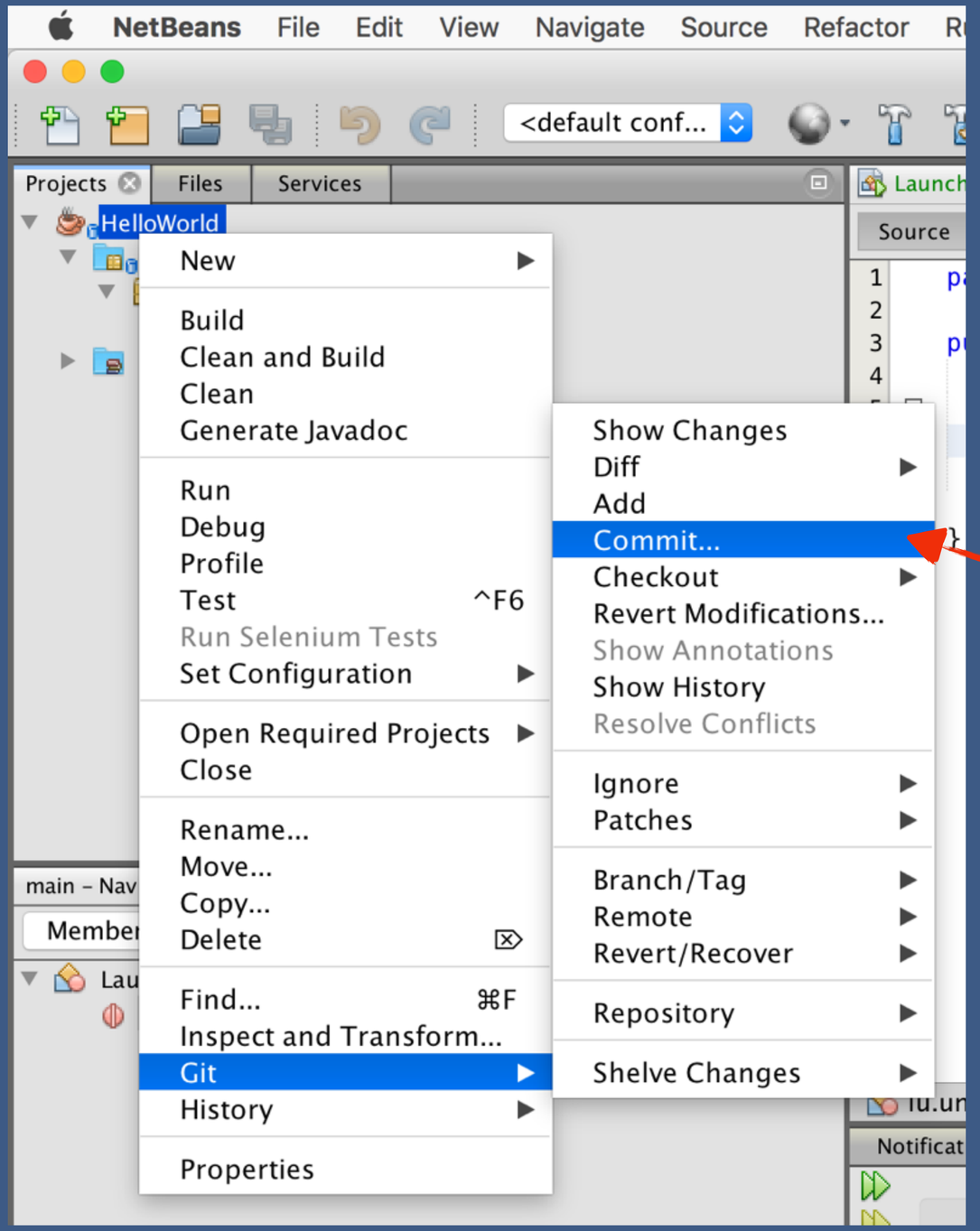
Open Solution

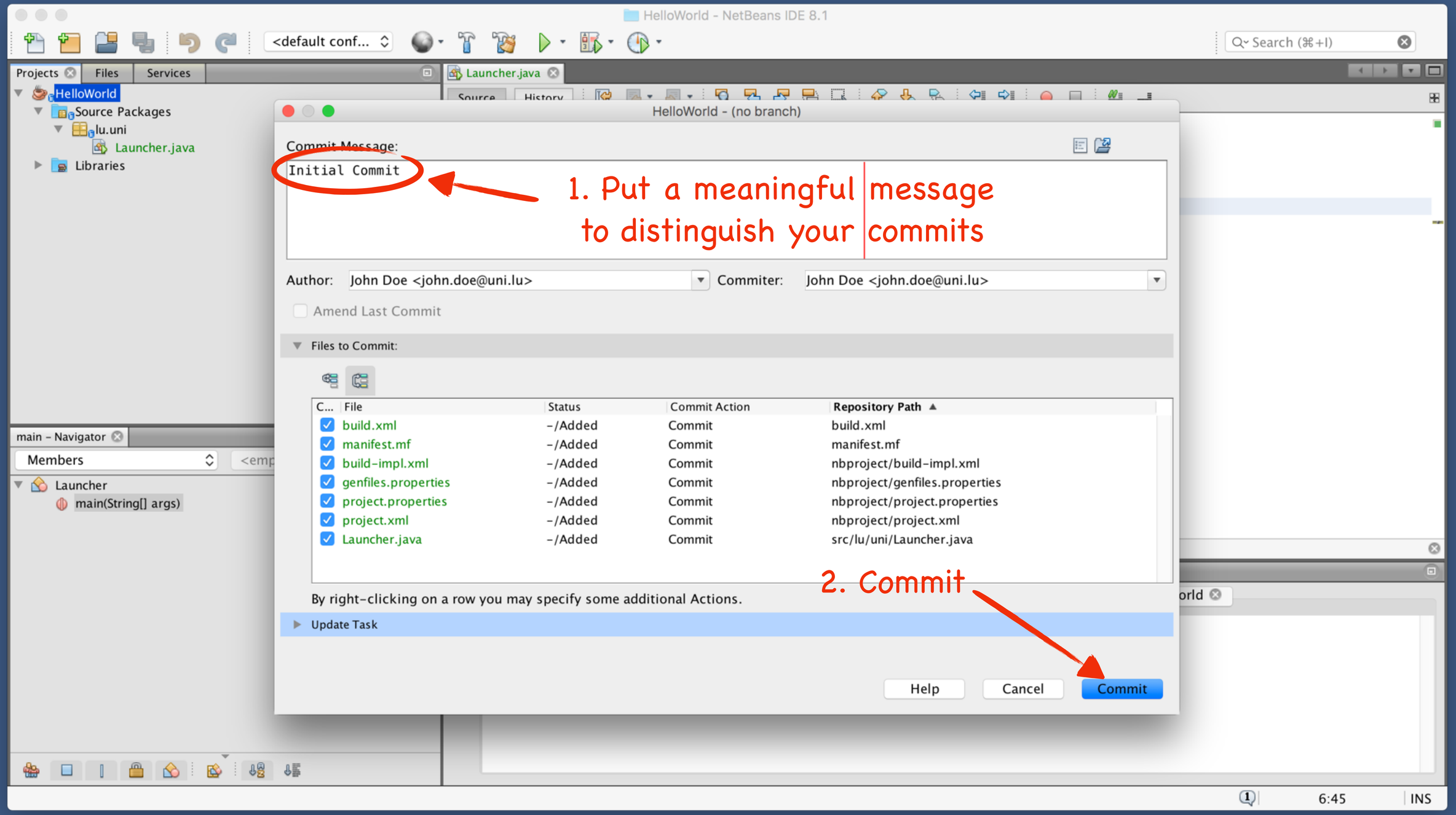
NetBeans

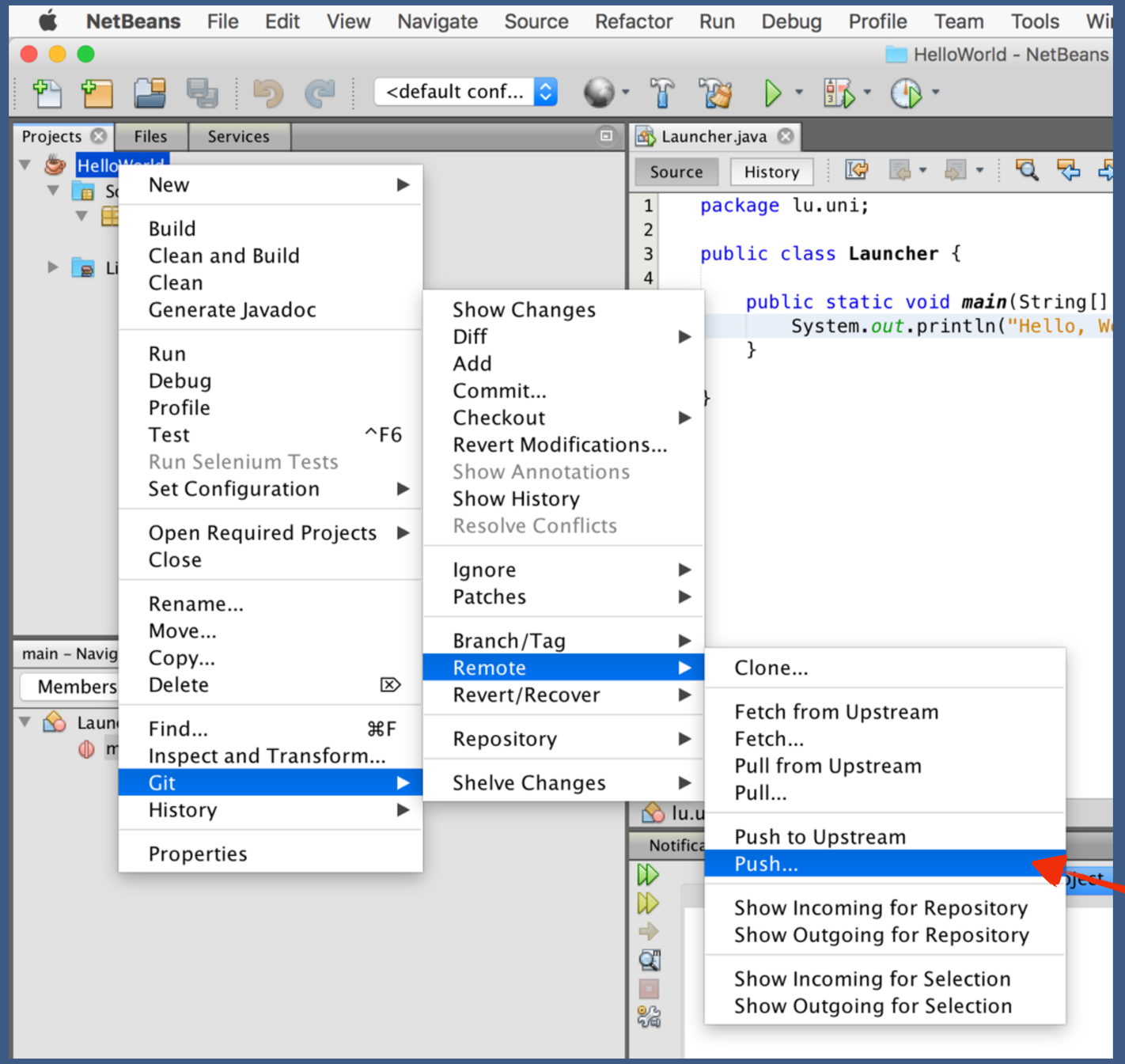
Initial repository setup

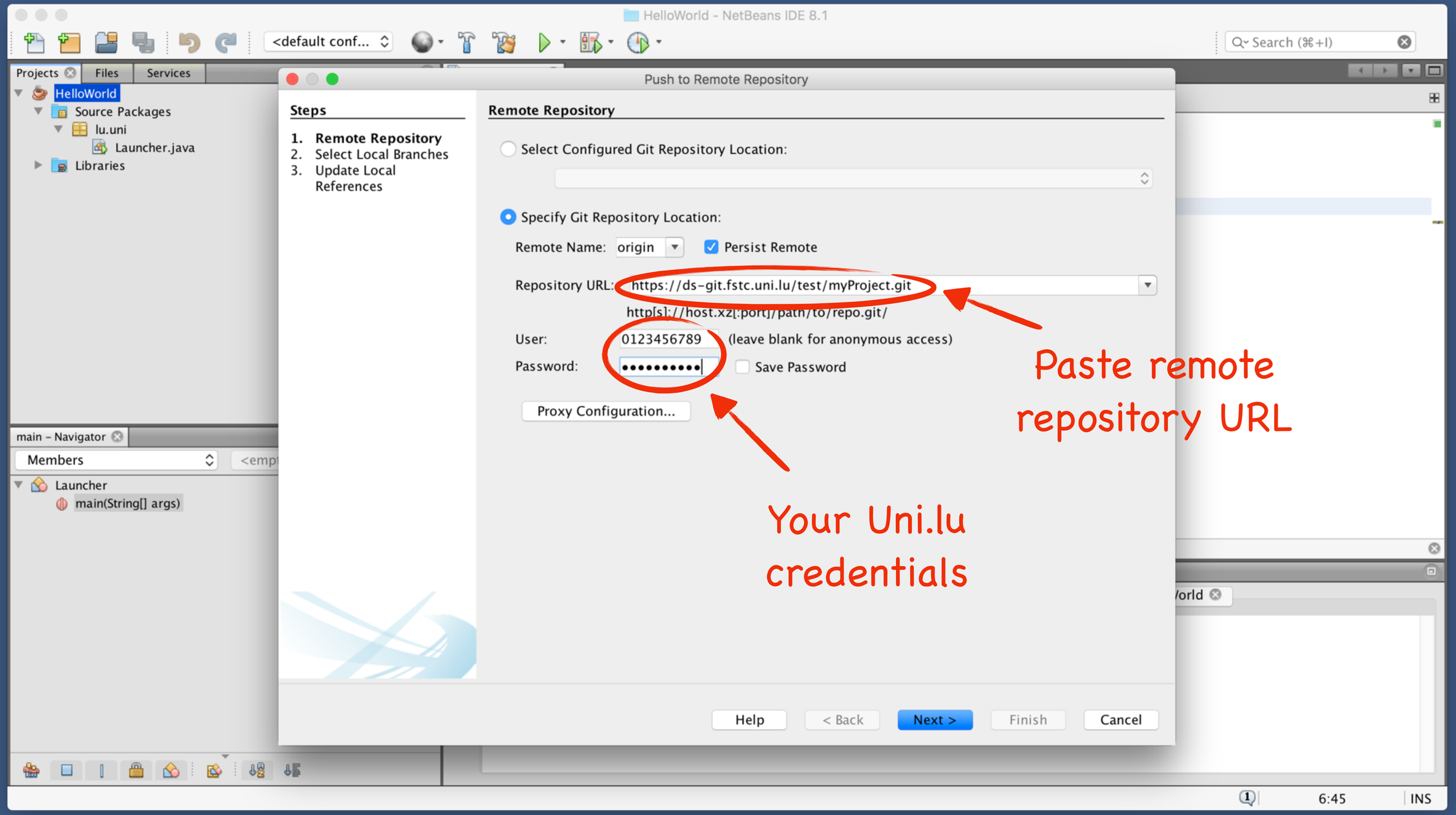


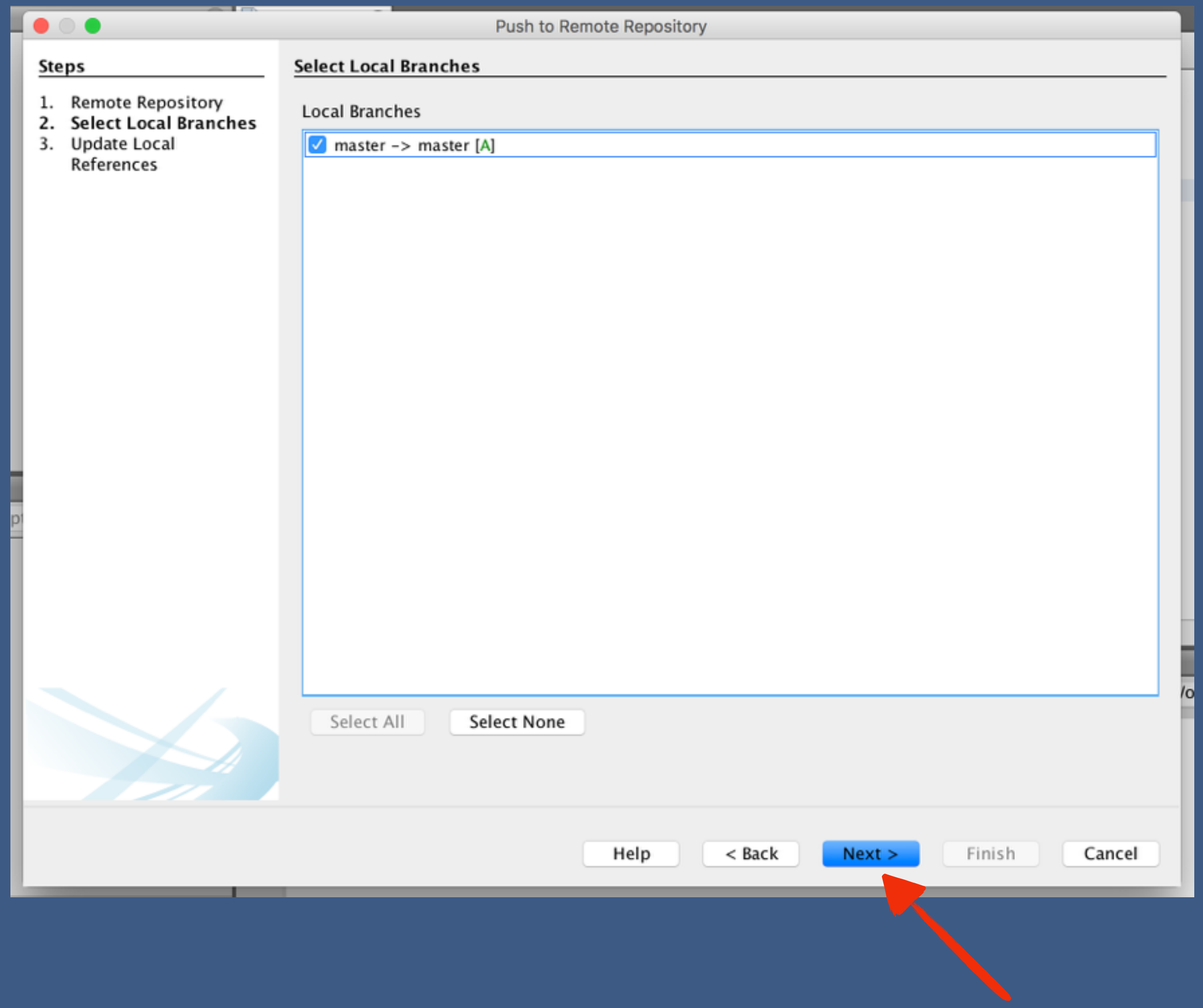


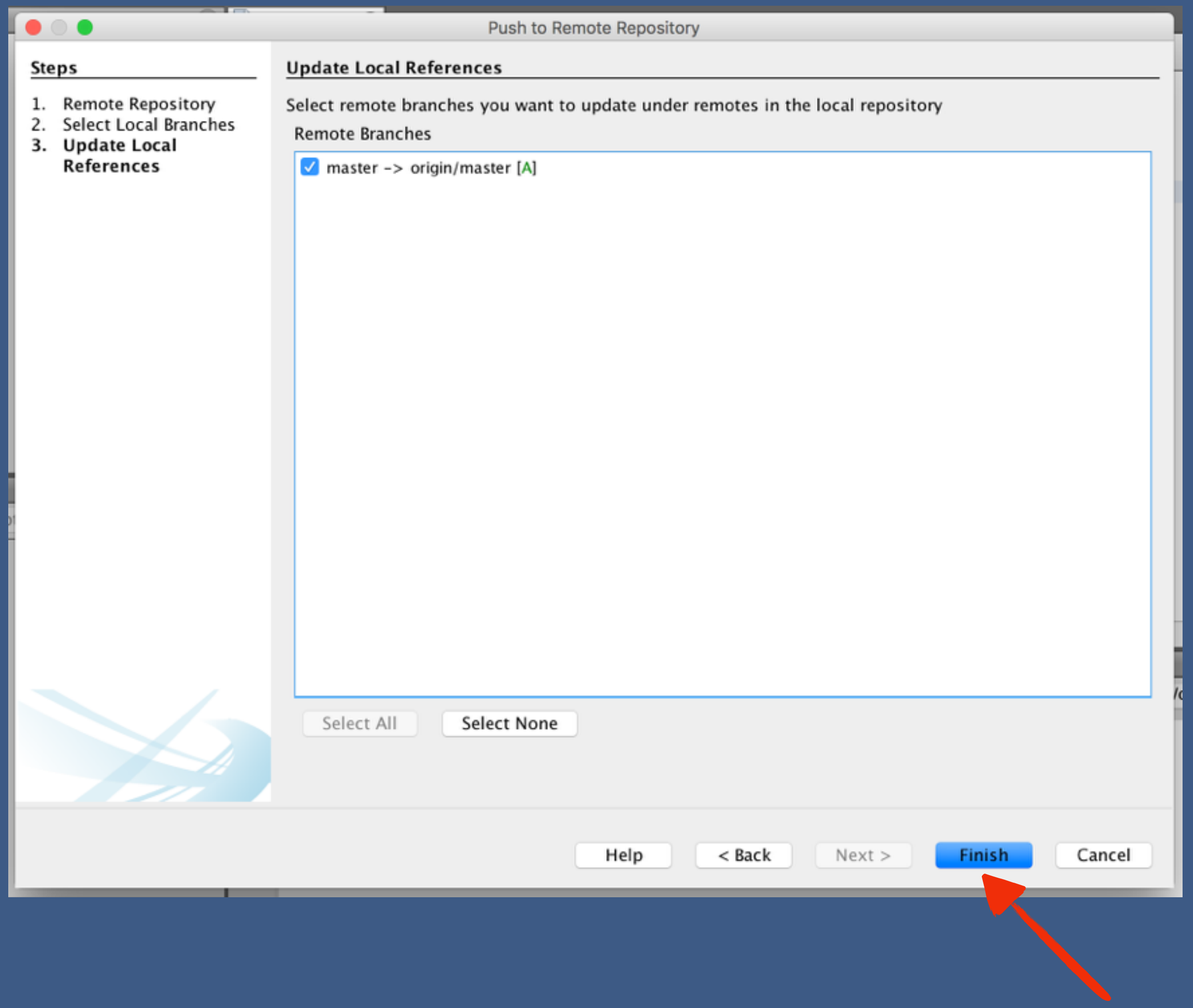

NetBeans
File
Edit
View
Navigate
Source
Refactor
<default conf...
Projects
Files
Services
HelloWorld
New
Build
Clean and Build
Clean
Generate Javadoc
Run
Debug
Profile
Test ^F6
Run Selenium Tests
Set Configuration
Open Required Projects
Close
Rename...
Move...
Copy...
Delete
Find... ⌘F
Inspect and Transform...
Git
History
Properties
Show Changes
Diff
Add
Commit...
Checkout
Revert Modifications...
Show Annotations
Show History
Resolve Conflicts
Ignore
Patches
Branch/Tag
Remote
Revert/Recover
Repository
Shelve Changes
Source
main – Nav
Member
Notificat


HelloWorld - NetBeans IDE 8.1
Projects
Files
Services
HelloWorld
Source Packages
lu.uni
Launcher.java
Libraries
Launcher.java
HelloWorld - (no branch)
Commit Message:
Initial Commit
1. Put a meaningful message to distinguish your commits
Author: John Doe <john.doe@uni.lu>
Commiter: John Doe <john.doe@uni.lu>
Amend Last Commit
Files to Commit:
C... File Status Commit Action Repository Path
build.xml -/Added Commit build.xml
manifest.mf -/Added Commit manifest.mf
build-impl.xml -/Added Commit nbproject/build-impl.xml
genfiles.properties -/Added Commit nbproject/genfiles.properties
project.properties -/Added Commit nbproject/project.properties
project.xml -/Added Commit nbproject/project.xml
Launcher.java -/Added Commit src/lu/uni/Launcher.java
By right-clicking on a row you may specify some additional Actions.
Update Task
2. Commit
Help
Cancel
Commit
main - Navigator
Members
Launcher
main(String[] args)
6:45
INS

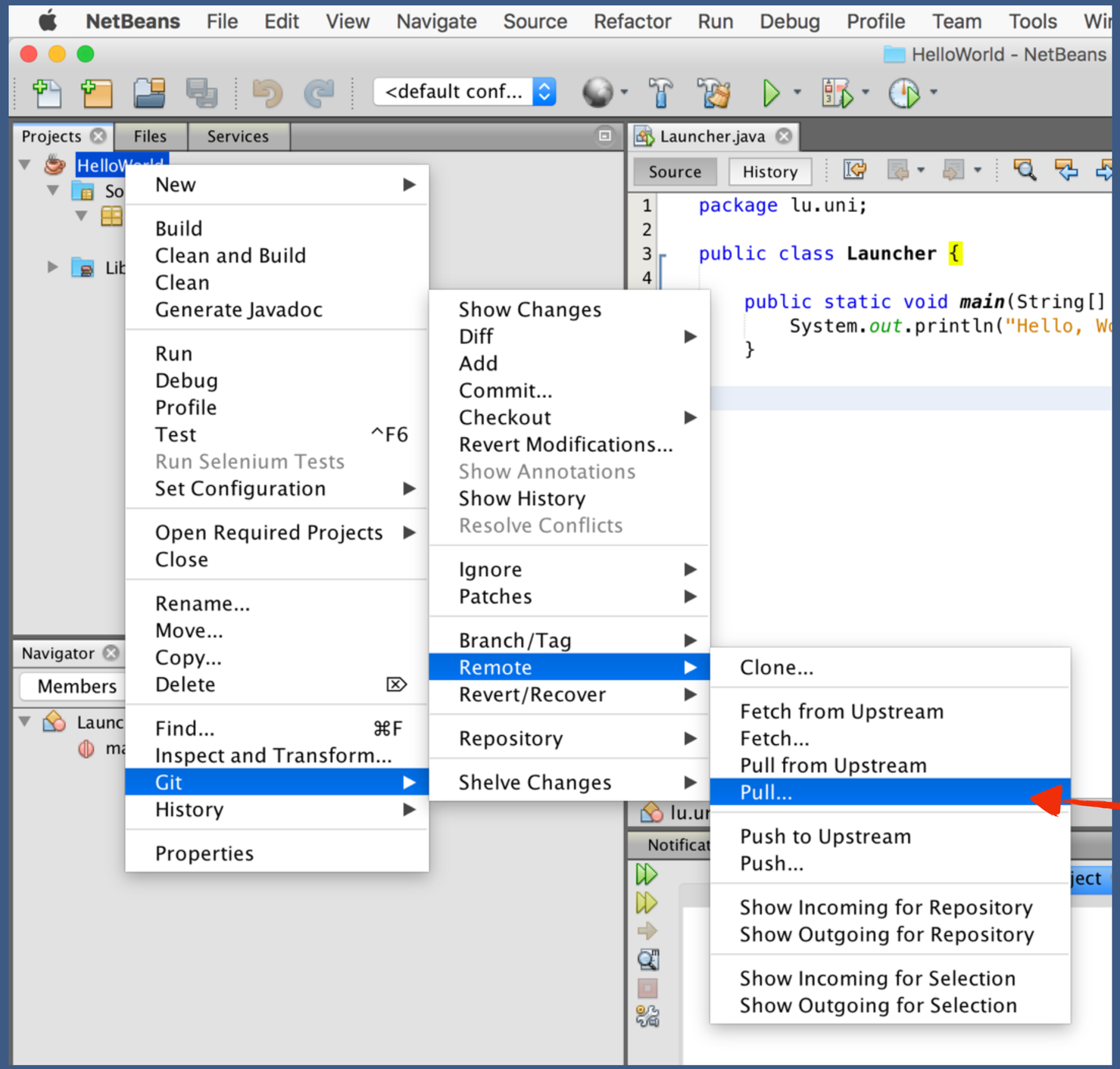

NetBeans File Edit View Navigate Source Refactor Run Debug Profile Team Tools
HelloWorld - NetBeans
<default conf...
Projects Files Services
Launcher.java
Source History
1 package lu.uni;
2
3 public class Launcher {
4
public static void main(String[]
System.out.println("Hello, W
}
}
New
Build
Clean and Build
Clean
Generate Javadoc
Run
Debug
Profile
Test ^F6
Run Selenium Tests
Set Configuration
Open Required Projects
Close
Rename...
Move...
Copy...
Delete
Find... ⌘F
Inspect and Transform...
Git
History
Properties
Show Changes
Diff
Add
Commit...
Checkout
Revert Modifications...
Show Annotations
Show History
Resolve Conflicts
Ignore
Patches
Branch/Tag
Remote
Revert/Recover
Repository
Shelve Changes
Clone...
Fetch from Upstream
Fetch...
Pull from Upstream
Pull...
Push to Upstream
Push...
Show Incoming for Repository
Show Outgoing for Repository
Show Incoming for Selection
Show Outgoing for Selection
main – Navig
Members
Notifica

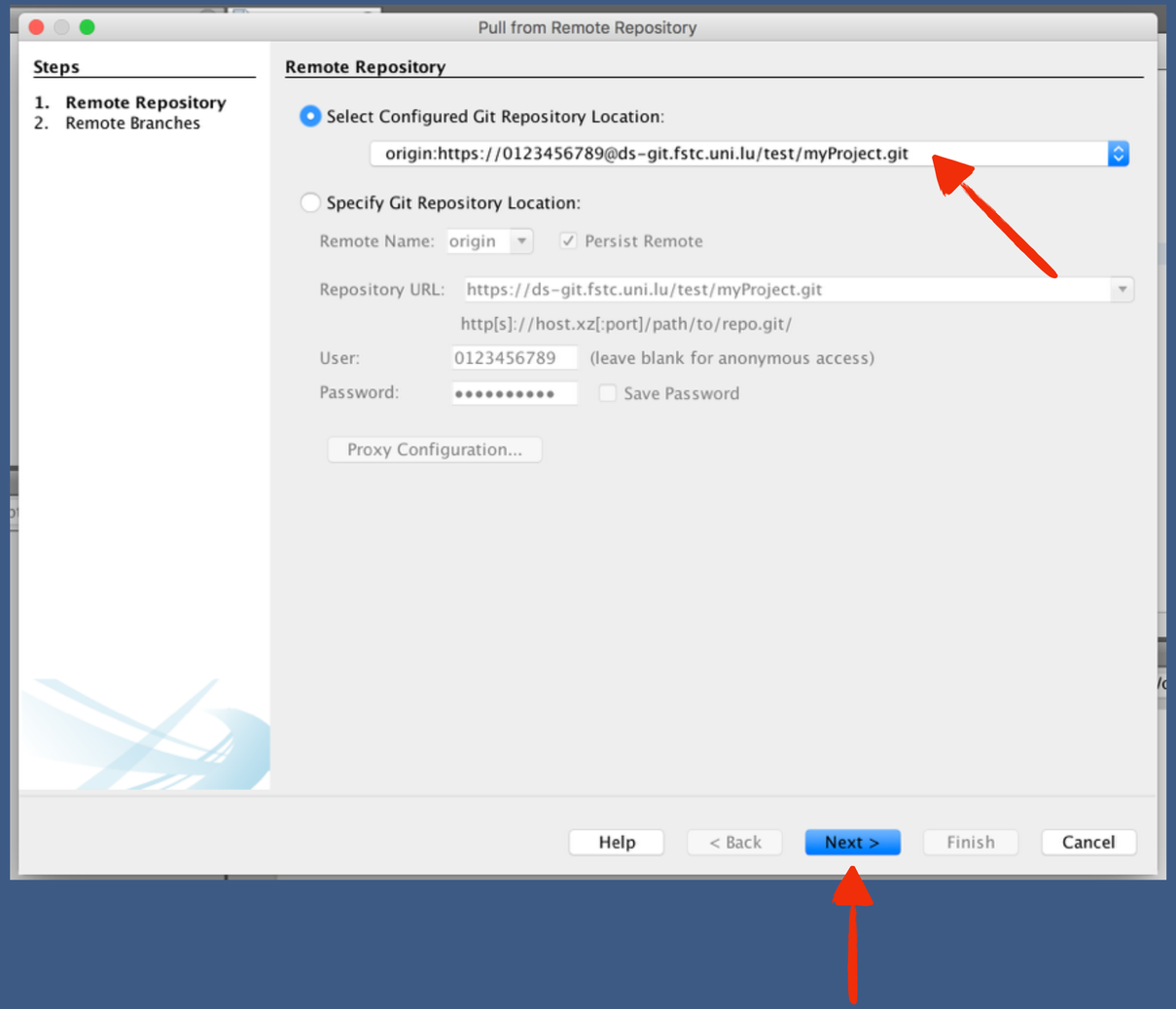

HelloWorld - NetBeans IDE 8.1
Search (⌘+I)
Projects
Files
Services
HelloWorld
Source Packages
lu.uni
Launcher.java
Libraries
main – Navigator
Members
Launcher
main(String[] args)
Push to Remote Repository
Steps
1. Remote Repository
2. Select Local Branches
3. Update Local References
Remote Repository
Select Configured Git Repository Location:
Specify Git Repository Location:
Remote Name: origin
Persist Remote
Repository URL: https://ds-git.fstc.uni.lu/test/myProject.git
http[s]://host.xz[:port]/path/to/repo.git/
User: 0123456789
(leave blank for anonymous access)
Password:
Save Password
Proxy Configuration...
Paste remote repository URL
Your Uni.lu credentials
Help
< Back
Next >
Finish
Cancel
6:45
INS

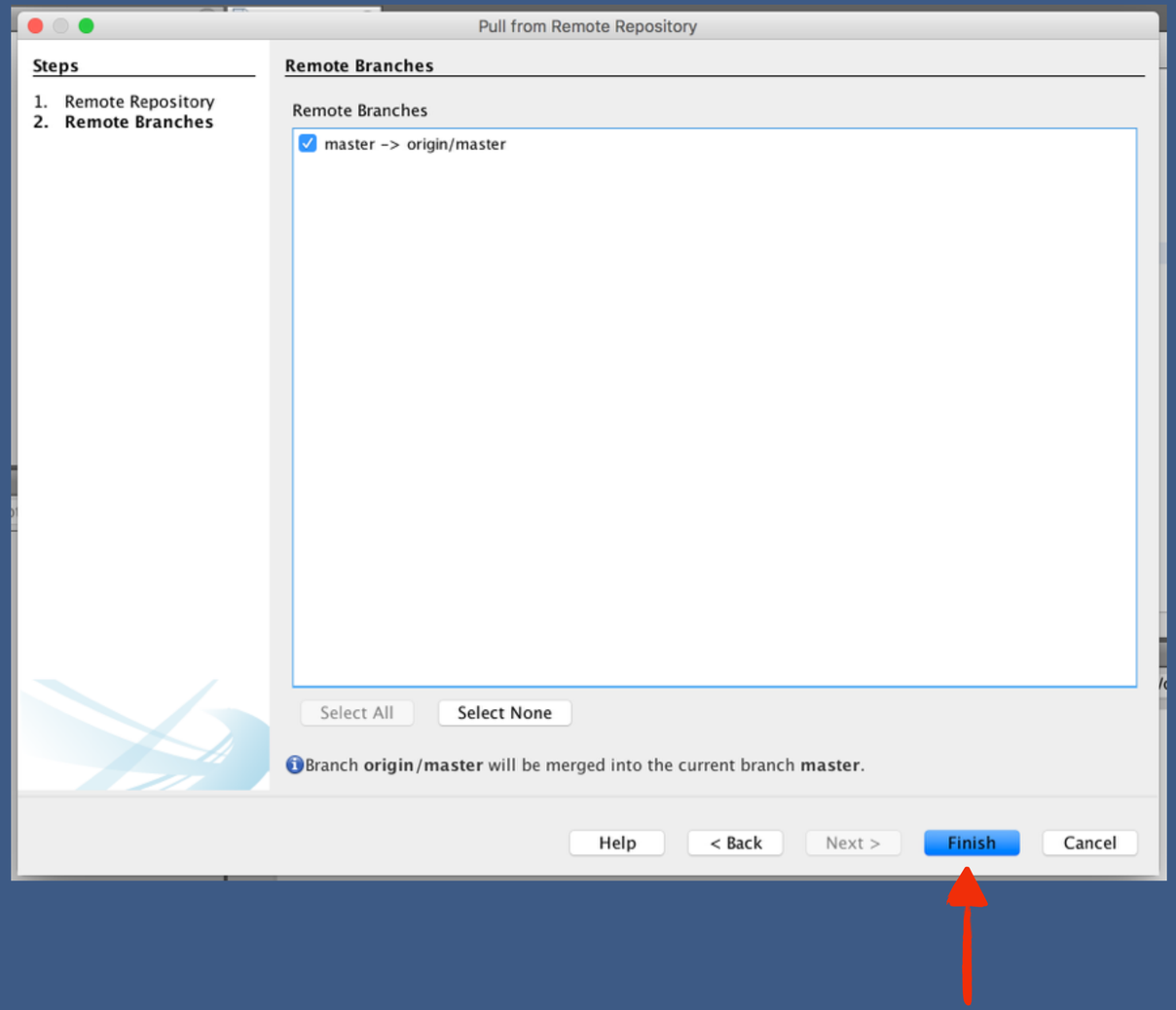

Push to Remote Repository
Steps
1. Remote Repository
2. Select Local Branches
3. Update Local References
Select Local Branches
Local Branches
master -> master [A]
Select All
Select None
Help
< Back
Next >
Finish
Cancel


Push to Remote Repository
Steps
1. Remote Repository
2. Select Local Branches
3. Update Local References
Update Local References
Select remote branches you want to update under remotes in the local repository
Remote Branches
master -> origin/master [A]
Select All
Select None
Help
< Back
Next >
Finish
Cancel

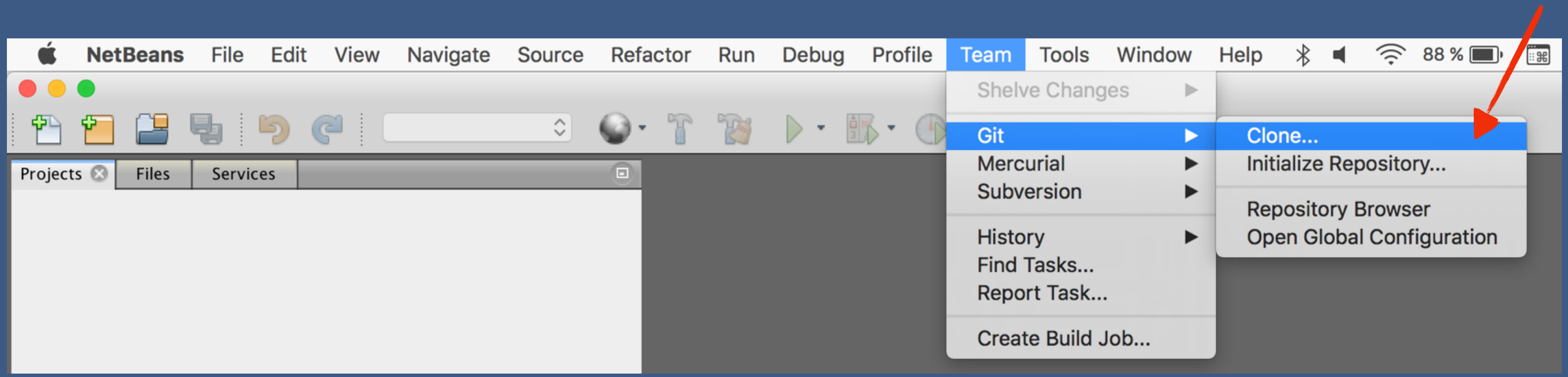
Pull current version

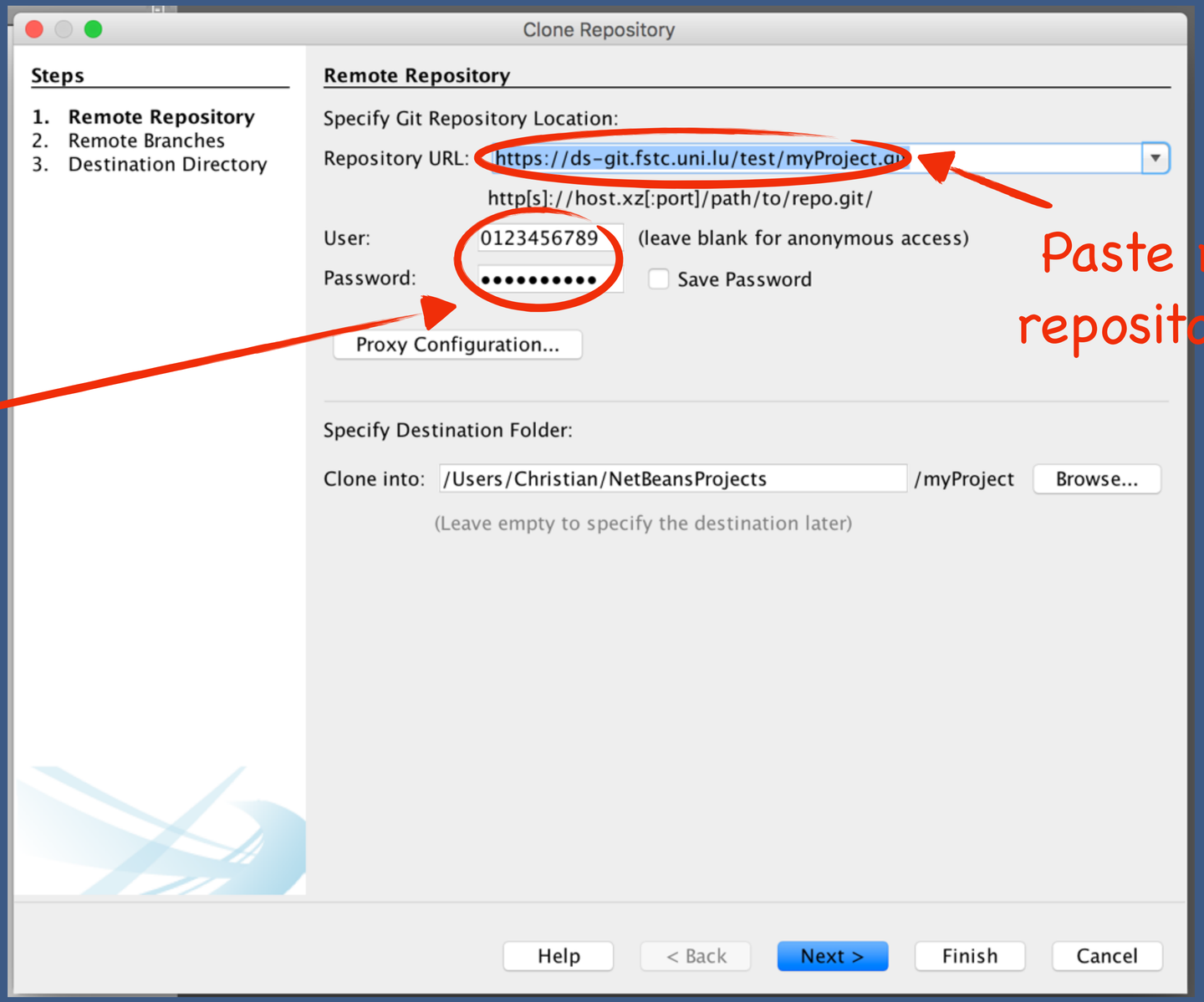



Pull from Remote Repository
Steps
1. Remote Repository
2. Remote Branches
Remote Repository
Select Configured Git Repository Location:
origin:https://0123456789@ds-git.fstc.uni.lu/test/myProject.git
Specify Git Repository Location:
Remote Name: origin
Persist Remote
Repository URL: https://ds-git.fstc.uni.lu/test/myProject.git
http[s]://host.xz[:port]/path/to/repo.git/
User: 0123456789
(leave blank for anonymous access)
Password:
Save Password
Proxy Configuration...
Help
< Back
Next >
Finish
Cancel


Pull from Remote Repository
Steps
1. Remote Repository
2. Remote Branches
Remote Branches
Remote Branches
master -> origin/master
Select All
Select None
Branch origin/master will be merged into the current branch master.
Help
< Back
Next >
Finish
Cancel

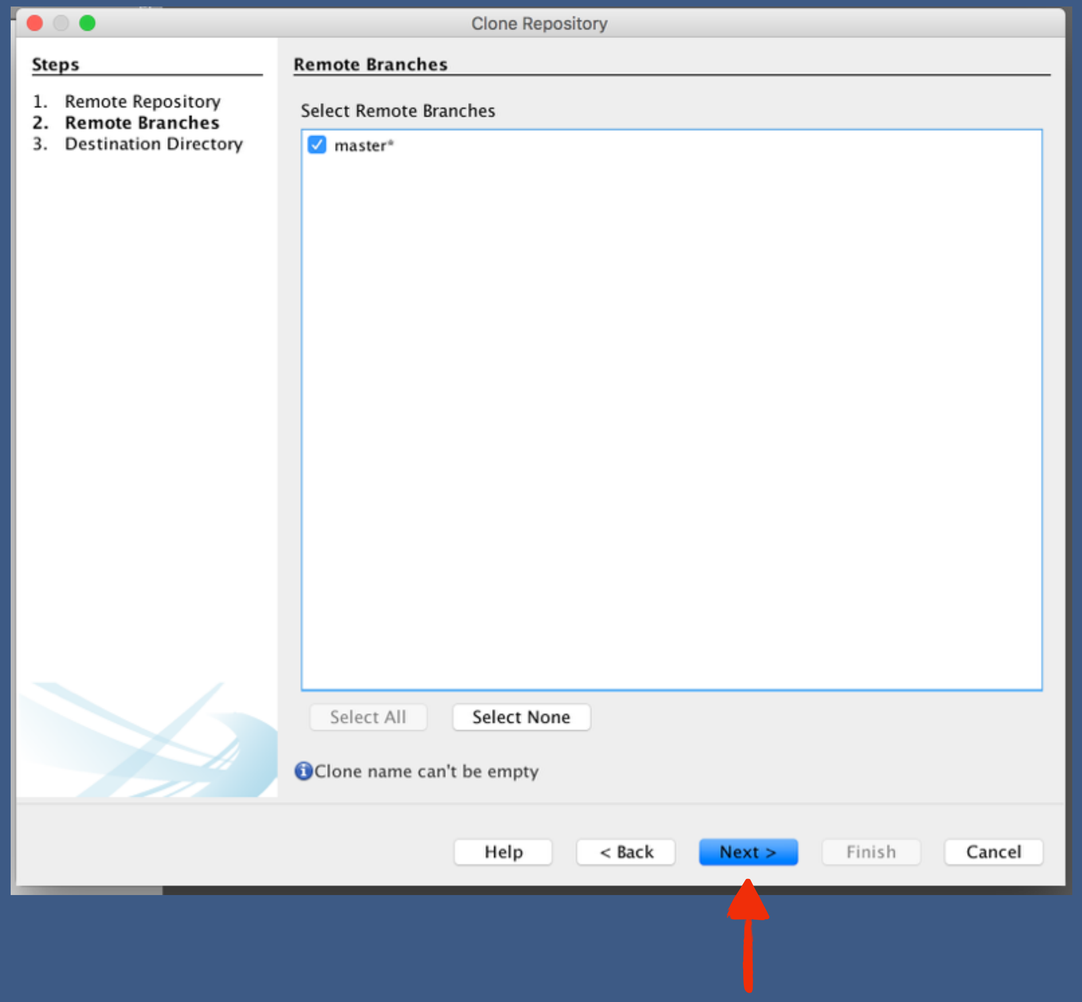
Clone existing project

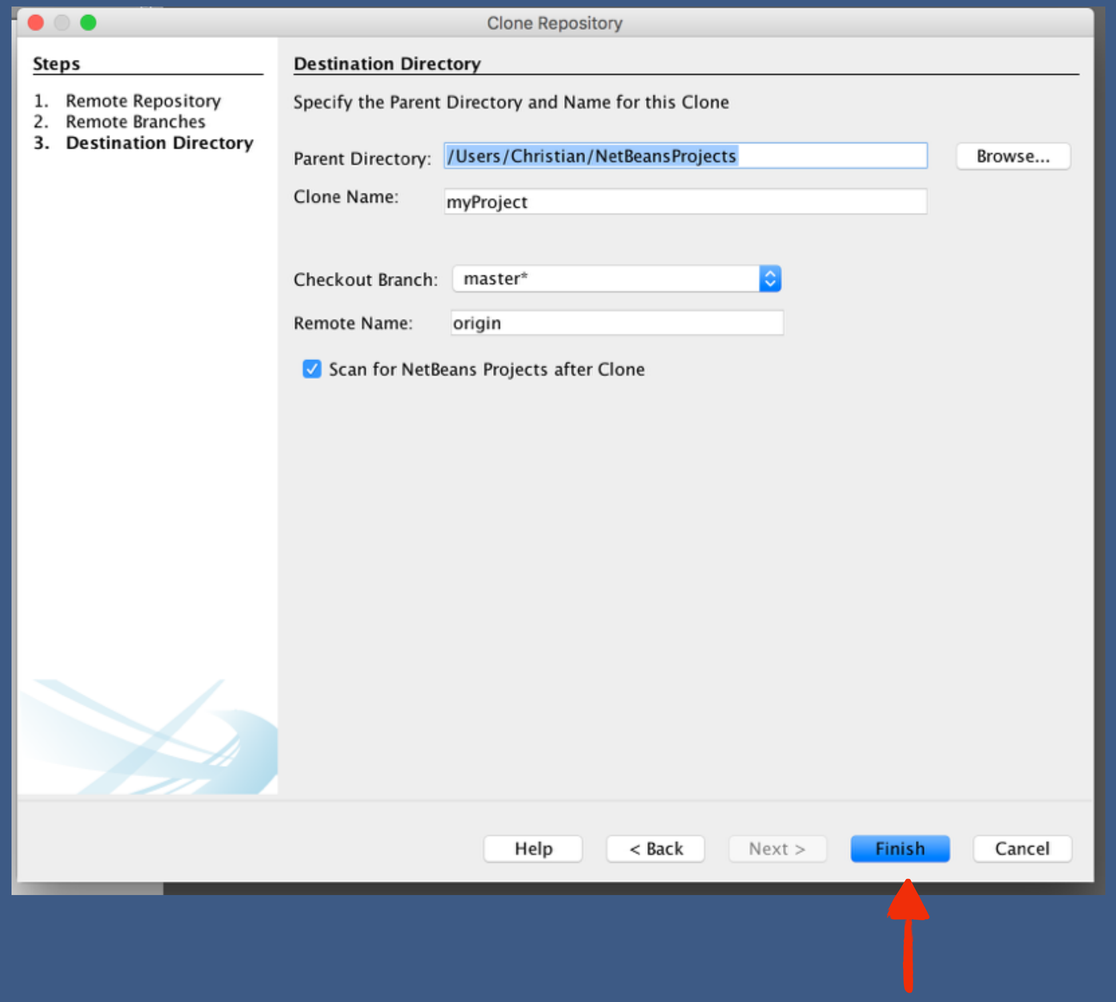




Your Uni.lu credentials

Paste remote repository URL


Clone Repository
Steps
1. Remote Repository
2. Remote Branches
3. Destination Directory
Remote Branches
Select Remote Branches
master*
Select All
Select None
Clone name can't be empty
Help
< Back
Next >
Finish
Cancel


Clone Repository
Steps
1. Remote Repository
2. Remote Branches
3. Destination Directory
Destination Directory
Specify the Parent Directory and Name for this Clone
Parent Directory:
/Users/Christian/NetBeansProjects
Browse...
Clone Name:
myProject
Checkout Branch:
master*
Remote Name:
origin
Scan for NetBeans Projects after Clone
Help
< Back
Next >
Finish
Cancel

Terminal

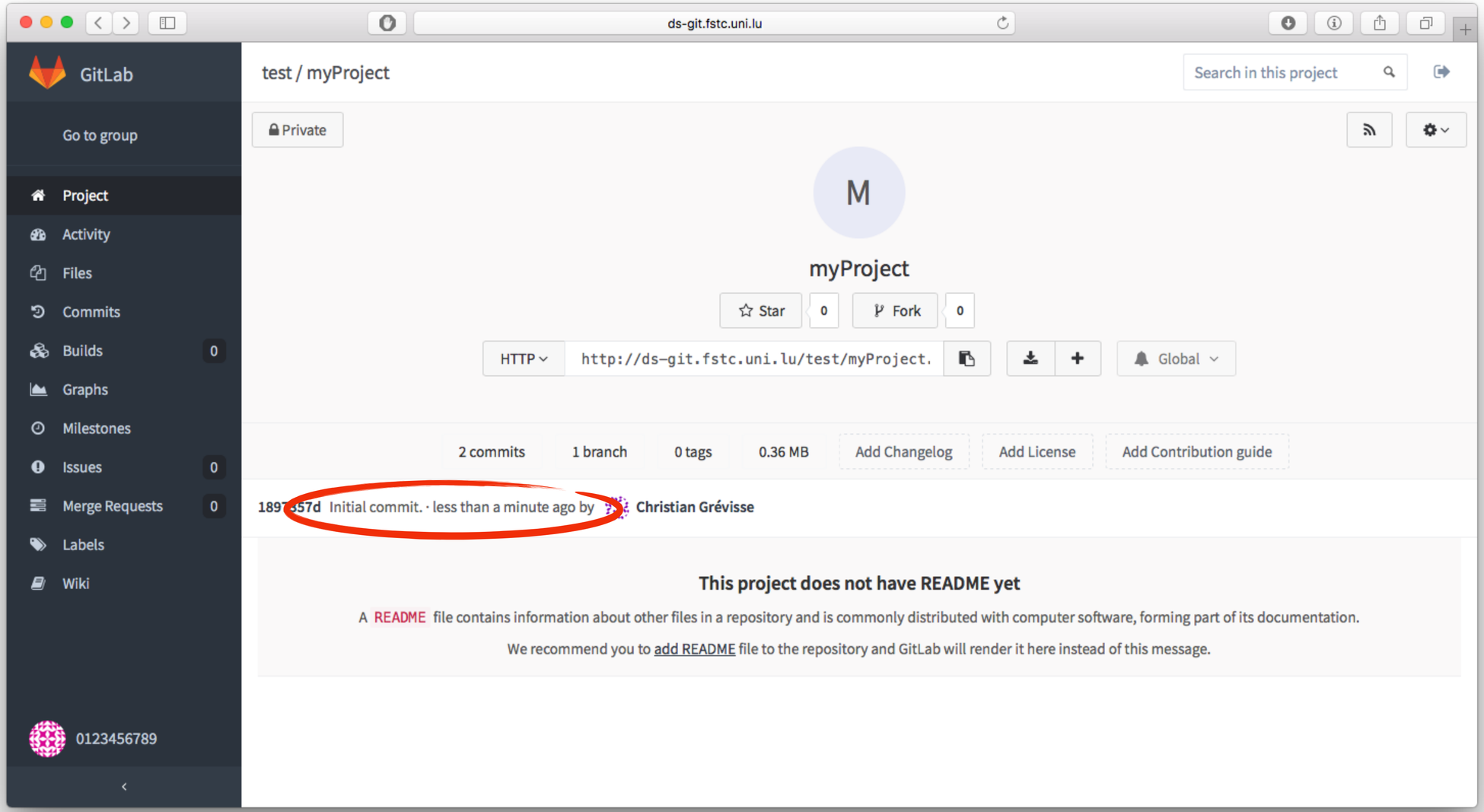
Initial repository setup & pull current version

```
$ echo 'Hello, World!' > Test.txt
$ git init
Initialized empty Git repository in ~/GitTest/.git/
$ git add Test.txt
$ git commit -m "Initial commit"
[master (root-commit) bfcf3c0] Initial commit
 1 file changed, 1 insertion(+)
 create mode 100644 Test.txt
$ git remote add origin https://ds-git.fstc.uni.lu/test/myProject.git
$ git push -u origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 241 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://ds-git.fstc.uni.lu/test/myProject.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working directory clean
$ git pull
Already up-to-date.
```

Clone existing project

```
$ git clone https://ds-git.fstc.uni.lu/test/myProject.git
Cloning into 'myProject'...
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), done.
Checking connectivity... done.
```

GitLab Dashboard



Further reading

Git Documentation

GitLab CE Documentation

EGit Documentation

Xcode Source Control Management Help

How To Use Git Source Control with Xcode in iOS 7

Git in Visual Studio

Git in NetBeans

Further reading

Scott Chacon and Ben Straub

Pro Git

2nd edition, 2014.

Apress, Berkely, CA, USA.

Available for free: https://git-scm.com/book/en/v2