

**Total-QA**

Future of Software Testing

DevOps Tutorial-2 Resolve merge conflicts in Git and GitHub

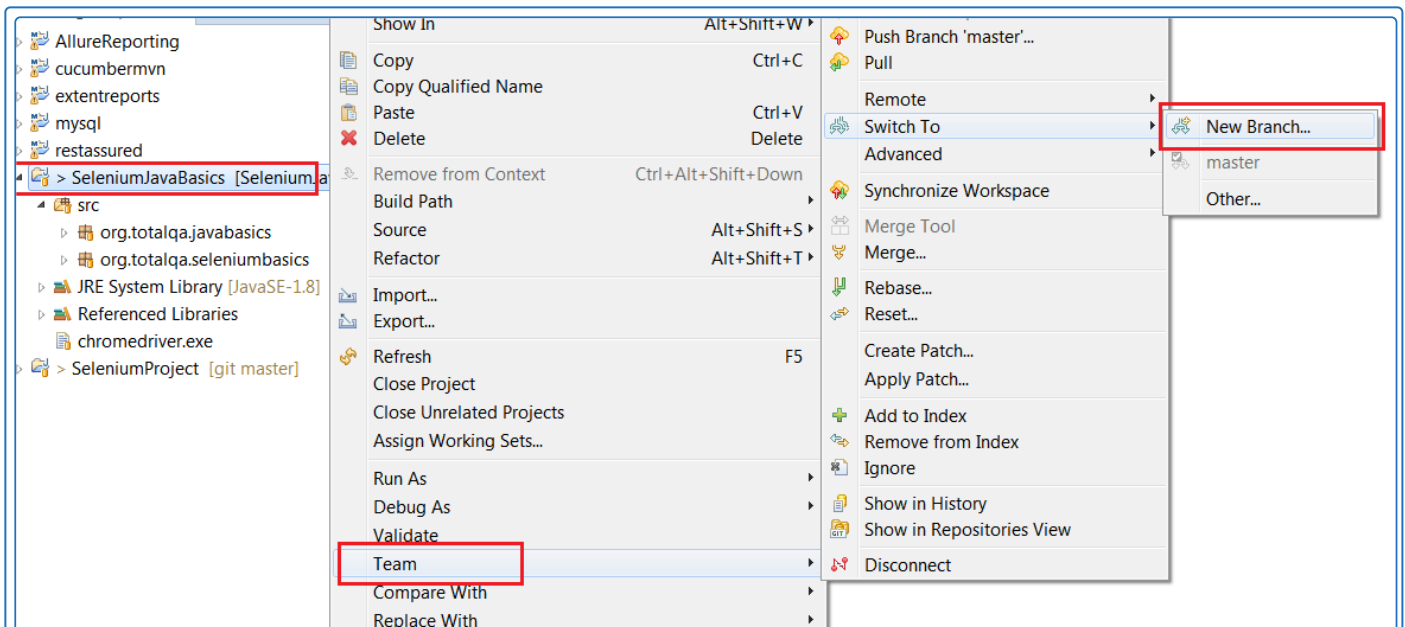
total-qa August 30, 2018 No Comments

In this tutorial we will learn about resolving the merge conflicts in Git and `Github`. Git can often resolve differences between merged branches. Usually, the changes are on different lines, or even in different files, which makes the merge simple for computers to understand.

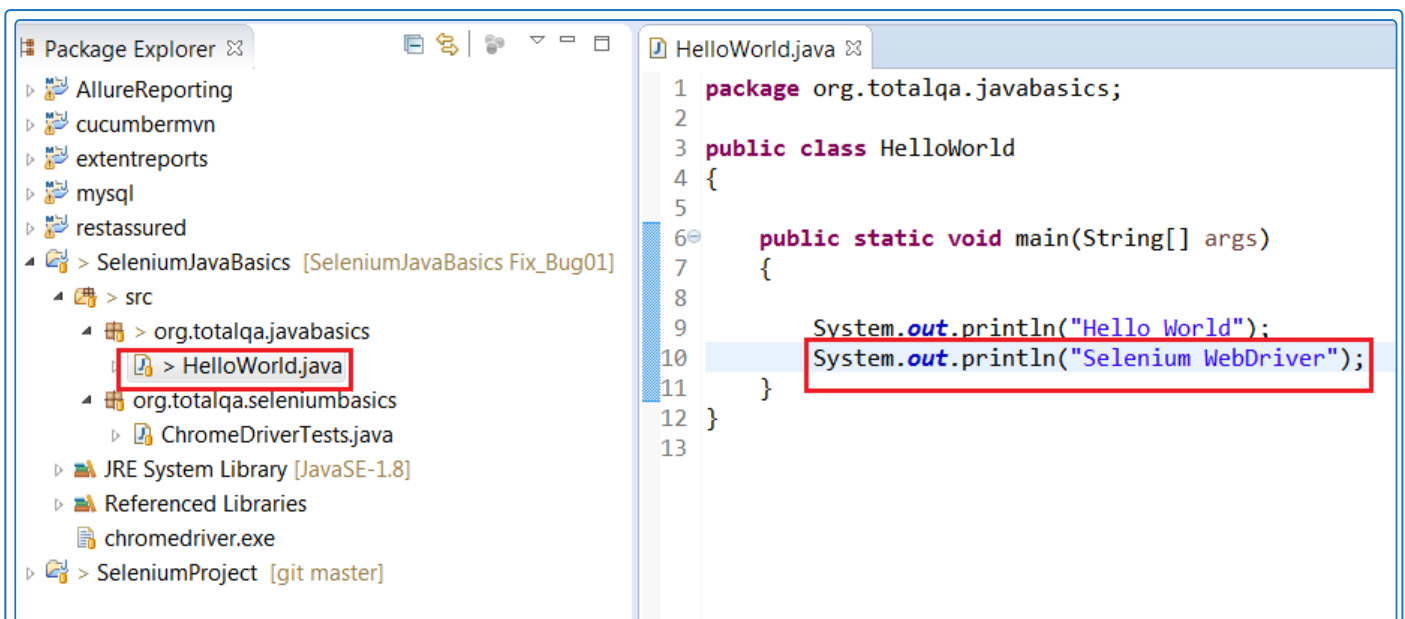
However, sometimes there are competing changes that Git needs your help with to decide which changes to incorporate in the final `merge`. Often, merge `conflicts` happen when people make different changes to the same line of the same file, or when one person edits a file and another person deletes the same file. You must `resolve` the conflict before you can merge the branches.

Lets discuss this scenario in detail:

1. Generally developers or testers create a new branch and modify the `HelloWorld.java` in there own branch `Fix_Bug01`.

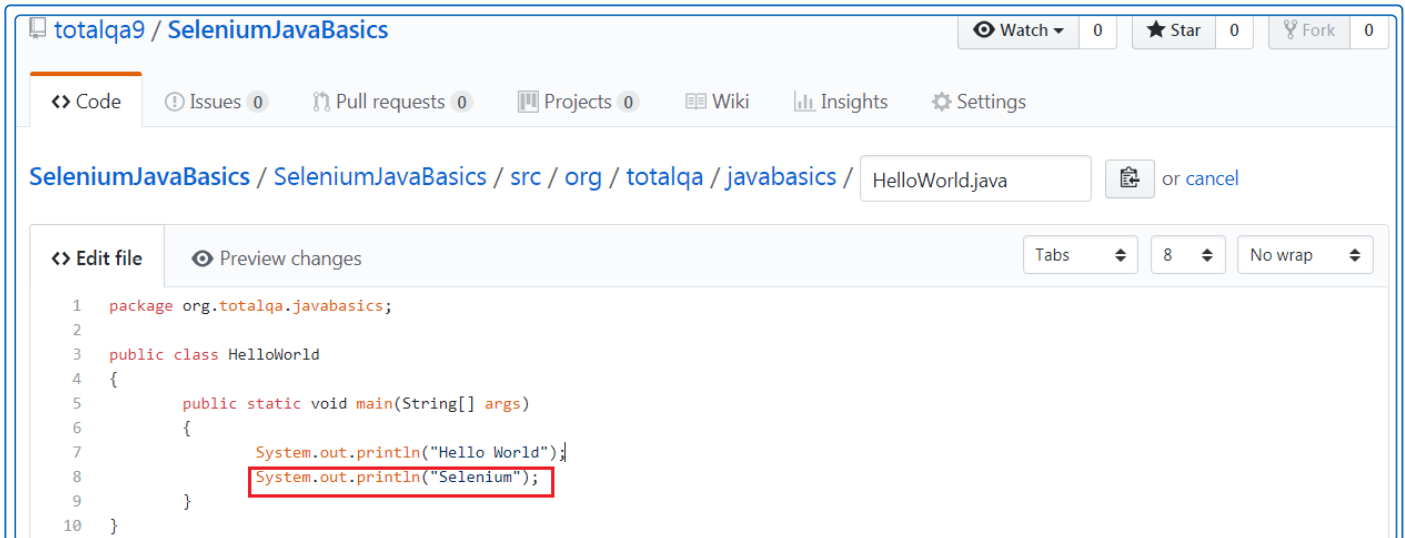


Creating a new branch for safely experimenting the changes



Adding a new line

2. During the same time if another team member modify the same file and same lines in the `master` branch in the Github Repository.



totalqa9 / SeleniumJavaBasics

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

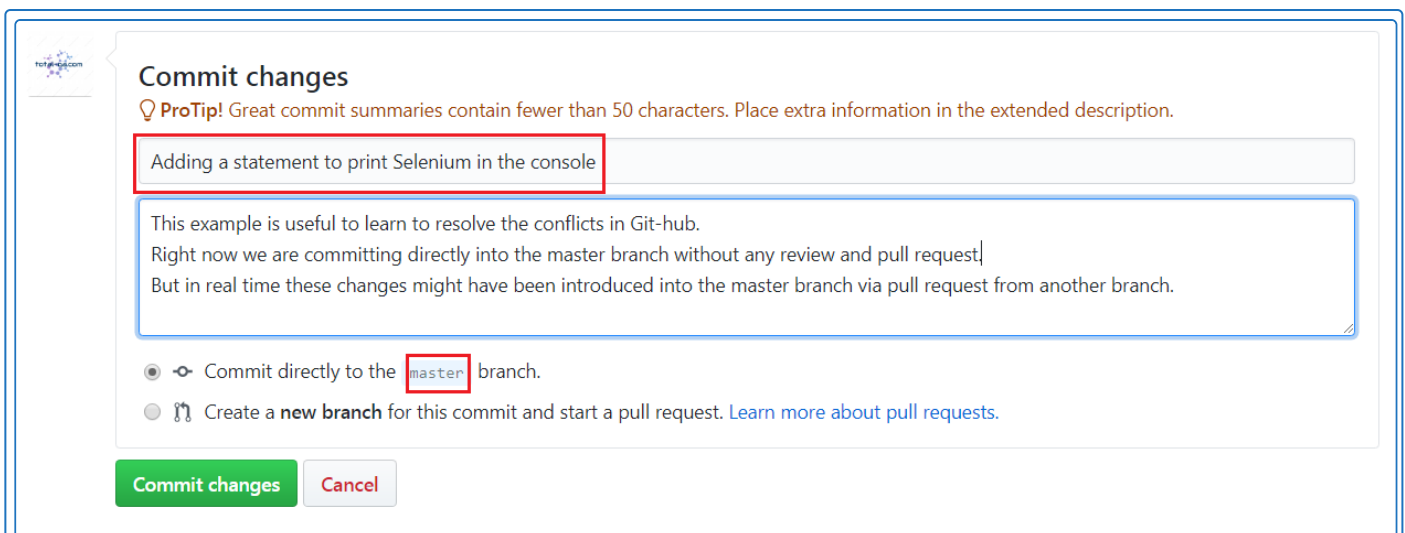
SeleniumJavaBasics / SeleniumJavaBasics / src / org / totalqa / javabasics / HelloWorld.java or cancel

Edit file Preview changes Tabs 8 No wrap

```

1 package org.totalqa.javabasics;
2
3 public class HelloWorld
4 {
5     public static void main(String[] args)
6     {
7         System.out.println("Hello World");
8         System.out.println("Selenium");
9     }
10 }
```

Updating the File in Github master branch



Commit changes

ProTip! Great commit summaries contain fewer than 50 characters. Place extra information in the extended description.

Adding a statement to print Selenium in the console

This example is useful to learn to resolve the conflicts in Git-hub.
Right now we are committing directly into the master branch without any review and pull request
But in real time these changes might have been introduced into the master branch via pull request from another branch.

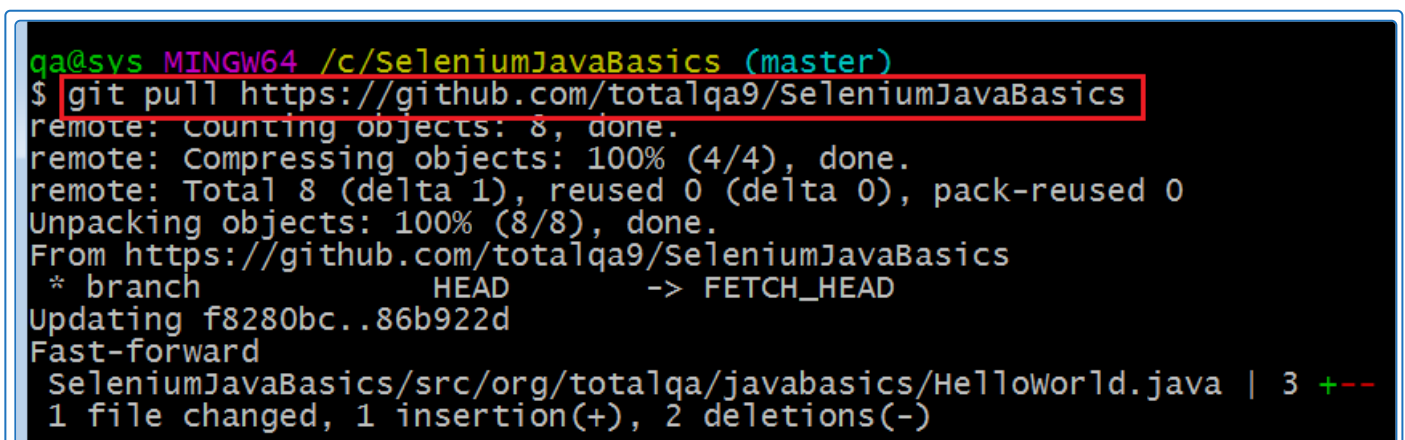
☒ Commit directly to the master branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel

Committing the changes

3. In Local Repository if the master branch is refreshed from Global Repository and try to merge these changes in the branch `Fix_Bug01` which introduces the merge conflicts.



```

qa@sys MINGW64 /c/SeleniumJavaBasics (master)
$ git pull https://github.com/totalqa9/SeleniumJavaBasics
remote: Counting objects: 8, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (8/8), done.
From https://github.com/totalqa9/SeleniumJavaBasics
 * branch                HEAD      -> FETCH_HEAD
Updating f8280bc..86b922d
Fast-forward
 SeleniumJavaBasics/src/org/totalqa/javabasics/HelloWorld.java | 3 +--
 1 file changed, 1 insertion(+), 2 deletions(-)
```

Refresh the Master

```
qa@sys MINGW64 /c/SeleniumJavaBasics (master)
$ git checkout Fix_Bug01
Switched to branch 'Fix_Bug01'
Your branch and 'master' have diverged,
and have 2 and 1 different commits each, respectively.
(use "git pull" to merge the remote branch into yours)

qa@sys MINGW64 /c/SeleniumJavaBasics (Fix_Bug01)
$ git merge master
Auto-merging SeleniumJavaBasics/src/org/totalqa/javabasics/HelloWorld.java
CONFLICT (content): Merge conflict in SeleniumJavaBasics/src/org/totalqa/javabasics/HelloWorld.java
Automatic merge failed; fix conflicts and then commit the result.
```

Conflicts due to changes in the same file and same line

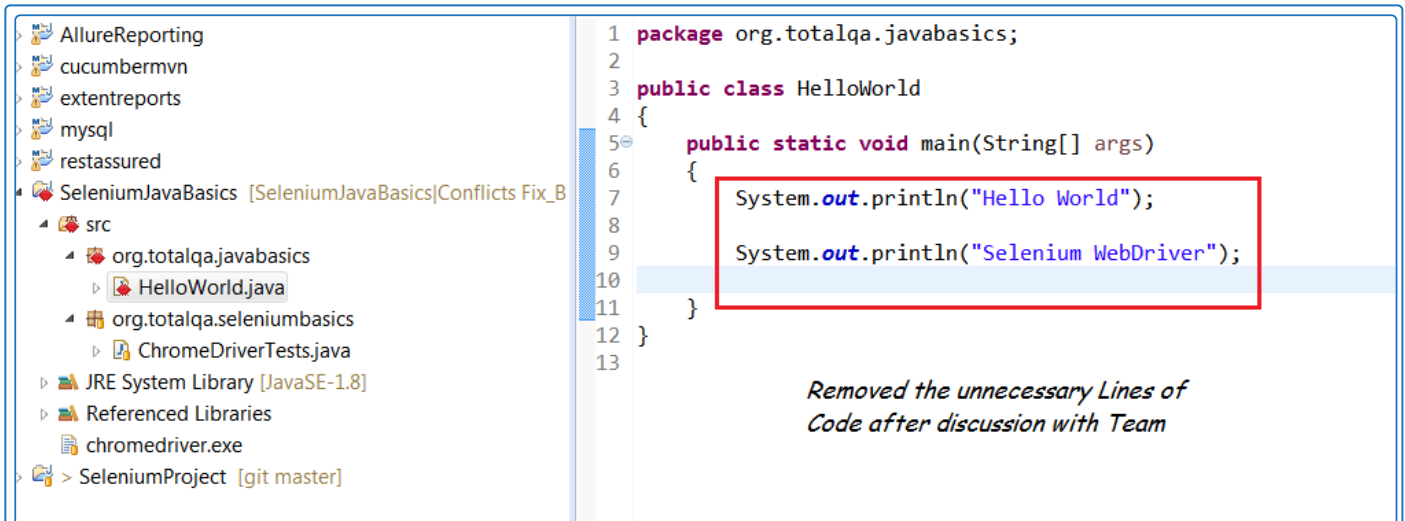
4. Fixing the merge Conflicts by ignoring the lines which are not required.

```
Package Explorer
├── AllureReporting
├── cucumbermvn
├── extentreports
├── mysql
├── restassured
├── SeleniumJavaBasics [SeleniumJavaBasics]Conflicts Fix_B
    └── src
        ├── org.totalqa.javabasics
        │   └── HelloWorld.java
        ├── org.totalqa.seleniumbasics
        │   └── ChromeDriverTests.java
        ├── JRE System Library [JavaSE-1.8]
        ├── Referenced Libraries
        ├── chromedriver.exe
        └── SeleniumProject [git master]

HelloWorld.java
1 package org.totalqa.javabasics;
2
3 public class HelloWorld
4 {
5     public static void main(String[] args)
6     {
7         System.out.println("Hello World");
8 <<<<<<< HEAD
9         System.out.println("Selenium WebDriver");
10 =====
11 >>>>>> master
12         System.out.println("Selenium");
13     }
14 }
15

master :: Global Repository Changes
HEAD :: Local Repository Changes
```

Merge Conflicts



```

1 package org.totalqa.javabasics;
2
3 public class HelloWorld
4 {
5     public static void main(String[] args)
6     {
7         System.out.println("Hello World");
8         System.out.println("Selenium WebDriver");
9     }
10 }
11
12 }
13

```

Removed the unnecessary Lines of Code after discussion with Team

Deleted the unnecessary lines of code

5. Execute the below commands in the Git bash to resolve the merge conflict.

```
git status
```

```
git add
```

```
git commit -m "Commit Message"
```

```

qa@sys MINGW64 /c/SeleniumJavaBasics (Fix_Bug01|MERGING)
$ git status
On branch Fix_Bug01
Your branch and 'master' have diverged,
and have 2 and 1 different commits each, respectively.
(use "git pull" to merge the remote branch into yours)
You have unmerged paths.
(fix conflicts and run "git commit")
(use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)

        both modified:   SeleniumJavaBasics/src/org/totalqa/javabasics/HelloWorld.java

no changes added to commit (use "git add" and/or "git commit -a")

qa@sys MINGW64 /c/SeleniumJavaBasics (Fix_Bug01|MERGING)
$ git add SeleniumJavaBasics/src/org/totalqa/javabasics/HelloWorld.java

qa@sys MINGW64 /c/SeleniumJavaBasics (Fix_Bug01|MERGING)
$ git commit -m "Fix merge Conflicts"
[Fix_Bug01 91a8ad9] Fix merge Conflicts

qa@sys MINGW64 /c/SeleniumJavaBasics (Fix_Bug01)
$

```

Fixing the merge conflicts

Conclusion:

In this example we have learnt how to resolve the merge conflicts in Github and Git.

📁 DevOps, JAVA, REST API, Selenium, TESTNG 🔗 bash, DevOPS, Git, git branch, git commit, git pull, git status, github, merge, Pull, Push, resolve conflicts

Related posts...

- » [Execute Selenium WebDriver Tests from Jenkins 2.0 Pipeline](#)
 - » [DevOps Tutorial-3 Triggering CI Build with Github Push Notifications](#)
 - » [DevOps Tutorial-1 Pushing Code to Github](#)
 - » [Integrating GitHub with Jenkins for Continuous Integration and Continuous Delivery](#)
-