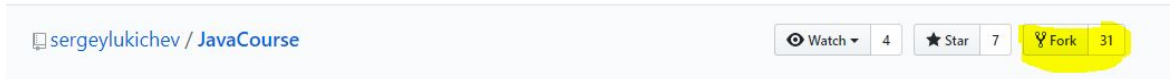


Создание веток (бранчей) и пулл риквестов (pull requests)

Что необходимо для начала:

- Аккаунт на github.com
- git клиент (windows - <https://gitforwindows.org/> mac - часть x-code, linux - встроено)
- Репозиторий на github.com, с веткой мастер, например <https://github.com/GITUSERNAME/JavaCourse>, который может быть создан с нуля, так и форкнут у преподавателя



- Если мы форкнули репозиторий, то его нужно клонировать себе на компьютер при помощи git clone. Не забудьте перейти в папку этого репозитория у себя на компьютере

```
$ git clone https://github.com/altaskov/JavaCourse.git
Cloning into 'JavaCourse'...
remote: Enumerating objects: 749, done.
remote: Total 749 (delta 0), reused 0 (delta 0), pack-reused 749
Receiving objects: 100% (749/749), 156.95 KiB | 558.00 KiB/s, done.
Resolving deltas: 100% (151/151), done.

Ndrew@Mashinko MINGW64 ~/Projects
$ cd JavaCourse/
```

- ВАЖНО: Если репозиторий был форкнут - вкладки Issues не будет! Оставить комментарий мы сможем при открытии pull request

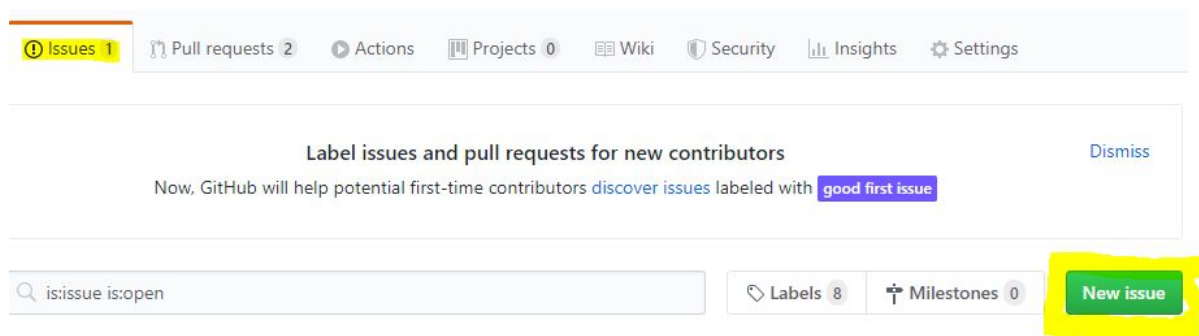
0. Для начала убедимся в GitBash, что мы находимся в рабочей папке, где находится наш проект и что мы находимся на ветке master командой: git status.

```
Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (master)
$ git status
On branch master
```

1. Создаем новый issue на github.com:

ВАЖНО: Если репозиторий был форкнут - вкладки Issues не будет! Оставить комментарий мы сможем при открытии pull request

<https://github.com/GITUSERNAME/JavaCourse/issues> -> зеленая кнопка New Issue справа.



Вводим заголовок и условие задачи.

GH-9-temperature-amplitude

Related Issues Beta, Try it.

Write Preview

AA B i “ <> 🔗 ☰ ☷ ☹ @ 📌 ↶

Task:
GH-9-temperature-amplitude text
Сделать:
1) To
2) Сë
5)
10)

Attach files by dragging & dropping, selecting or pasting them. 📎

Styling with Markdown is supported

Submit new issue

2. Создаем новую ветку командой в gitbash
`git checkout -b GH-9-temperature-amplitude` (Без пробелов)
где GH-9-temperature-amplitude - название ветки, а девятка в GH-9 - номер созданного в предыдущем пункте issue.

```
Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (master)
$ git checkout -b GH-9-temperature-amplitude
Switched to a new branch 'GH-9-temperature-amplitude'

Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (GH-9-temperature-amplitude)
```

3. Создаем проект в IDE (Eclipse или IntelliJ) и пишем код, как обычно.
4. Когда задача решена, переходим в окно gitbash и смотрим список файлов на добавление командой `git status`,

```
Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (GH-9-temperature-amplitude)
$ git status
On branch GH-9-temperature-amplitude
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   WeatherForecast/src/main/java/telran/weather/ForecastResult.java
        modified:   WeatherForecast/src/main/java/telran/weather/WeatherForecaster.java

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        WeatherForecast/src/main/java/telran/weather/README.md
```

после чего добавляем файлы командой `git add .`

И видим, что `git` зафиксировал изменения `git status`:

```
Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (GH-9-temperature-amplitude)
$ git add .

Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (GH-9-temperature-amplitude)
$ git status
On branch GH-9-temperature-amplitude
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   WeatherForecast/src/main/java/telran/weather/ForecastResult.java
        new file:   WeatherForecast/src/main/java/telran/weather/README.md
        modified:   WeatherForecast/src/main/java/telran/weather/WeatherForecaster.java
```

5. После добавления создаем commit, после чего файлы добавлены в локальную ветку (бранч):

`git commit -m "first solution"`, где `first solution` это комментарий.

На этом этапе `git` должен знать того, кто с ним работает, чтобы "подписать файлы" и подсказывает, что нужно сделать:

```
$ git commit -m "first solution"

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"
```

Когда коммит прошел нормально, мы видим следующее (в вашем конкретном случае может отличаться):

```
$ git commit -m "first solution"
[GH-9-temperature-amplitude 5e7a07e] first solution
3 files changed, 2 insertions(+)
create mode 100644 WeatherForecast/src/main/java/telran/weather/README.md
```

6. Теперь нужно запустить файлы в удаленной репозиторий на `github.com`:

`git push`

так как удаленной ветки не существует, `git` подсказывает, что команда, скорее всего, выглядит так:

`git push --set-upstream origin GH-9-temperature-amplitude`

(кстати, быстрое копирование в `gitbash` - выделяем строку и нажимаем на среднюю кнопку мыши, что сразу же вставит эту выделенную строку)

```
$ git push
fatal: The current branch GH-9-temperature-amplitude has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin GH-9-temperature-amplitude

Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (GH-9-temperature-amplitude)
$ git push --set-upstream origin GH-9-temperature-amplitude

Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 6 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (10/10), 713 bytes | 713.00 KiB/s, done.
Total 10 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), completed with 5 local objects.
remote:
remote: Create a pull request for 'GH-9-temperature-amplitude' on GitHub by visiting:
remote:   https://github.com/aliaskov/JavaCourse/pull/new/GH-9-temperature-amplitude
remote:
To https://github.com/aliaskov/JavaCourse.git
 * [new branch]      GH-9-temperature-amplitude -> GH-9-temperature-amplitude
Branch 'GH-9-temperature-amplitude' set up to track remote branch 'GH-9-temperature-amplitude' from 'origin'.
```

7. Теперь можно открывать pull request: возвращаемся в окно браузера на `github.com` в свой репозиторий. Можем сразу сделать пул, так как `github` сразу подсказывает, что совсем недавно появилась новая ветка:

Your recently pushed branches:

🔗 GH-9-temperature-amplitude (6 minutes ago)

🔗 Compare & pull request

Или же сделать все вручную: нажимаем кнопку `New Pull Request`, выбираем ветки, ИЗ которой и В которую предлагаем изменения и нажимаем `Create pull request`:

base: master
compare: GH-9-temperature-amplitude
✓ Able to merge. These branches can be automatically merged.

Create pull request
Discuss and review the changes in this comparison with others.

в открывшемся окне указываем ссылку на изначальный issue.

Если написать #, то выпадет список issues, из которого можно выбрать нужный.

Если вы делали форк репозитория, то issues не будет, но можно оставить комментарий.

Gh 9 temperature amplitude

Write
Preview
AA B i “ <> ↺ ⋮ ≡ ✓ @ 📎 ↶

Вот моё невероятное решение

Attach files by dragging & dropping, selecting or pasting them.

☒ Allow edits from maintainers. [Learn more](#)
Create pull request

Смело жмем Create pull request

Список файлов и переписка будет видна во вкладке открытого PR:

2 commits
4 files changed
0 commit comments

Commits on Mar 04, 2020
andrewliaskovski... Added hello world

Commits on Mar 27, 2020
aliaskov first solution

Showing 4 changed files with 16 additions and 0 deletions.

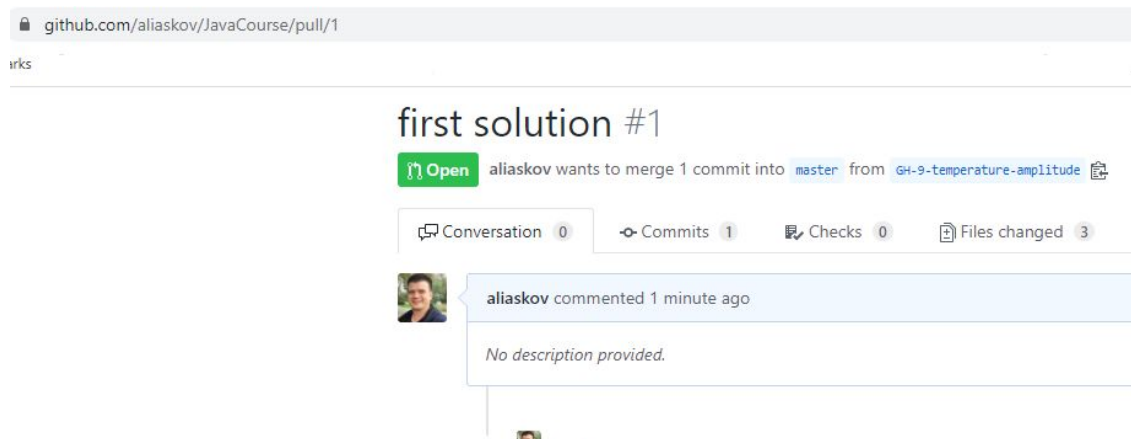
14 Helloworld/src/Main.java

```

... @@ -0,0 +1,14 @@
1 + package HelloWorld;
2 +
3 + public class Main {
4 +     public static void main(String[] args) throws InterruptedException {
5 +         boolean flag = true;
6 +         System.out.println("Hello world");
7 +         System.out.println("Hello from Andrew");

```


8. Ссылку на созданный pull request можно (URL из браузера, например https://github.com/GITHUBUSER/Java_course/pull/2) отправить преподавателю, если проверка не требуется, то pull request можно слить (смерджить), в нашем случае из ветки GH-9-temperature-amplitude в ветку master.



9. Для того, чтобы приступить к работе над следующей задачей, в gitbash выполняем `git checkout master`
`git pull`
это заберет обновленный удаленный мастер в локальную папку, после чего можно вернуться к шагу 1.

Как обновить репозиторий, который был форкнут:

- 1) Проверяем связи с начальным репозиторием
`$ git remote -v`
origin <https://github.com/aliaskov/JavaCourse.git> (fetch)
origin <https://github.com/aliaskov/JavaCourse.git> (push)
- 2) Добавляем новый источник
`git remote add upstream https://github.com/sergeylukichev/JavaCourse`
- 3) Обновляем и получаем новые ветки
`git fetch upstream`

```
Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (GH-9-temperature-amplitude)
$ git remote -v
origin https://github.com/aliaskov/JavaCourse.git (fetch)
origin https://github.com/aliaskov/JavaCourse.git (push)

Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (GH-9-temperature-amplitude)
$ git remote add upstream https://github.com/sergeylukichev/JavaCourse

Ndrew@Mashinko MINGW64 ~/Projects/JavaCourse (GH-9-temperature-amplitude)
$ git fetch upstream
remote: Enumerating objects: 176, done.
remote: Counting objects: 100% (176/176), done.
remote: Compressing objects: 100% (53/53), done.
remote: Total 656 (delta 122), reused 152 (delta 111), pack-reused 480
Receiving objects: 100% (656/656), 105.39 KiB | 547.00 KiB/s, done.
Resolving deltas: 100% (217/217), completed with 33 local objects.
From https://github.com/sergeylukichev/JavaCourse
 * [new branch]      GH-4-diff21      -> upstream/GH-4-diff21
 * [new branch]      GH-6-four-seasons -> upstream/GH-6-four-seasons
 * [new branch]      GH-8-longest-substring -> upstream/GH-8-longest-substring
 * [new branch]      db-tests         -> upstream/db-tests
 * [new branch]      dependabot/maven/Drafts/JsonDemo/com.fasterxml.jackson.core-jackson-databind-2.9.10.3 -> upstream/dependabot/maven/Drafts/JsonDemo/com.fasterxml.jackson.core-jackson-databind-2.9.10.3
 * [new branch]      master           -> upstream/master
 * [new branch]      school-controller -> upstream/school-controller
 * [new branch]      test-branch       -> upstream/test-branch
```

4) Переходим на мастер ветку

```
git checkout master
```

5) Сливаем ветки, возможны конфликты, но если повезло и конфликтов нет, то просто выходим из vi при помощи :x

```
git merge upstream/master
```

```
$ git merge upstream/master
Removing Multithreading/AssemblyLine/test/de/telran/prodline/AssemblyLineTest.java
Removing Multithreading/AssemblyLine/src/de/telran/prodline/AssemblyLine.java
Removing Multithreading/AssemblyLine/src/de/telran/model/parts/Frame.java
Removing Multithreading/AssemblyLine/src/de/telran/model/parts/Engine.java
Removing Multithreading/AssemblyLine/src/de/telran/model/parts/Component.java
Removing Multithreading/AssemblyLine/src/de/telran/model/parts/Car.java
Merge made by the 'recursive' strategy.
 Drafts/CollectorDemo/.gitignore | 26 ++
 Drafts/CollectorDemo/src/Box.java | 24 ++
 Drafts/CollectorDemo/src/Main.java | 48 +++++
 Drafts/CollectorDemo/src/Person.java | 19 ++
 Drafts/CounterDemo/.gitignore | 26 ++
 Drafts/CounterDemo/src/Counter.java | 28 ++
 Drafts/CounterDemo/src/CounterThread.java | 21 ++
 Drafts/CounterDemo/src/Main.java | 19 ++
 Drafts/DayOfWeek/.gitignore | 8 +
 Drafts/DayOfWeek/src/DayOfWeek.java | 9 +
 Drafts/DayOfWeek/src/DayOfWeekResolver.java | 30 +++++
 Drafts/DayOfWeek/src/Main.java | 11 +
 Drafts/GitDemo/.classpath | 6 +
 Drafts/GitDemo/.gitignore | 1 +
 Drafts/GitDemo/.project | 17 ++
 .../GitDemo/.settings/org.eclipse.jdt.core.prefs | 11 +
 Drafts/GitDemo/src/Main.java | 9 +
 Drafts/GuessNumberGame/.gitignore | 26 ++
 .../GuessNumberGame/src/de/telran/game/Game.java | 68 +++++
 .../src/de/telran/game/GameTest.java | 25 ++
 .../src/de/telran/game/NumberSource.java | 5 +
 .../src/de/telran/game/ResultOutput.java | 5 +
 .../src/de/telran/game/TestOutput.java | 10 +
 Drafts/JsonDemo/.gitignore | 26 ++
 Drafts/JsonDemo/pom.xml | 48 +++++
 .../src/de/telran/game/ResultOutput.java | 5 +
 .../src/de/telran/game/TestOutput.java | 10 +
 Drafts/JsonDemo/.gitignore | 26 ++
 Drafts/JsonDemo/pom.xml | 48 +++++
```

6) Заливаем это все на github, в свой репозиторий

```
git push origin master
```

```
$ git push origin master
Enumerating objects: 634, done.
Counting objects: 100% (606/606), done.
Delta compression using up to 6 threads
Compressing objects: 100% (328/328), done.
Writing objects: 100% (576/576), 130.29 KiB | 21.71 MiB/s, done.
Total 576 (delta 178), reused 537 (delta 160), pack-reused 0
remote: Resolving deltas: 100% (178/178), completed with 10 local objects.
To https://github.com/aliaskov/JavaCourse.git
 336410c..34aa2e1 master -> master
```