

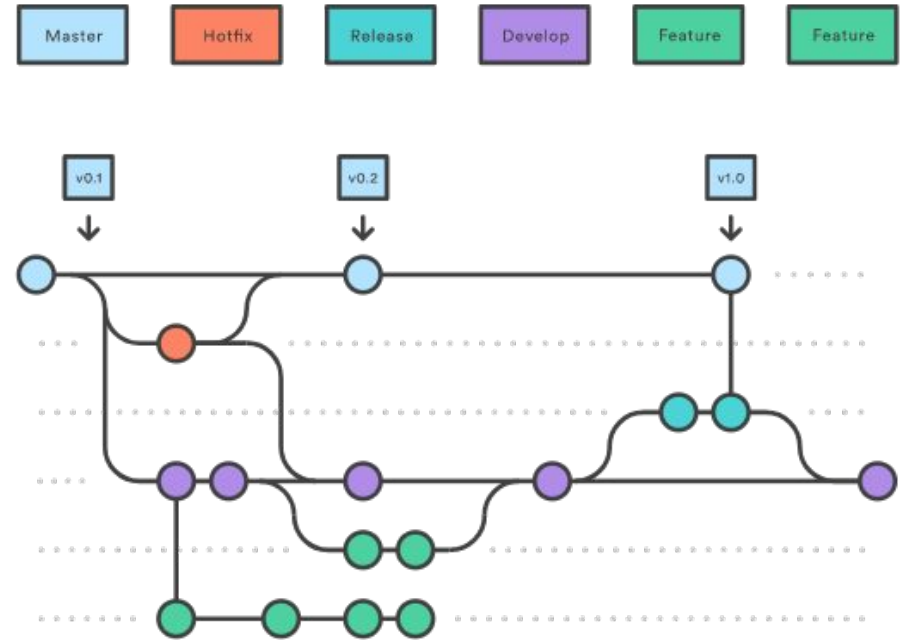
Основы Git ч.2

Ветвление

Основы ветвления

Используя ветвление, вы отклоняетесь от основной линии разработки и продолжаете работу независимо от неё, не вмешиваясь в основную линию. Все изменения, которые будут сделаны в новой ветки никак не затронут файлы из других веток.

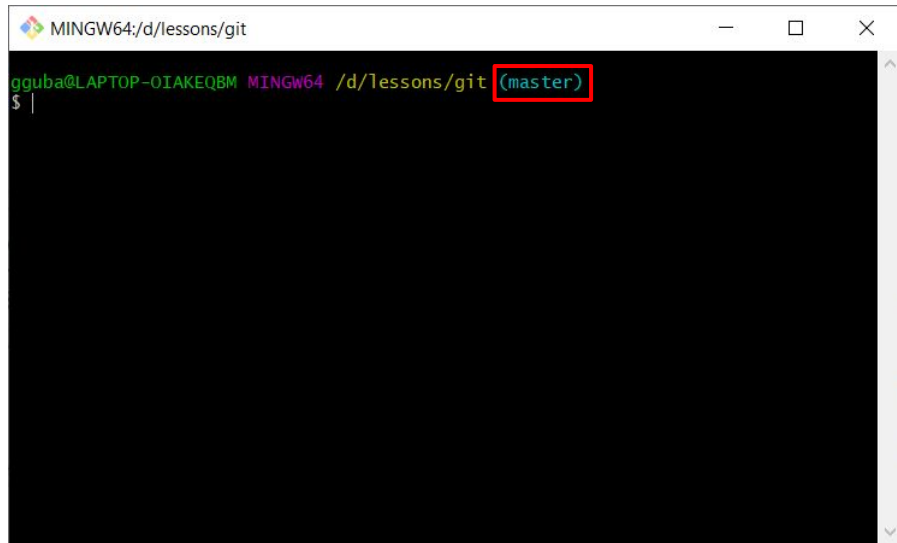
Очень грубо говоря, вы создаёте копию вашего репозитория со всеми изменениями которые были в коммите от которого вы отклонились.



Ох мой master

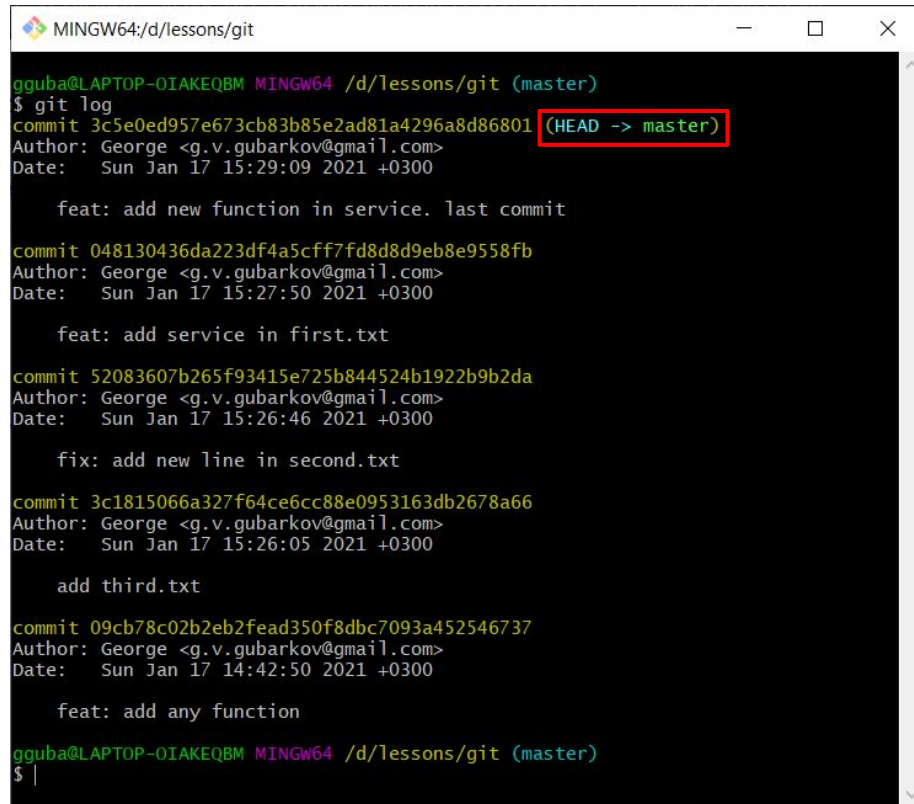
По умолчанию при работе с git существует одна ветка - master. Обычно она является главной.

Никаких разработок и коммитов в данной ветке обычно делать нельзя.

A screenshot of a terminal window titled 'MINGW64:/d/lessons/git'. The prompt shows the user 'gguba@LAPTOP-OIAKEQBM' in a 'MINGW64' environment at the directory '/d/lessons/git'. The text '(master)' is highlighted with a red rectangle, indicating the current branch. The prompt ends with a dollar sign and a vertical bar '\$ |'.

А голову свою ты не забыл(HEAD)?

HEAD — это указатель на текущую ветку, которая, в свою очередь, является указателем на последний коммит, сделанный в этой ветке. Это значит, что HEAD будет родителем следующего созданного коммита. Как правило, самое простое считать HEAD снимком **вашего последнего коммита**.



```
MINGW64:/d/lessons/git
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git log
commit 3c5e0ed957e673cb83b85e2ad81a4296a8d86801 (HEAD -> master)
Author: George <g.v.gubarkov@gmail.com>
Date: Sun Jan 17 15:29:09 2021 +0300

    feat: add new function in service. last commit

commit 048130436da223df4a5cff7fd8d8d9eb8e9558fb
Author: George <g.v.gubarkov@gmail.com>
Date: Sun Jan 17 15:27:50 2021 +0300

    feat: add service in first.txt

commit 52083607b265f93415e725b844524b1922b9b2da
Author: George <g.v.gubarkov@gmail.com>
Date: Sun Jan 17 15:26:46 2021 +0300

    fix: add new line in second.txt

commit 3c1815066a327f64ce6cc88e0953163db2678a66
Author: George <g.v.gubarkov@gmail.com>
Date: Sun Jan 17 15:26:05 2021 +0300

    add third.txt

commit 09cb78c02b2eb2fead350f8dbc7093a452546737
Author: George <g.v.gubarkov@gmail.com>
Date: Sun Jan 17 14:42:50 2021 +0300

    feat: add any function

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ |
```

А что с ветками?

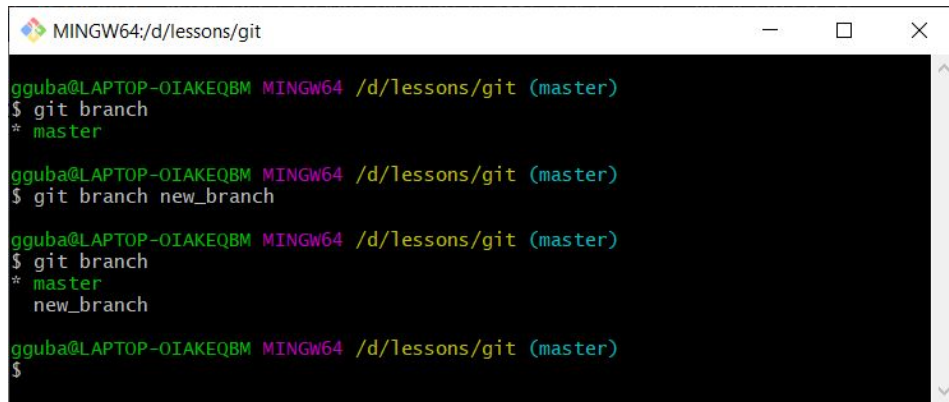
Для просмотра созданных веток используется команда *git branch*. Для создания новых веток *git branch name_branch*

```
$ git branch
```

```
$ git branch name_branch
```

Звездочка указывает на ветку в которой вы сейчас находитесь.

Когда вы создаёте новую ветку, без указания дополнительных флагов - ветка создается от HEAD.



```
MINGW64:/d/lessons/git

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git branch
* master

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git branch new_branch

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git branch
* master
  new_branch

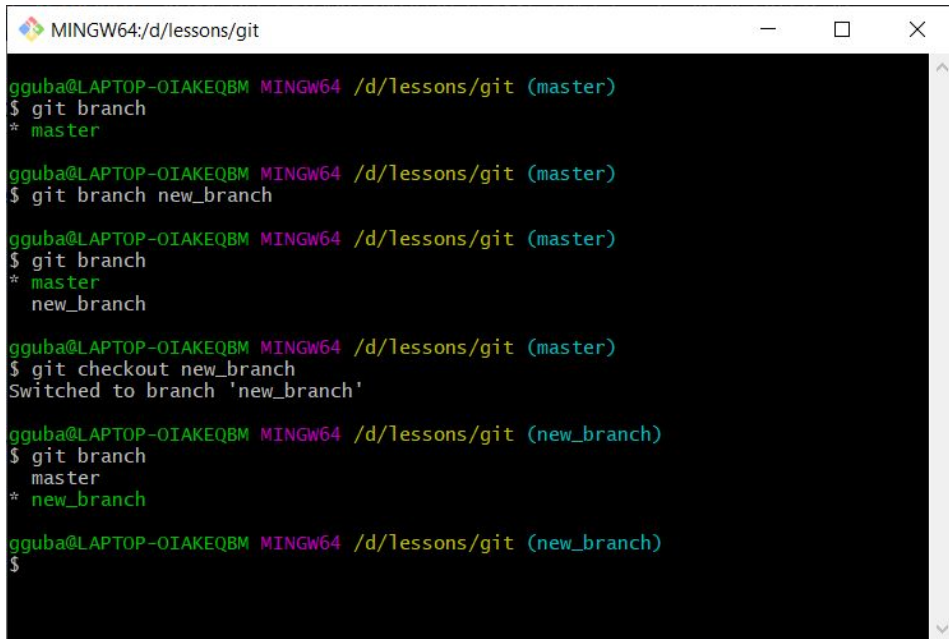
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$
```

Смена ветки

Для того чтобы сменить `git checkout name_branch`.

```
$ git checkout name_branch
```

Теперь всё что вы сделаете в репозитории не затронет ветки `master`.

A screenshot of a terminal window titled 'MINGW64:/d/lessons/git'. The window shows a series of Git commands and their outputs. The user is currently on the 'master' branch. They run 'git branch' and see 'master'. Then they run 'git branch new_branch' to create a new branch. They run 'git branch' again and see both 'master' and 'new_branch'. Finally, they run 'git checkout new_branch' and receive the message 'Switched to branch 'new_branch''. They run 'git branch' again and see 'master' and 'new_branch', with 'new_branch' marked with an asterisk to indicate it is the current branch. The prompt changes from '\$' to '\$ 'new_branch\$'.

```
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git branch
* master

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git branch new_branch

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git branch
* master
  new_branch

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git checkout new_branch
Switched to branch 'new_branch'

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git branch
  master
* new_branch

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$
```

Разница веток

Работа в new_branch

```
MINGW64:/d/lessons/git
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ touch new_branch_file

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git status
On branch new_branch
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    new_branch_file

nothing added to commit but untracked files present (use "git add" to track)

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git add .

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git commit -m "add file new_branch_file in new_branch"
[new_branch 0d05a66] add file new_branch_file in new_branch
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 new_branch_file

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git status
On branch new_branch
nothing to commit, working tree clean

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ ls
first.txt new_branch_file second.txt third.txt

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$
```

Переключение в master

```
MINGW64:/d/lessons/git
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ touch new_branch_file

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git status
On branch new_branch
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    new_branch_file

nothing added to commit but untracked files present (use "git add" to track)

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git add .

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git commit -m "add file new_branch_file in new_branch"
[new_branch 0d05a66] add file new_branch_file in new_branch
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 new_branch_file

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git status
On branch new_branch
nothing to commit, working tree clean

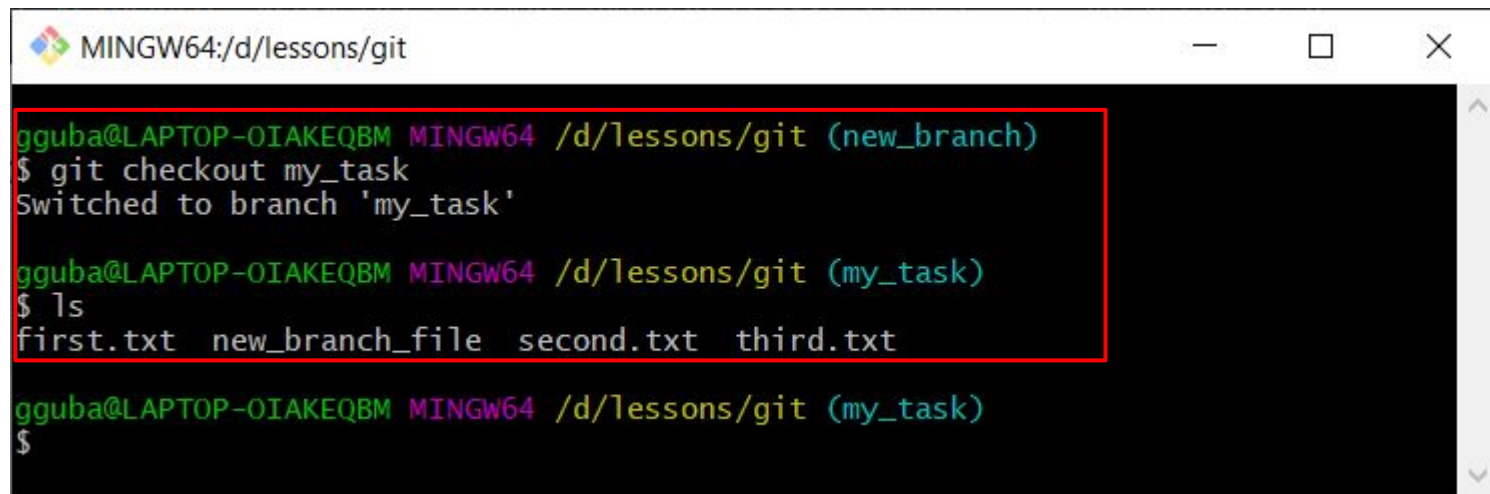
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ ls
first.txt new_branch_file second.txt third.txt

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git checkout master
Switched to branch 'master'

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ ls
first.txt second.txt third.txt

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$
```

Файлы

A screenshot of a MINGW64 terminal window. The title bar shows the MINGW64 logo and the path "/d/lessons/git". The terminal has a black background with green and white text. A red rectangle highlights the first two command sequences. The first sequence shows a 'git checkout' command being executed, switching to a new branch. The second sequence shows a 'ls' command being executed, listing the files in the current directory. The third sequence shows the prompt for another command.

```
MINGW64:/d/lessons/git

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git checkout my_task
Switched to branch 'my_task'

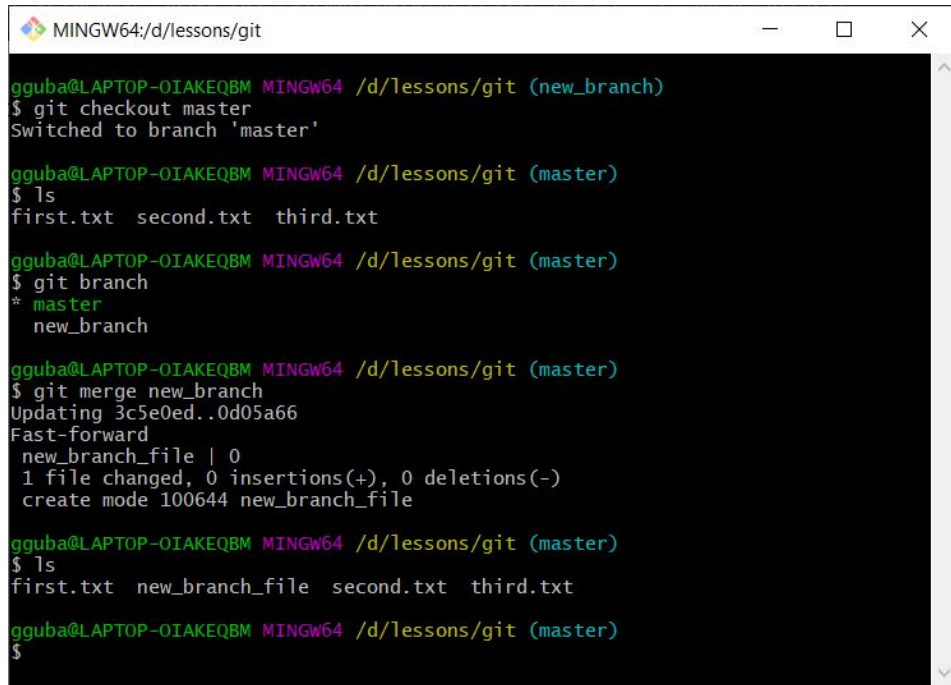
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (my_task)
$ ls
first.txt  new_branch_file  second.txt  third.txt

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (my_task)
$
```

Файлы “копируются” из той ветки от которой вы создали новую

Слияние

Когда вы закончили работать над задачей и хотите влить результат в главную ветку необходимо использовать команду git merge name_branch из ветки В которую вы хотите ВЛИТЬ

A terminal window titled 'MINGW64:/d/lessons/git' showing a series of Git commands and their outputs. The user creates a new branch, checks out master, lists files, creates a new branch, and then merges it back into master, showing a fast-forward merge of a new file.

```
MINGW64:/d/lessons/git
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (new_branch)
$ git checkout master
Switched to branch 'master'

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ ls
first.txt  second.txt  third.txt

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git branch
* master
  new_branch

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ git merge new_branch
Updating 3c5e0ed..0d05a66
Fast-forward
 new_branch_file | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 new_branch_file

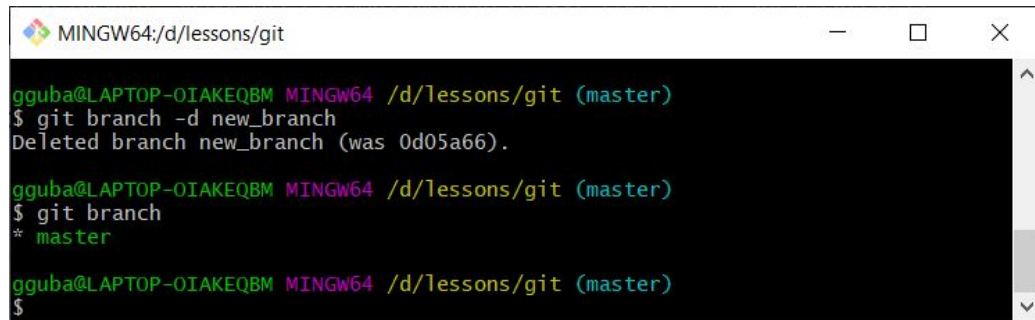
gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$ ls
first.txt  new_branch_file  second.txt  third.txt

gguba@LAPTOP-OIAKEQBM MINGW64 /d/lessons/git (master)
$
```

Удаление ветки

Для удаления ветки используется команда `git branch -d name_branch`

`$ git branch -d name_branch`

A screenshot of a Windows terminal window titled 'MINGW64:/d/lessons/git'. The window has standard Windows window controls (minimize, maximize, close) in the top right corner. The terminal output shows a user named 'gguba' on a machine named 'LAPTOP-OIAKEQBM' in a 'MINGW64' environment, located at '/d/lessons/git'. The user is on the 'master' branch. They enter the command 'git branch -d new_branch', and the terminal responds with 'Deleted branch new_branch (was 0d05a66)'. Then, they enter 'git branch', and the terminal shows '* master'. Finally, they enter another '\$' prompt, and the terminal shows '\$'.