

## GIT HW

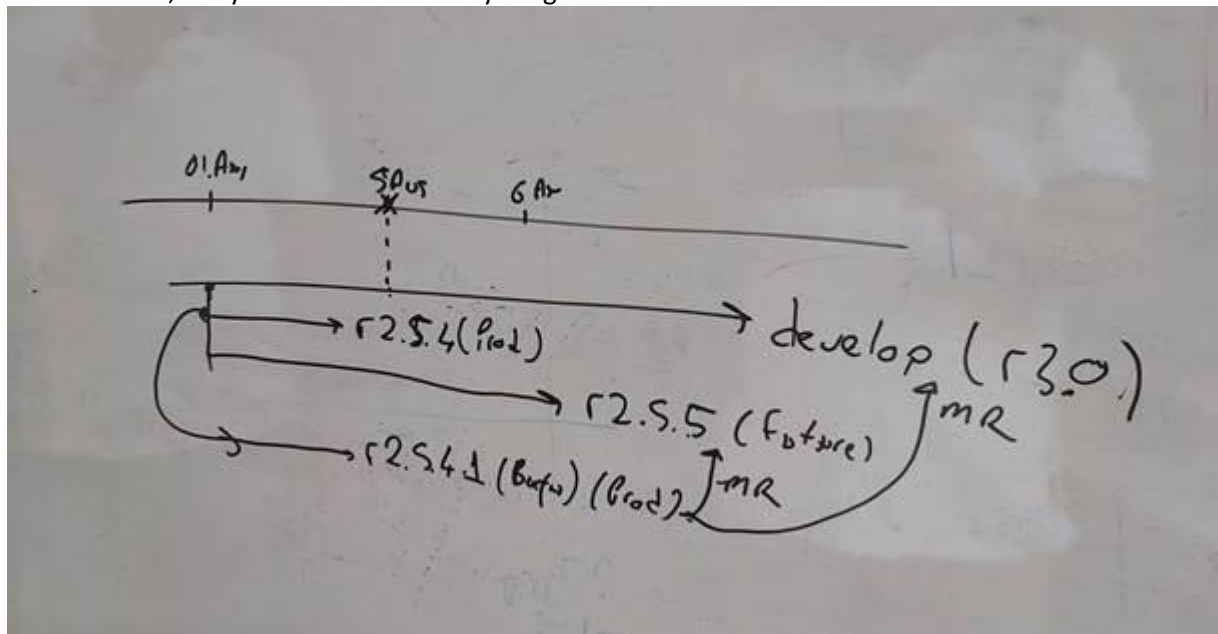
### Definition

In this exercise you will use git.

#### Scenario:

Our product's r2.5.4 version is being used in production and we have r2.5.5 that will be released in the future. Also there is r3.0 version which is in the development process. Suddenly an error occurred in r2.5.4 and we are creating a branch called r2.5.4.1 Bugfix in order to solve this. Then we add those changes to our future branch and development branch.

Now, carry out this scenario in your git environment.



#### Objectives :

- To learn how to use git.
- To understand importance of version control.
- To carry out a real world scenario

**Exercise Keywords:** git, version control, merge, branch.

## SOLVE

```
MINGW64/c/Users/90555/desktop/GitHW
$ touch i2i.txt
$ git status
On branch master
No commits yet
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    i2i.txt
nothing added to commit but untracked files present (use "git add" to track)
$ git add .
$ git commit -m "first commit by v2.5.4"
[master (root-commit) 29ed978] first commit by v2.5.4
1 file changed, 1 insertion(+)
 create mode 100644 i2i.txt
$ git log
commit 29ed978c3d33a4b62da830d99cbb68b35fdc99 (HEAD -> master)
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:38:31 2023 +0300

    first commit by v2.5.4
$ git add .
$ git commit -m "second commit by v2.5.4"
[master c2813e4] second commit by v2.5.4
1 file changed, 3 insertions(+), 1 deletion(-)
$ git log
commit c2813e4415d6a3177ee6469d297347a2cd7a6db3 (HEAD -> master)
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:39:16 2023 +0300

    second commit by v2.5.4
commit 29ed978c3d33a4b62da830d99cbb68b35fdc99
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:38:31 2023 +0300

    first commit by v2.5.4
$
```

i2i.txt - Not Defteri

Dosya Düzen Biçim Görünüm Yardım

first commit by v2.5.4

second commit by v2.5.4

```
MINGW64~/c:/Users/90555/desktop/GitHW
$ git branch
* master

$ git branch future

$ git switch guture
fatal: invalid reference: guture

$ git switch future
Switched to branch 'future'

$ git status
On branch future
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:   i2i.txt

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

$ git add .

$ git commit -m "third commit by v2.5.5"
[future da7efaf] third commit by v2.5.5
1 file changed, 4 insertions(+), 1 deletion(-)

$ git log
commit da7efaf220021a5715df06cd7c4e8e8693751 (HEAD -> future)
Author: Okan Cezik <okancezik@gmail.com>
Date: Sun Jul 30 18:41:13 2023 +0300

    third commit by v2.5.5

commit c2813e4415d6a3177ee6469d297347a2cd7a6db3 (master)
Author: Okan Cezik <okancezik@gmail.com>
Date: Sun Jul 30 18:39:16 2023 +0300

    second commit by v2.5.4

commit 29ed9787c3d33a4b62da830d99cbb68b35fdc99
Author: Okan Cezik <okancezik@gmail.com>
Date: Sun Jul 30 18:38:31 2023 +0300

    first commit by v2.5.4

$
```

i2i.txt - Not Defteri

Dosya Düzen Biçim Görünüm Yardım

first commit by v2.5.4

second commit by v2.5.4

third commit by v2.5.5

```
MINGW64~/c/Users/90555/desktop/GitHW
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (future)
$ git switch master
Switched to branch 'master'
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (master)
$ git branch
  future
* master
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (master)
$ git branch develop
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (master)
$ git switch develop
Switched to branch 'develop'
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (develop)
$ git status
On branch develop
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:   izi.txt
no changes added to commit (use "git add" and/or "git commit -a")
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (develop)
$ git add .
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (develop)
$ git commit -m "fourth commit by v3.0.0"
[develop a3f3ff6] fourth commit by v3.0.0
1 file changed, 3 insertions(+), 1 deletion(-)
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (develop)
$ git log
commit a3f3ff63964452cfac63eefcd889b45d42af417 (HEAD -> develop)
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:43:22 2023 +0300

    fourth commit by v3.0.0
commit c2813e4415d6a3177ee6469d297347a2cd7a6db3 (master)
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:39:16 2023 +0300

    second commit by v2.5.4
commit 28ed9787c3d33e4b62de6830d99cbb68b35fdc99
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:38:31 2023 +0300

    first commit by v2.5.4
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (develop)
$
```

izi.txt - Not Deferi

Dosya Düzen Biçim Görünüm Yardım

first commit by v2.5.4

second commit by v2.5.4

fourth commit by v3.0.0

```
MINGW64~/c/Users/90555/desktop/GitHW
$ git checkout 29ed9787c3d33a4b62daf830d99cbb68b35fdc99
Note: switching to '29ed9787c3d33a4b62daf830d99cbb68b35fdc99'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

  git switch -c <new-branch-name>

Or undo this operation with:

  git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 29ed978 first commit by v2.5.4
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW ((29ed978...))
$ git branch
* HEAD detached at 29ed978
  develop
  future
  master
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW ((29ed978...))
$ git branch bugfix
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW ((29ed978...))
$ git switch bugfix
Switched to branch 'bugfix'
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (bugfix)
$ git add
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (bugfix)
$ git commit -m "fixed bugs"
[bugfix 29aac6a] fixed bugs
1 file changed, 4 insertions(+), 1 deletion(-)
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (bugfix)
$ git log
commit 29aac6ad5c9252f8bd29436034c1e70a46f3cfe5 (HEAD -> bugfix)
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:45:20 2023 +0300

    Fixed bugs

commit 29ed9787c3d33a4b62daf830d99cbb68b35fdc99
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:38:31 2023 +0300

    first commit by v2.5.4
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (bugfix)
$
```

i2i.txt - Not Defteri

Dosya Düzen Biçim Görünüm Yardım

first commit by v2.5.4

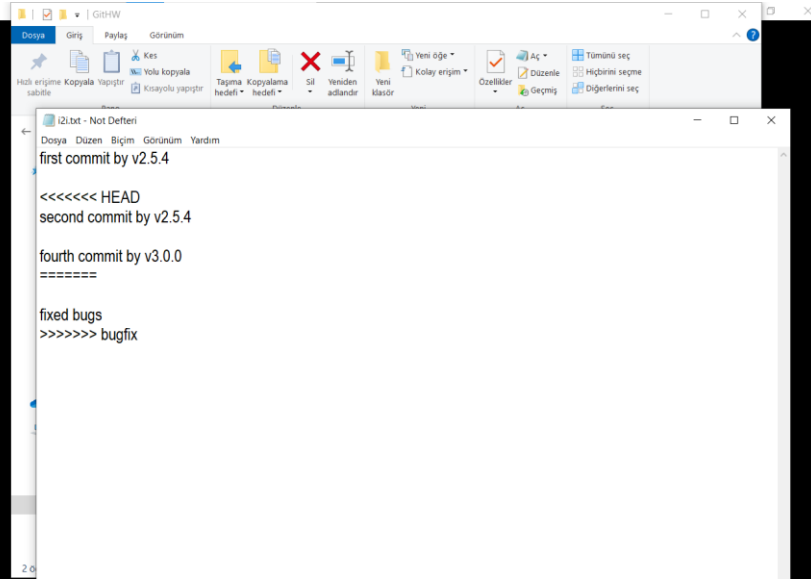
fixed bugs

```
MINGW64~/Users/90555/desktop/GitHW
$ git branch
* bugfix
develop
future
master

MINGW64~/Users/90555/desktop/GitHW (bugfix)
$ git switch develop
Switched to branch 'develop'

MINGW64~/Users/90555/desktop/GitHW (develop)
$ git merge bugfix
Auto-merging i2i.txt
CONFLICT (content): Merge conflict in i2i.txt
Automatic merge failed; fix conflicts and then commit the result.

MINGW64~/Users/90555/desktop/GitHW (develop|MERGING)
$
```



```
MINGW64~/Users/90555/desktop/GitHW
CONFLICT (content): Merge conflict in i2i.txt
Automatic merge failed; fix conflicts and then commit the result.

MINGW64~/Users/90555/desktop/GitHW (develop|MERGING)
$ git status
On branch develop
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:   i2i.txt

no changes added to commit (use "git add" and/or "git commit -a")
MINGW64~/Users/90555/desktop/GitHW (develop|MERGING)
$ git add .

MINGW64~/Users/90555/desktop/GitHW (develop|MERGING)
$ git commit -m "merged with bugfix"
[develop d481ed4] merged with bugfix

MINGW64~/Users/90555/desktop/GitHW (develop)
$ git log
commit d481ed46f87b33b5d2f8ac59e79570de4ffb7a2e (HEAD -> develop)
Merge: a3f3ff6 29aac6a
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:46:42 2023 +0300

    merged with bugfix

commit 29aac6ad5c9252f8bd29436034c1e70a46f3cfe5 (bugfix)
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:45:20 2023 +0300

    fixed bugs

commit a3f3ff63964452cfae63eefcd889b45d42af417
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:43:22 2023 +0300

    fourth commit by v3.0.0

commit c2813e4415d6a3177ee6469d297347a2cd7a6db3 (master)
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:39:16 2023 +0300

    second commit by v2.5.4

commit 29e99787c3d33a4b62da830d99cbb68b35fdc99
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:38:31 2023 +0300

    first commit by v2.5.4

MINGW64~/Users/90555/desktop/GitHW (develop)
$
```

i2i.txt - Not Defteri

Dosya Düzen Biçim Görünüm Yardım

first commit by v2.5.4

fixed bugs

```
MINGW64 ~/Users/ACONRLO5/desktop/GitHW
$ git log
commit d87efa3f2f20021a5715df06cd7c44be8693751 (HEAD -> future)
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:41:13 2023 +0300

    third commit by v2.5.5

commit c2813e4415d6a3177ae6469d797347a2cd7a6db3 (master)
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:39:16 2023 +0300

    second commit by v2.5.4

commit 29ed9787c3d3b4b62da7830d99cbb6b15fdc99
Author: Okan Cezik <okacezik@gmail.com>
Date: Sun Jul 30 18:38:31 2023 +0300

    first commit by v2.5.4

MINGW64 ~/Users/ACONRLO5/desktop/GitHW (future)
$ git merge bugfix
Auto-merging i2i.txt
CONFLICT (content): Merge conflict in i2i.txt
Automatic merge failed; fix conflicts and then commit the result.
MINGW64 ~/Users/ACONRLO5/desktop/GitHW (future/MERGING)
$
```

first commit by v2.5.4

<<<<<< HEAD  
second commit by v2.5.4

third commit by v2.5.5  
=====

fixed bugs  
>>>>>> bugfix

```
MINGW64/c/Users/90555/desktop/GitHW
((fix conflicts and run "git commit")
 (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <files>..." to mark resolution)
    both modified:   i21.txt

no changes added to commit (use "git add" and/or "git commit -a")
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (future)MERGING
$ git add .
$ git commit -m "merged with bugfix"
[future 287fd8d] merged with bugfix
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (future)
$ git log
git: 'log' is not a git command. See 'git --help'.

The most similar command is
  log
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (future)
$ git log
commit 287fd8df2cd9a890787315b5f196f70ada00dc (HEAD -> future)
Merge: da7efa3 29aac6a
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:48:58 2023 +0300

    merged with bugfix

commit 29aac6ad5c9252f8bd29436034c1e70a46f3cfe5 (bugfix)
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:45:20 2023 +0300

    fixed bugs

commit da7efa3f2f20021a5715df06cd77c4ebe8693751
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:41:13 2023 +0300

    third commit by v2.5.5

commit c2813e4415d6a3177ee6469d297347a2cd7a6db3 (master)
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:39:16 2023 +0300

    second commit by v2.5.4

commit 29ed9787c3d3a4b62da830d99cbb68b35fdc99
Author: Okan Cezik <okacezik@gmail.com>
Date:   Sun Jul 30 18:38:31 2023 +0300

    first commit by v2.5.4
okan@LAPTOP-AXXNRL05 MINGW64 ~/desktop/GitHW (future)
$ |
```