software
development
academy

GIT basics

Marcin Chodkowski

software **development** academy

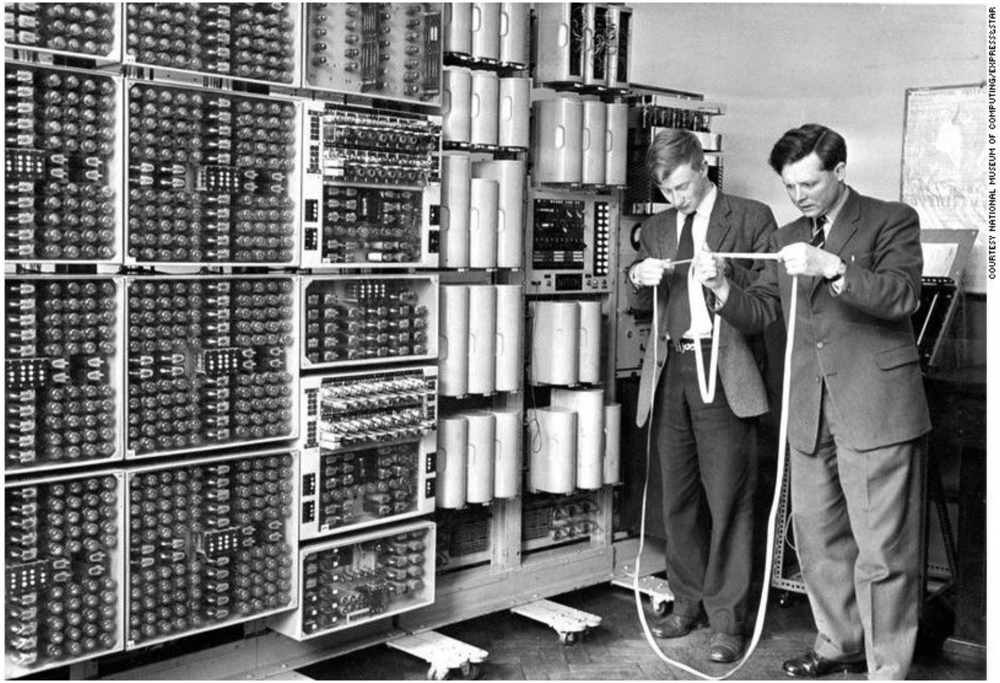
Command line - computer communication

GIT basics

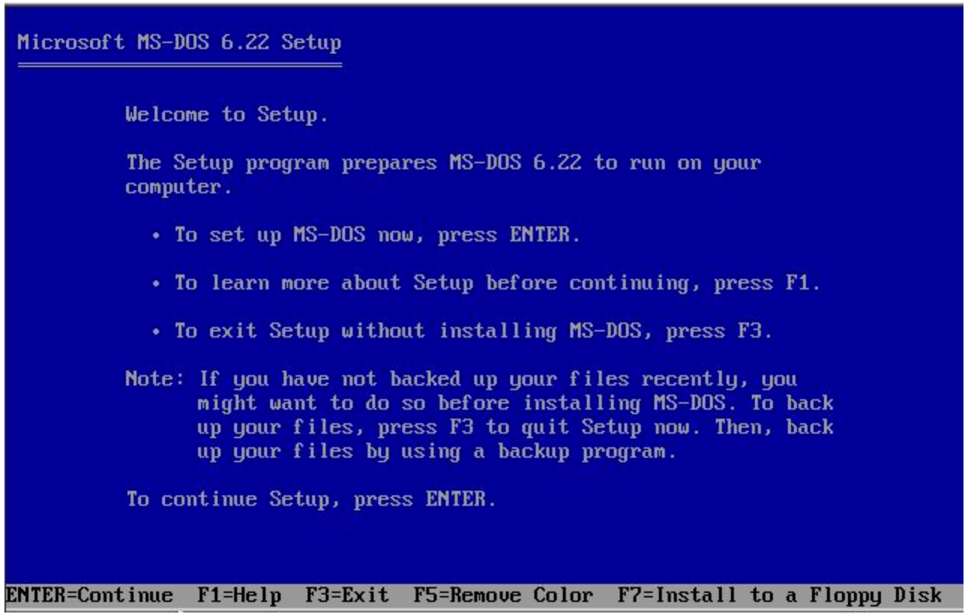
In this video you will learn:

- About the beginnings of communication with a computer.
- What is Git Bash?
- What are the basic command line commands?

Communication with a computer

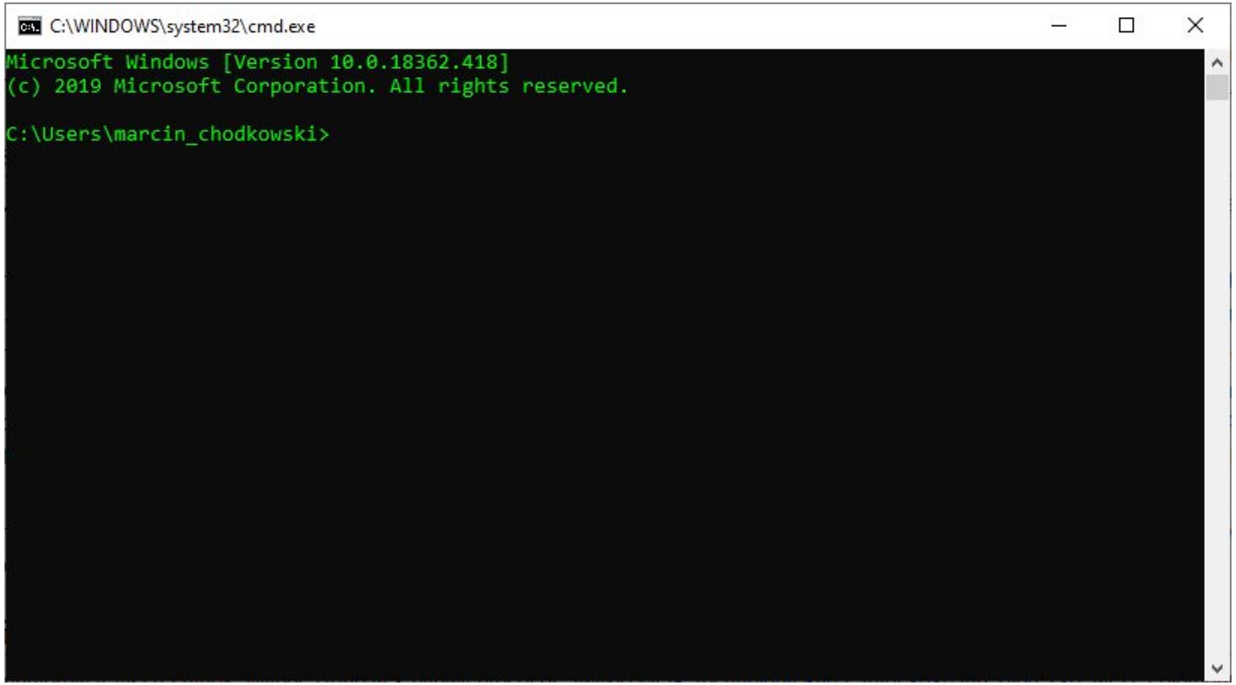


Communication with a computer



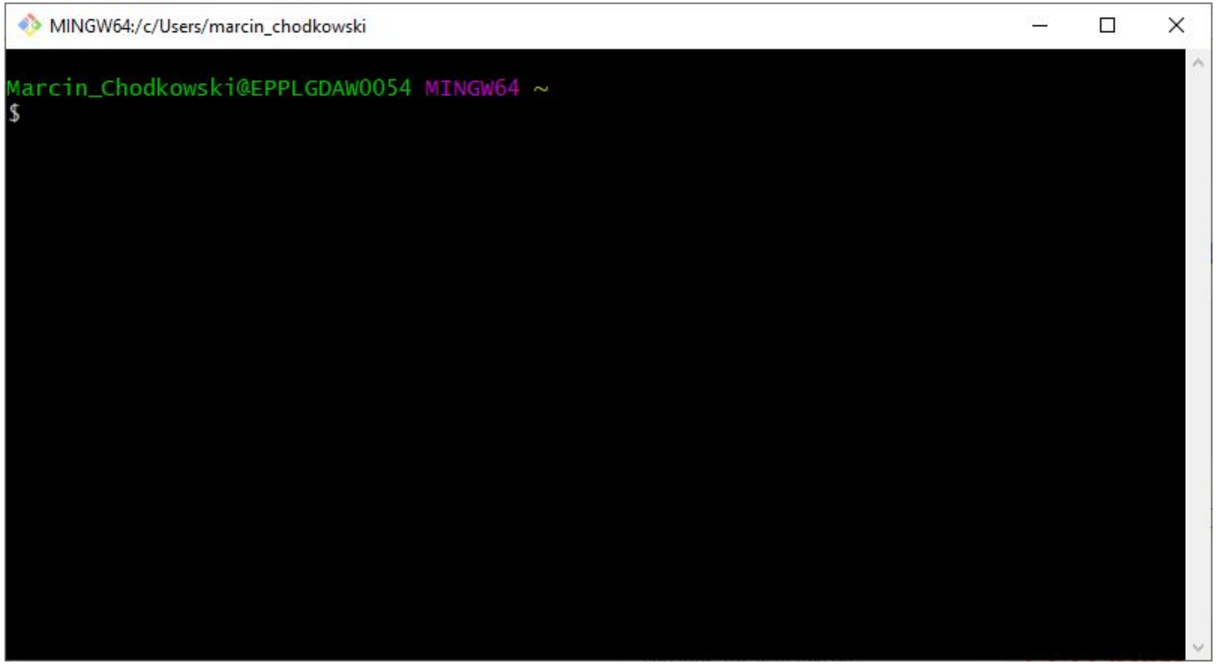
```
Microsoft MS-DOS 6.22 Setup
═══════════════════════════

         Welcome to Setup.

         The Setup program prepares MS-DOS 6.22 to run on your
         computer.

           • To set up MS-DOS now, press ENTER.

           • To learn more about Setup before continuing, press F1.

           • To exit Setup without installing MS-DOS, press F3.

         Note: If you have not backed up your files recently, you
               might want to do so before installing MS-DOS. To back
               up your files, press F3 to quit Setup now. Then, back
               up your files by using a backup program.

         To continue Setup, press ENTER.

ENTER=Continue  F1=Help  F3=Exit  F5=Remove Color  F7=Install to a Floppy Disk
```

Communication with a computer



```
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\marcin_chodkowski>
```

Git bash


MINGW64:/c/Users/marcin_chodkowski

```
Marcin_Chodkowski@EPPLGDAW0054 MINGW64 ~
$
```

Basic commands

pwd	print working directory
ls	print directory contents
mkdir	create a directory
touch	create an empty file
mv	move file
rm	delete file
cp	copy file

After watching this video:

- You know what the command line is.
- You learned the basic commands.

software **development** academy

Command line-instalation

GIT basics

software **development** academy

Command line- demo

GIT basics

software **development** academy

Version control systems - file versioning problem

GIT basics

In this video you will learn:

- What is the file versioning problem and what does it result from?
- What is a version control system?
- What functions does it perform?
- What are the types of version control systems?

File versioning

Version # 1:

Ala.

File versioning

Version # 1:

Ala.

Version # 2:

Ala has a cat and a dog.

File versioning

Version # 1:

Ala.

Version # 2:

Ala has a cat and a dog.

Version # 3:

Ala has a cat.

File versioning

Version # 1:

Ala.

~~Version # 2:~~

~~Ala has a cat and a dog.~~

~~Version # 3:~~

~~Ala has a cat.~~

Version control systems

Definition

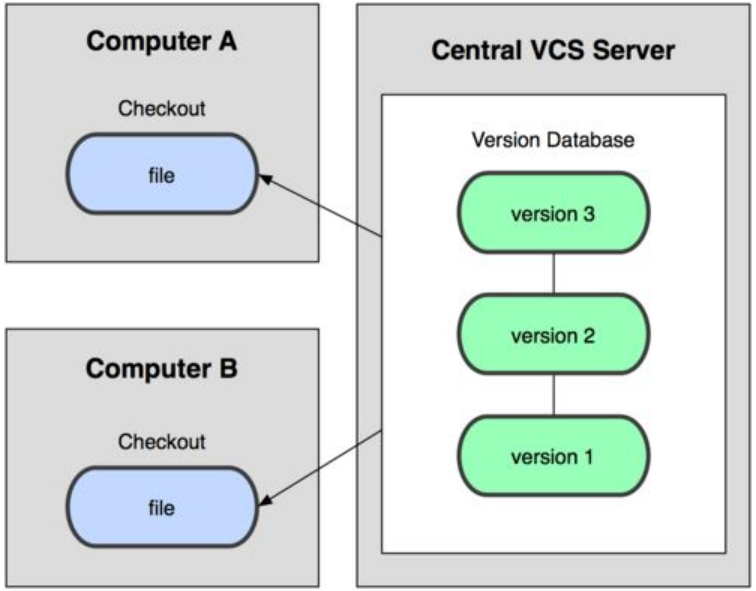
Version control system	software that tracks changes in files also allows you to restore older versions of files and view changes made to them.

Version control systems

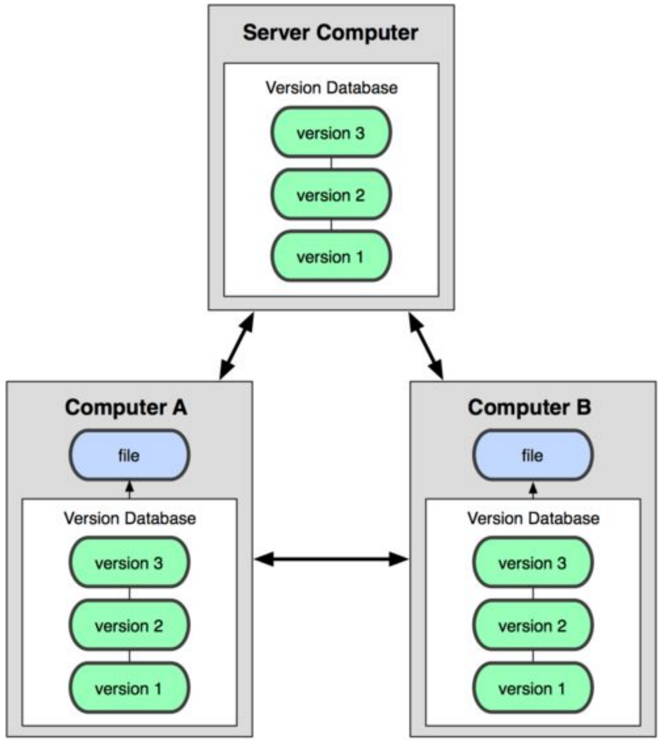
Features of the version control system:

- file change history
- the ability to restore any version of the file
- synchronization of changes introduced by various authors in a distributed environment

Centralized version control systems



Distributed version control systems



Distributed version control systems

Advantages:

- you can make changes without connecting to a remote server
- speed of work - unlike centralized systems, you don't have to communicate with a remote server with every command
- data security - each of the developers has a local copy of the repositories

After watching this video:

- You know the definition of a version control system.
- You know what functions it performs.
- You know the differences between centralized and distributed version control systems.

software **development** academy

What is GIT?

GIT basics

In this video you will learn:

- What is GIT?
- What is characterized by?

GIT

Git is a distributed version control system created in 2005.

It enables us to manage the source code by offering:

- speed
- efficient work with large projects
- full dispersion

GIT

Git has a huge user community.

There are many hosting sites for git repositories.

After watching this video:

- You know the assumptions of GIT.
- You know what you can use it for.
- You know the most popular hosting sites:
    - GitHub
    - GitLab
    - BitBucket

software **development** academy

Creating a local repository

GIT basics

In this video you will learn:

- What is a repository?
- How to create a local repository?

Repository

Definition

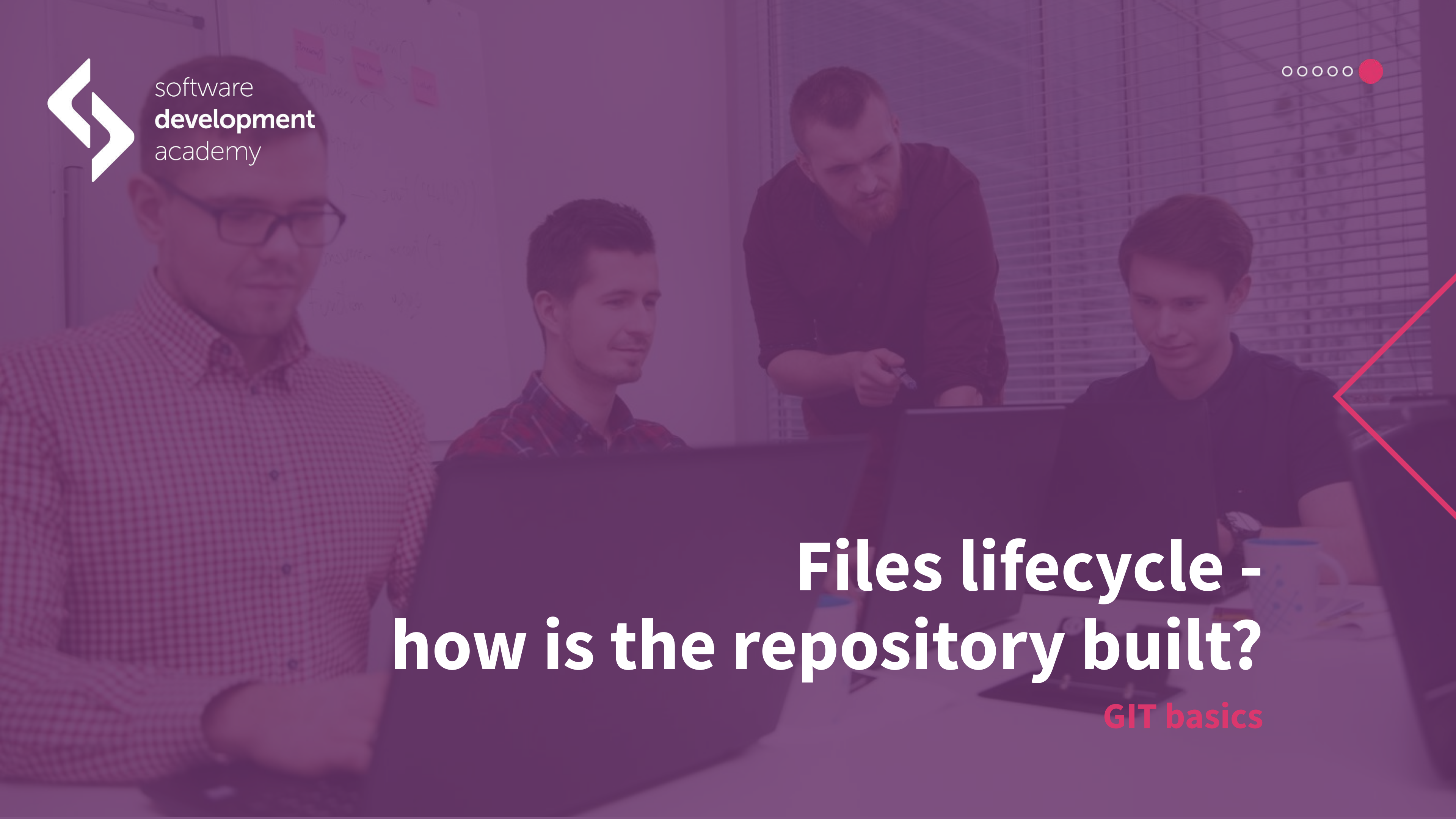
repository	source code and all information about changes made to it.

Creating a repository

git init command

After watching this video:

- You know what a repository is.
- You can create them.



software **development** academy

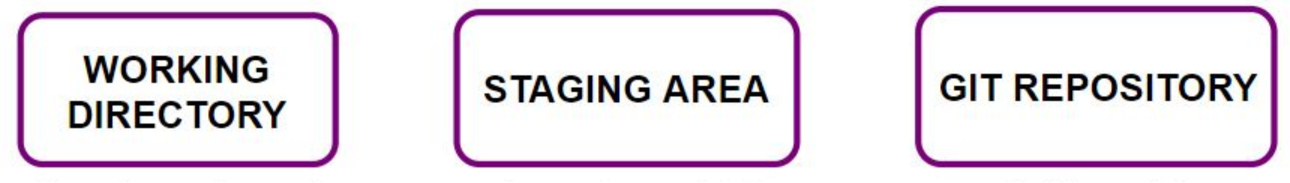
Files lifecycle - how is the repository built?

GIT basics

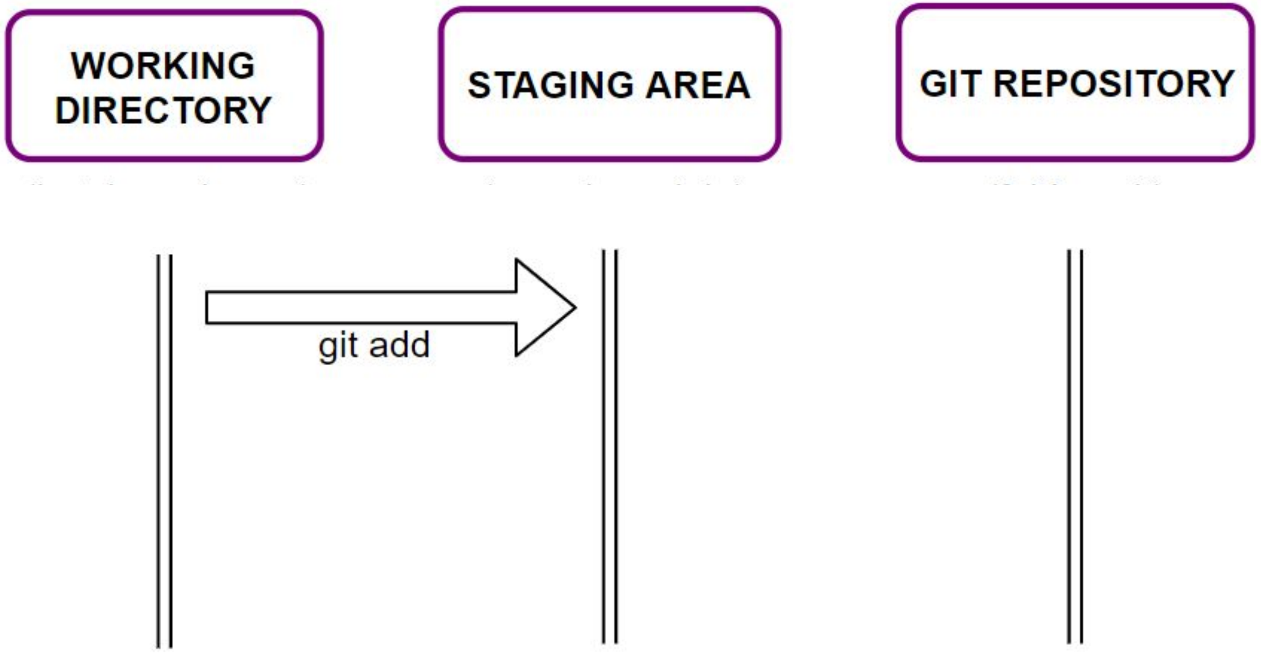
In this video you will learn:

- How is the repository built?
- How does the repository work in practice?
- What are the possible file states?

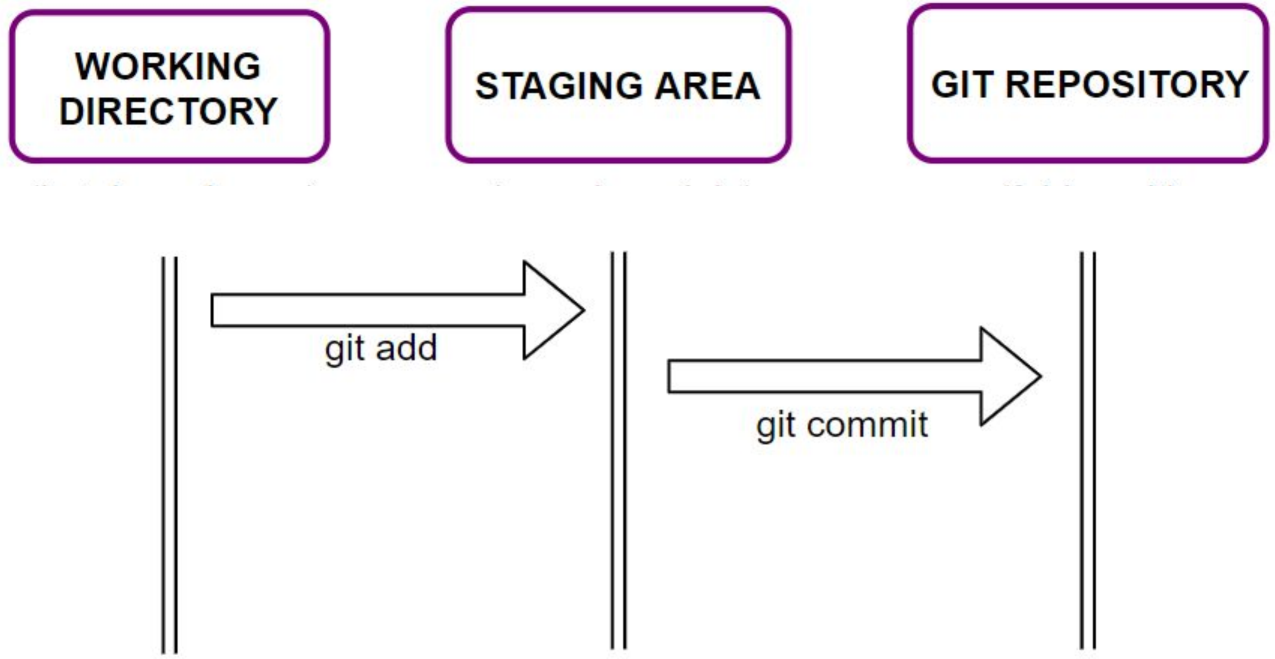
Local repository



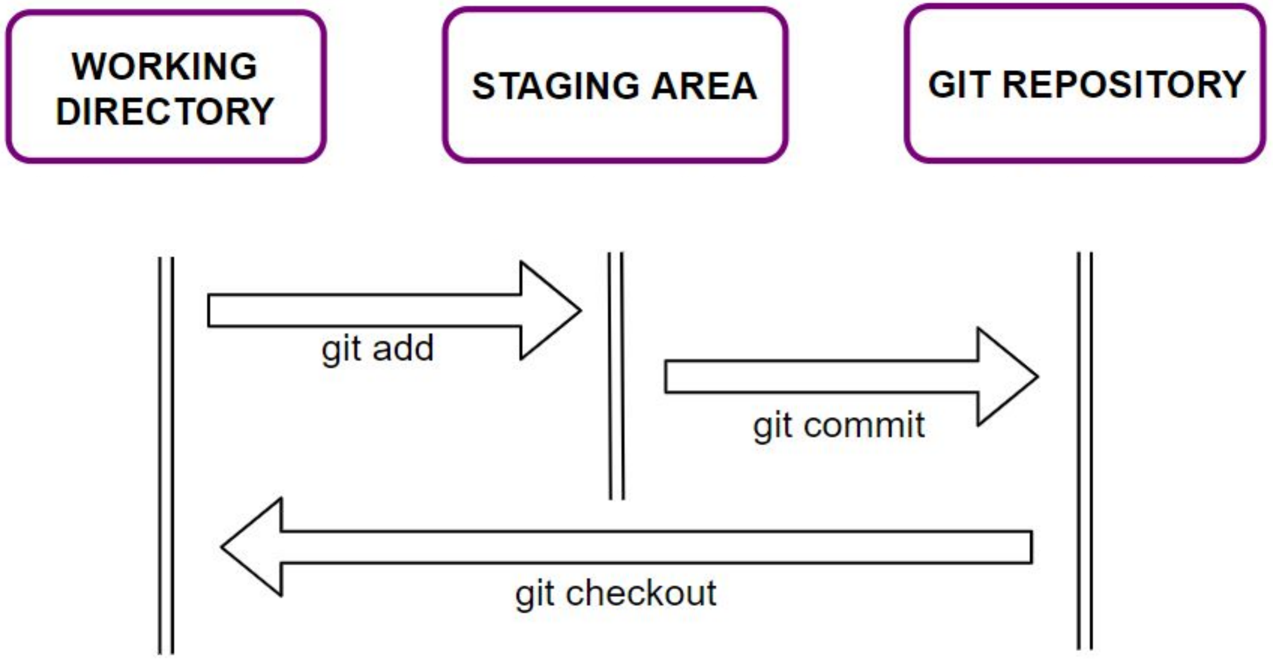
Local repository



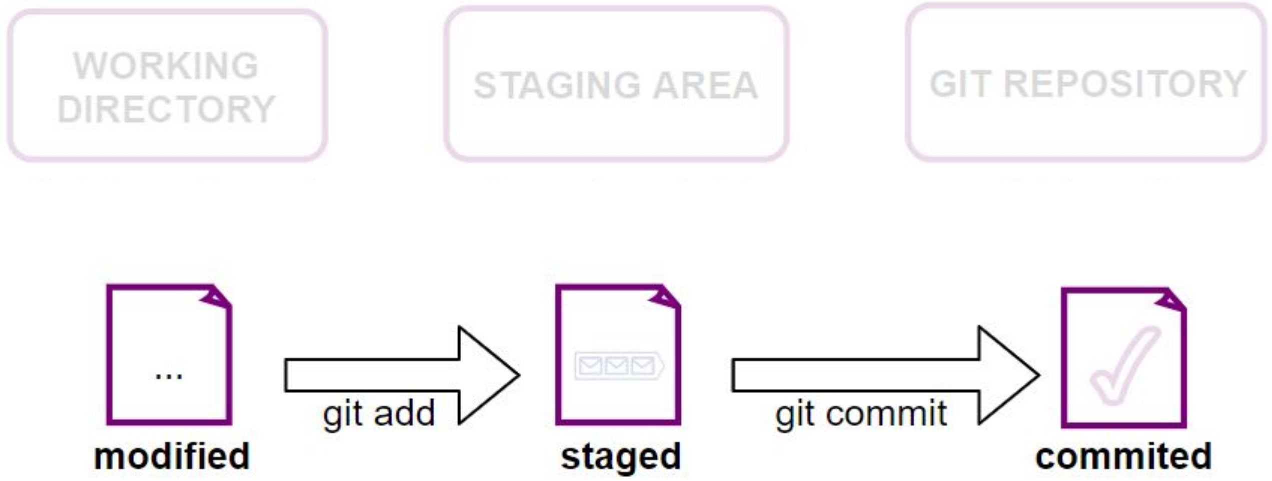
Local repository



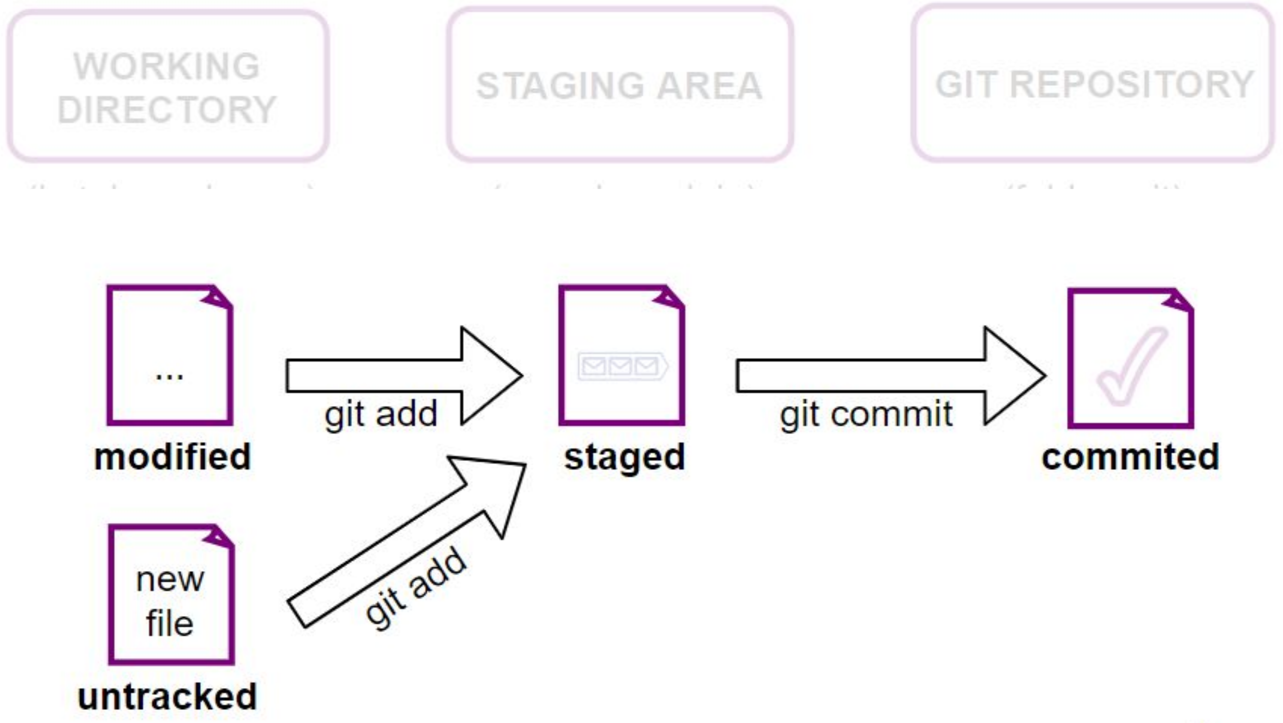
Local repository



Local repository



Local repository



After watching this video:

- You know how the repository is built.
- You got to know its operation in practice.

software **development** academy

Undoing a file's state - how not to create a bad commit?

GIT basics

In this video you will learn:

- How to undo the file status?
- How do **git reset**, **git checkout** and **git rm** work?

Undoing files from staging area

Command	
git reset	(used on file) works inversely to git add, roll back files from staging area to working directory

Reject changes

Command	
git checkout	(used on the file) restores the state of the file to the form saved by git.

Deleting Files

Command	
git rm	deletes the specified file / directory

After watching this video:

You know how the commands work:

- git reset
- git checkout
- git rm

software
development
academy

Change history - how to browse commits?

GIT basics

In this video you will learn:

- What is commit?
- How to view change history?
- How to undo unwanted commits?

What is commit?

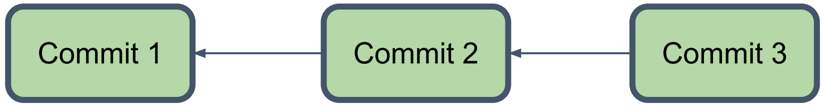
Definition

Commit is a snapshot of changes from the staging area.

The git commit command remembers a project snapshot, along with additional information, including the number of changed files, statistics of added and removed code lines, as well as the SHA-1 checksum of changes.

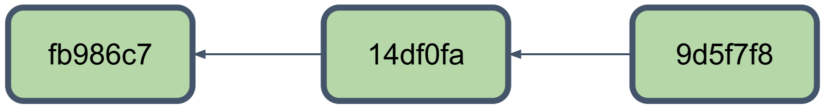
Commit history

After creating three more commits, their history could look like this:



Commit history

Using checksums, the commit history could look like this:



Commit history

Command	
git log	allows you to list the commits in order from newest to oldest

Optional arguments to the **git log** command:

--patch	displays the changes made in each commit
--stat	displays some statistics, e.g. the number of lines changed
--oneline	display each commit on one separate line

Undoing changes

Command	
git revert <commit>	allows reversing changes introduced in a specific commit. This command creates a new commit, which changes the effects of changes introduced in the commit that we want to remove.

After watching this video:

- You know what commit is.
- You learned the **git revert <commit>** command.

software **development** academy

Creating a remote repository - setting remotes

GIT basics

In this video you will learn:

- What is a remote repository?
- How to create them?
- What is remote?
- What does the remote repository creation and hooking look like?

Remote repository

Definition

Remote repository	version of the project kept on a server accessible via the Internet (or other network).

Creating a remote repository

Popular git repository hosting services:

- GitHub
- GitLab
- BitBucket

Remotes

Remote in a git is something similar to a pointer, pointing to another repository from which you may want to download the code or to which you may want to send the code.

Git remote command

Command	
git remote	prints the short names of all specified servers for this repository
git remote -v	prints the short names of all servers specified for this repository. In addition, the URL assigned to these shortcuts will be displayed.

After watching this video:

- You know what a remote repository is and how to create it.
- You know the git remote command.
- You learned how to create and connect to a remote repository.

software **development** academy

.gitignore file - ignoring files

GIT basics

In this video you will learn:

- What is the .gitignore file?
- What is it’s operating range?
- What does it look like in practice?

.gitignore file

Definition

.gitignore file	allows you to define the patterns that git looks for in file paths.

If the file path matches any of the patterns, git ignores this file in all commands.

The range of the .gitignore file

If the .gitignore file is placed in the main project directory, it will refer to the entire repository, while .gitignore files placed in internal folders refer only to the contents of these folders.

After watching this video:

- You know the function of the .gitignore file.
- You know how it is used in practice.

software
development
academy

Creating branches and working with them

GIT basics

In this video you will learn:

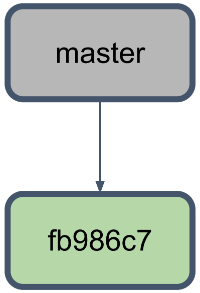
- What is a branch?
- What function does it perform?
- How to create it and how to switch between branches?

What is a branch?

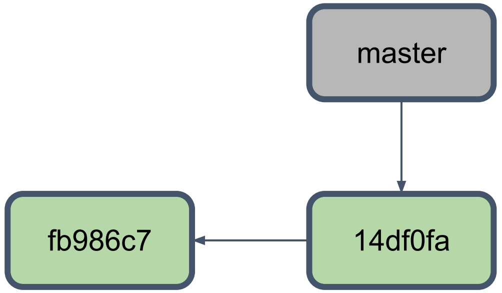
Branch	a ramification in our project where we will continue working. In fact, it is a shifting pointer to one of the sets of changes (commits).

The default branch name in the git is **master**.

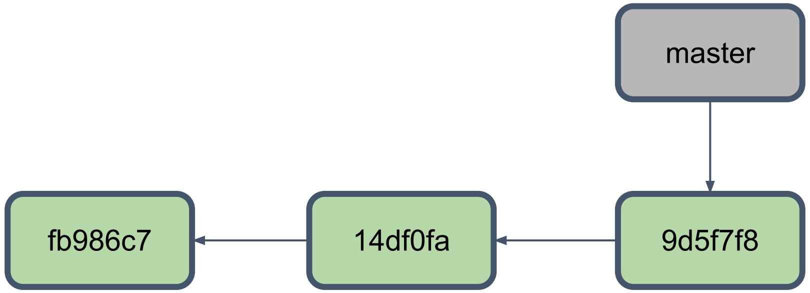
What is a branch?



What is a branch?



What is a branch?

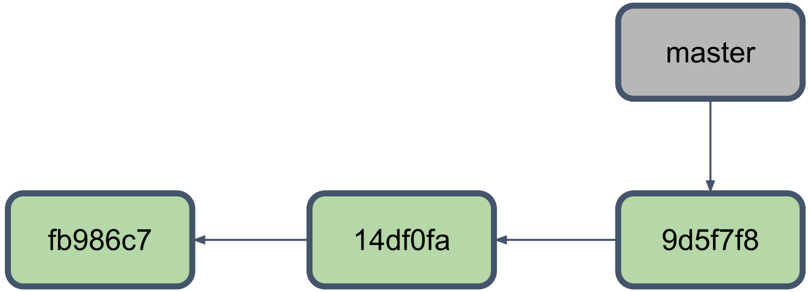


How to create a new branch?

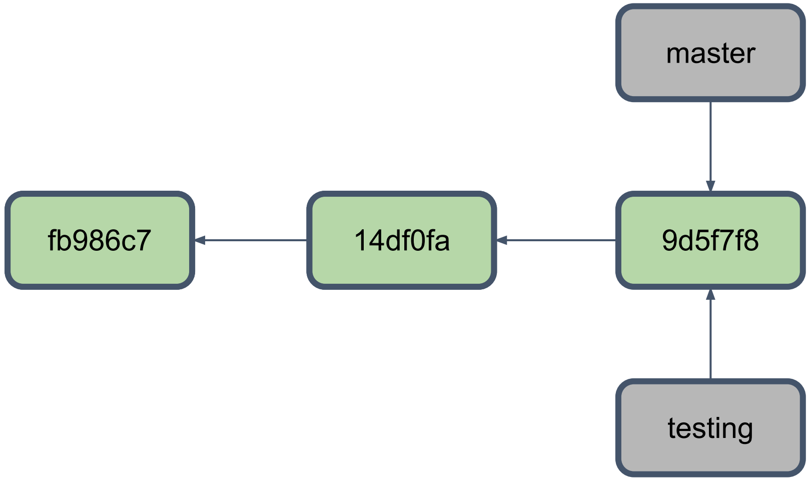
Command

git branch <branch_name>

lets you create a new branch

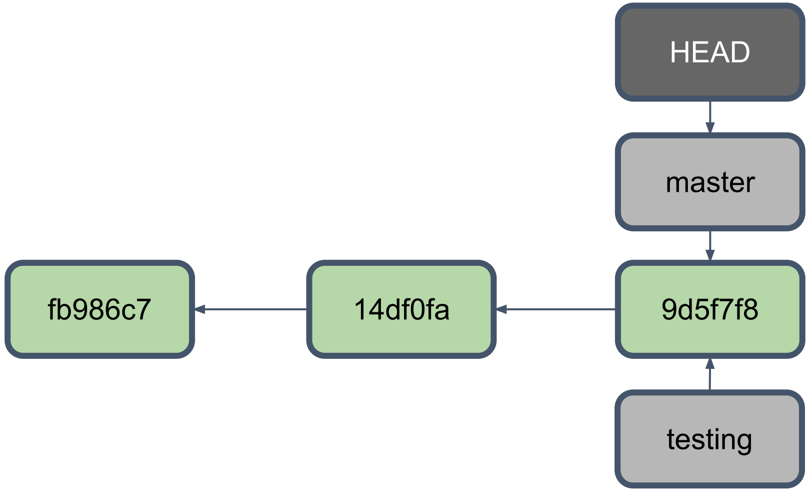


How to create a new branch?



HEAD pointer

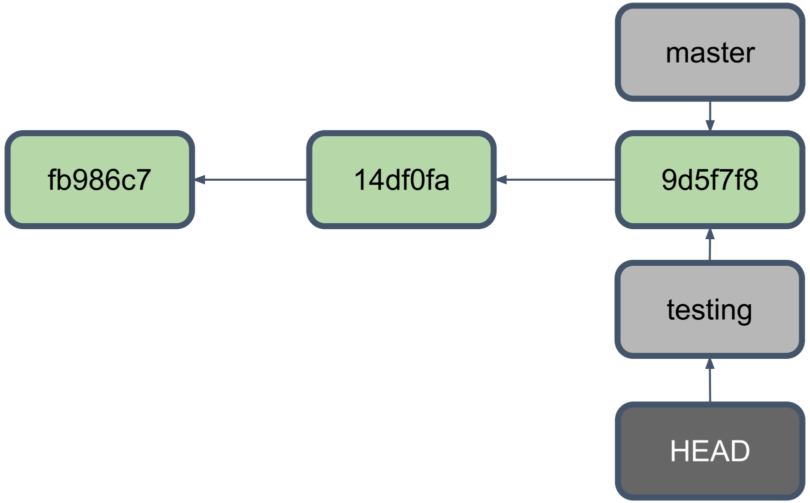
You can say that HEAD is a pointer to the local branch you are on. It is a reference to the last commit created (in a given branch).



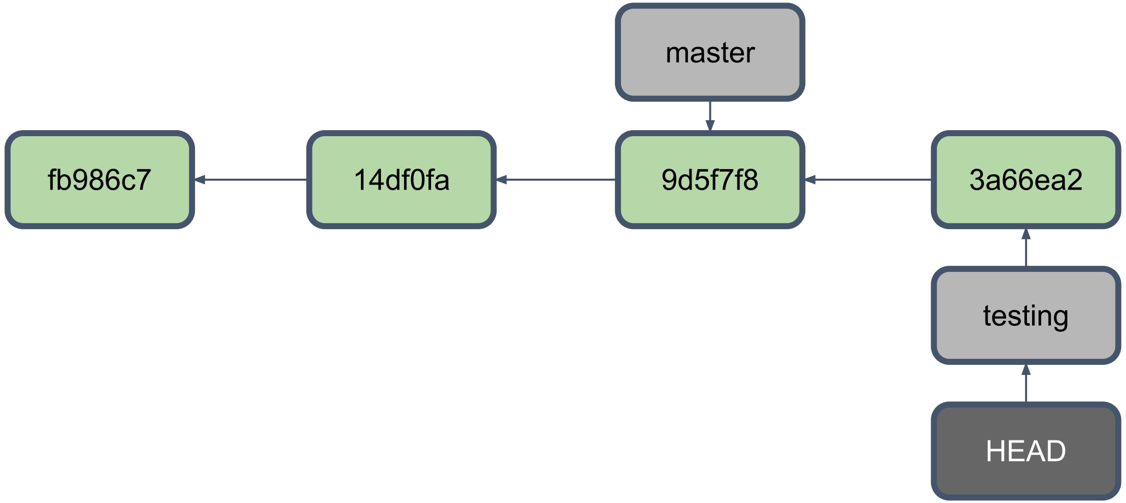
Switching between branches

Command	
git checkout <branch_name>	allows you to switch to the selected branch

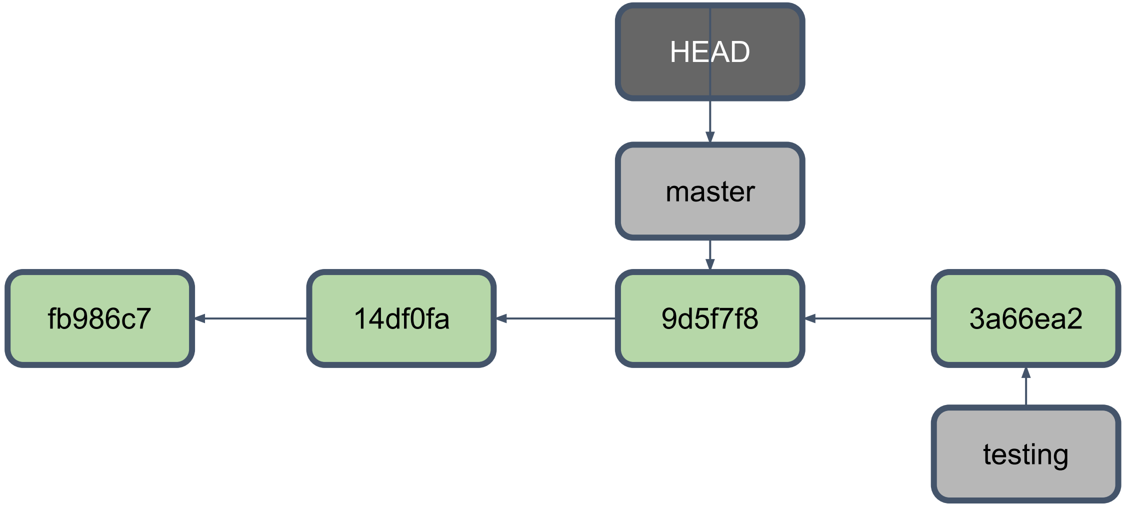
Switching between branches



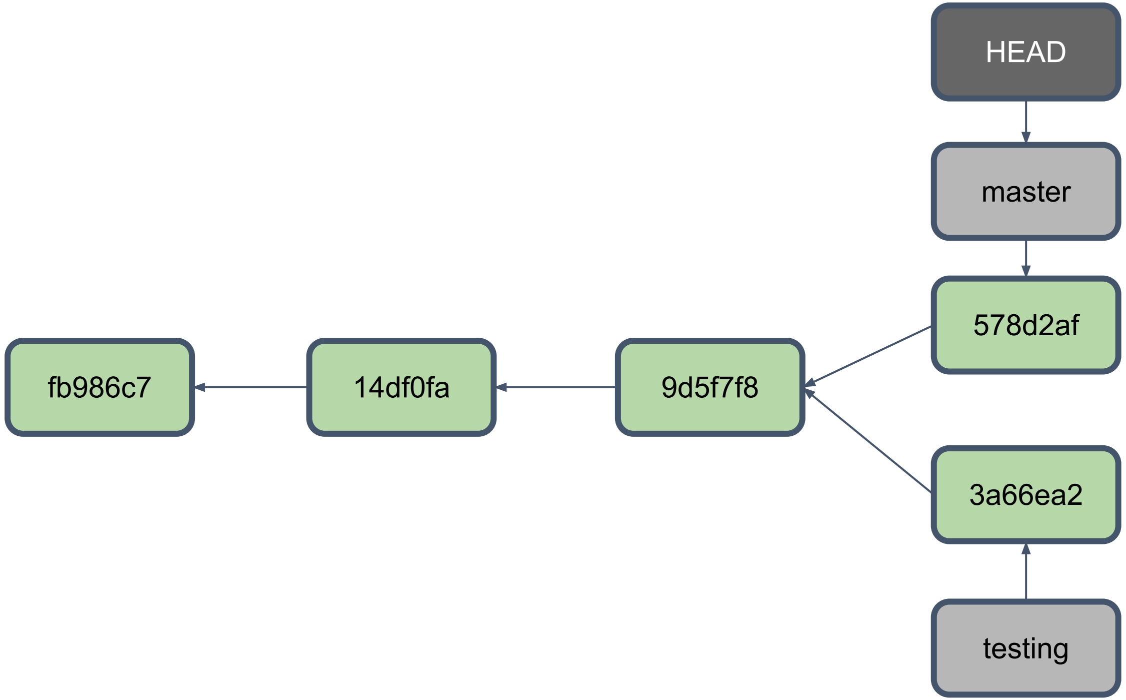
Switching between branches



Switching between branches



Switching between branches



After watching this video:

- You know what branch is.
- You got to know its functions.
- You saw how to create it and how to switch between branches.

software **development** academy

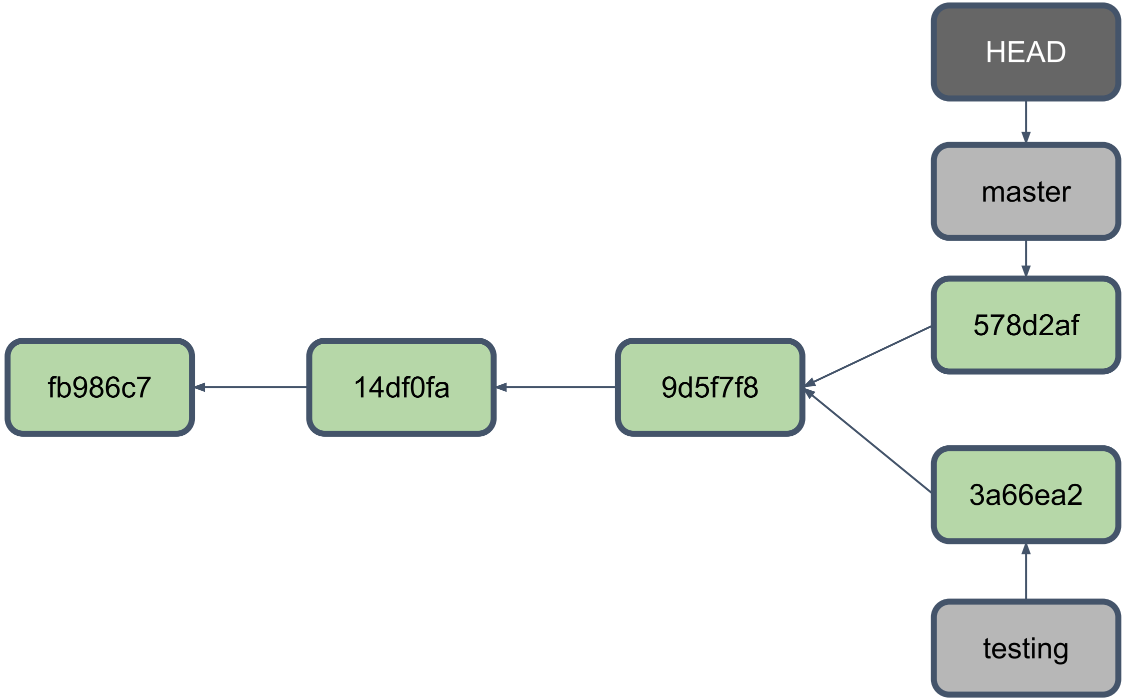
Merging branches

GIT basics

In this video you will learn:

- What is the merging of branches?

Merging branches



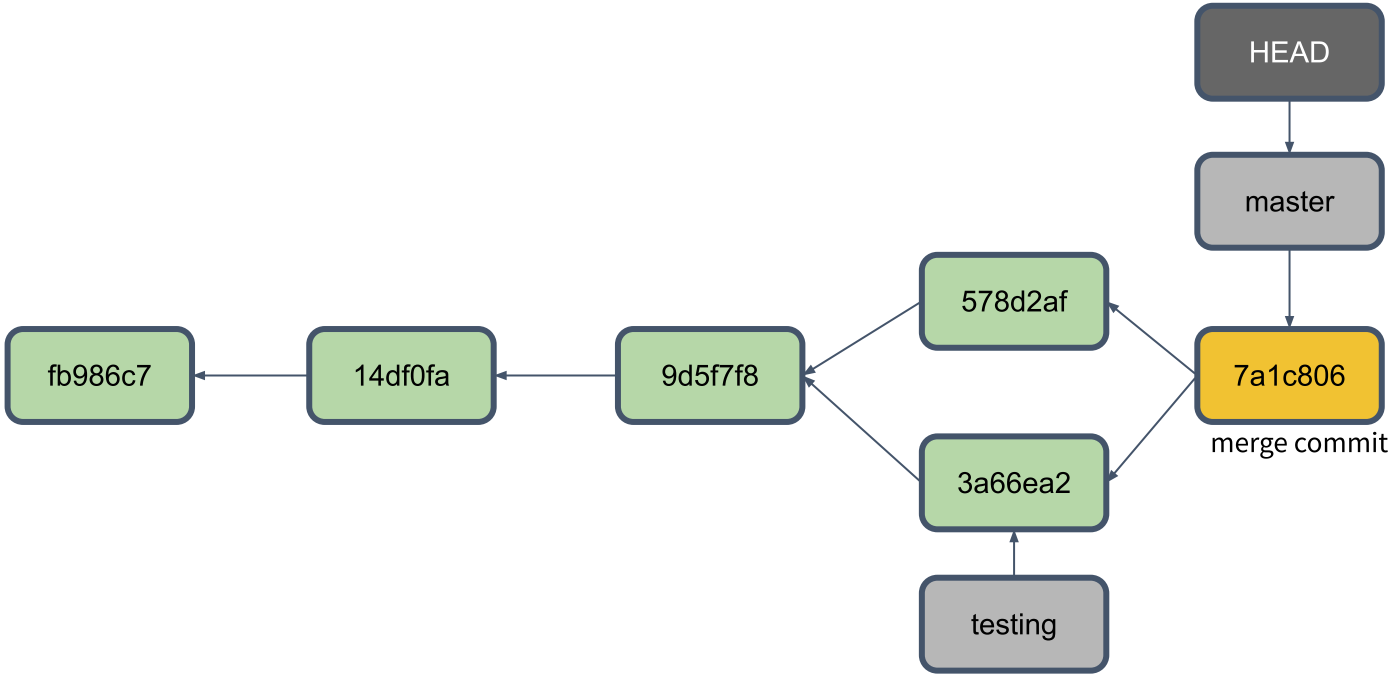
Merging branches

Command

git merge <branch_name>

the branch we are currently in will be updated with the changes from the selected branch.

Merging branches



Branch deleting

Command

git branch-d <branch_name>	delete the selected branch

After watching this video:

- You know how to merge branches.
- You saw how it looks in practice.

software **development** academy

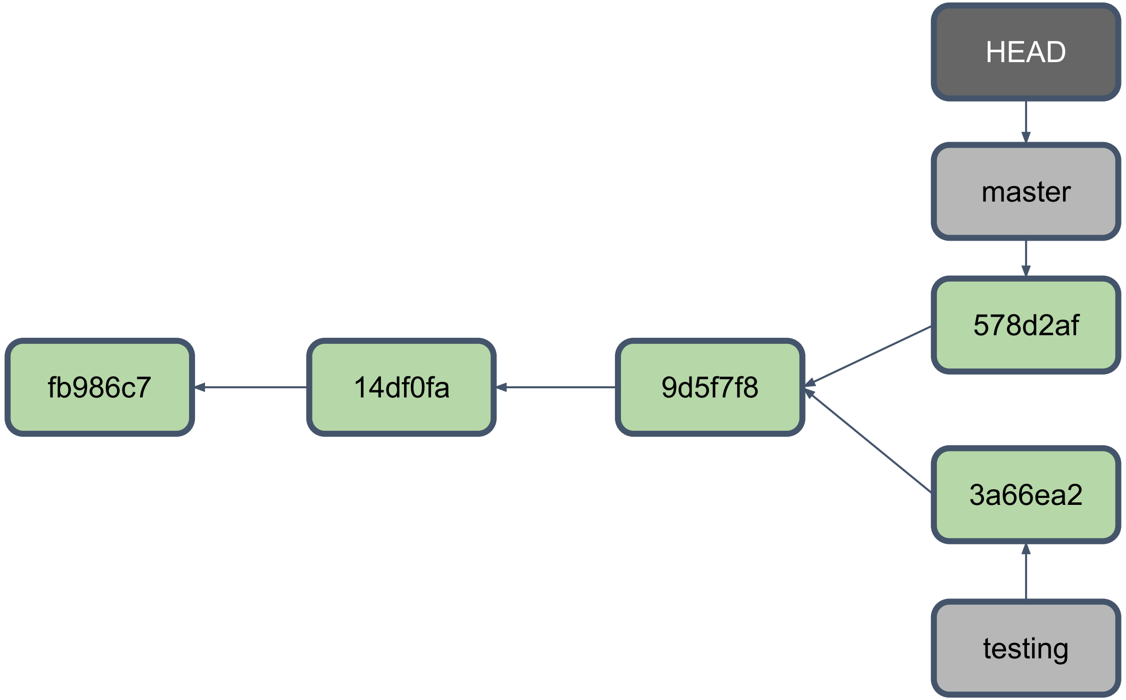
Merge conflicts

GIT basics

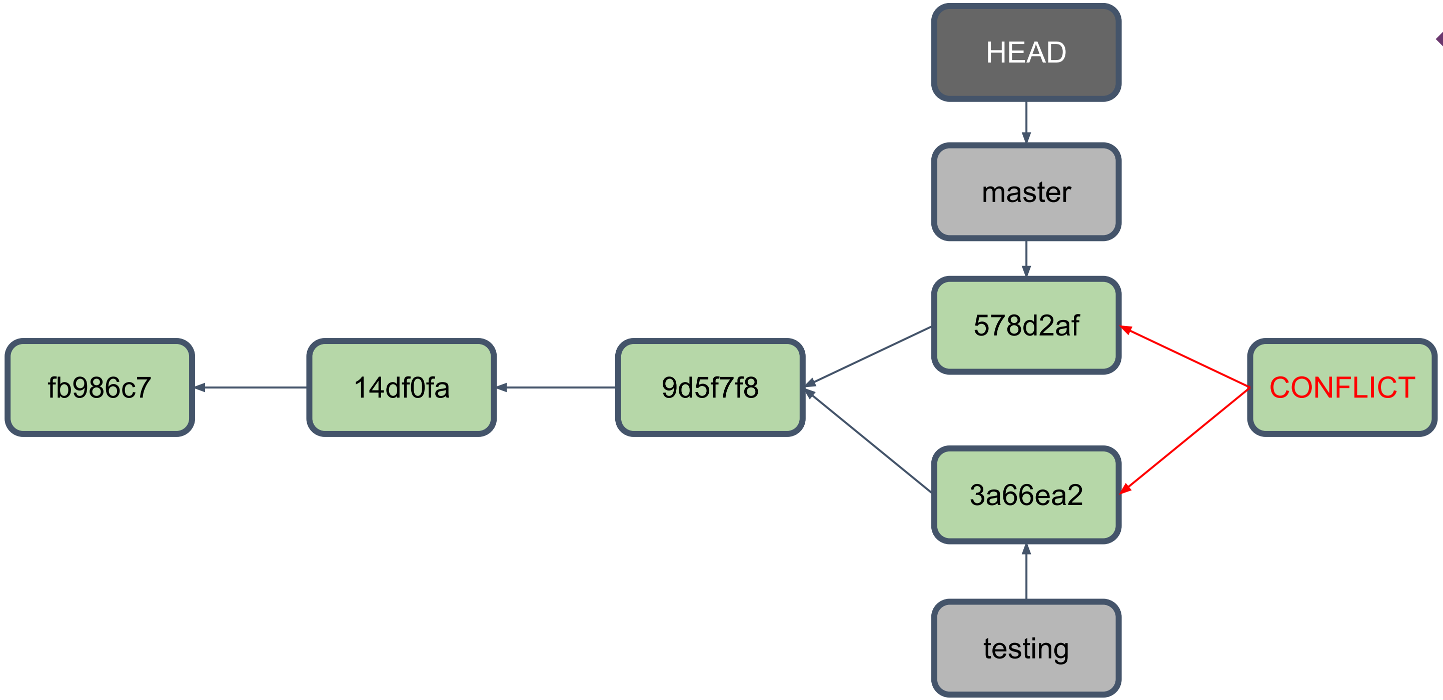
In this video you will learn:

- How do conflicts arise when merging branches?
- How to solve them?

Merge conflicts



Merge conflicts



Solving the conflict

```
<<<<<<< HEAD: fileA
Ala has a cat
=======
Alex has a dog
>>>>>>> testing: fileA
```

Solving the conflict

Alex has a cat

Solving the conflict

Alex has a dog

Solving the conflict

Alex has a cat and a dog

After watching this video:

- You learned in practice how to resolve conflicts between branches.

software
development
academy

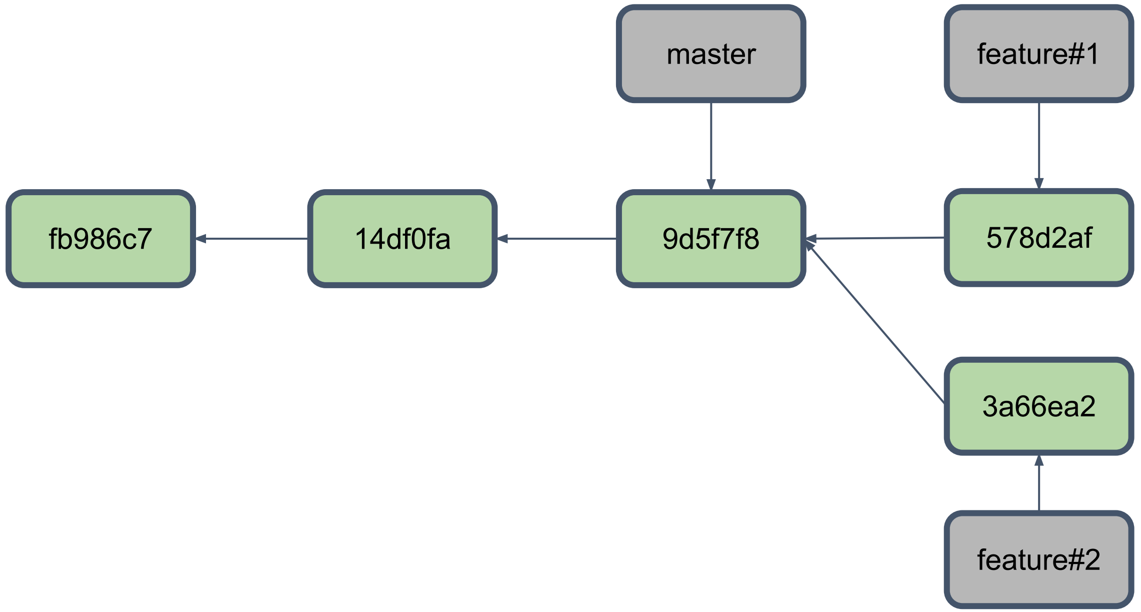
Fast forwarding - default merging strategy

GIT basics

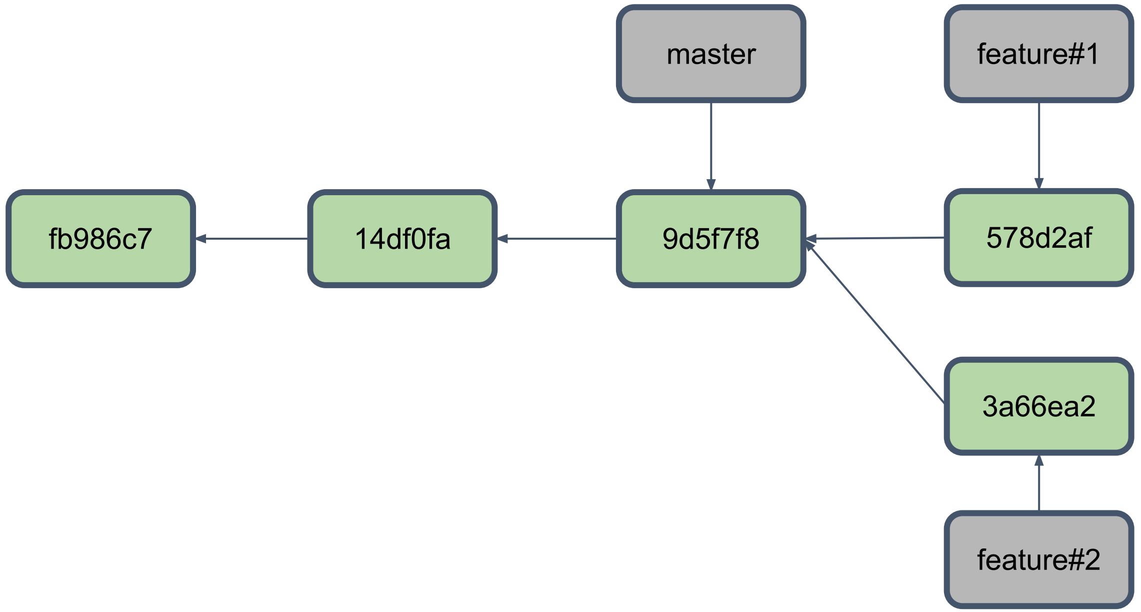
In this video you will learn:

- What is fast forwarding?

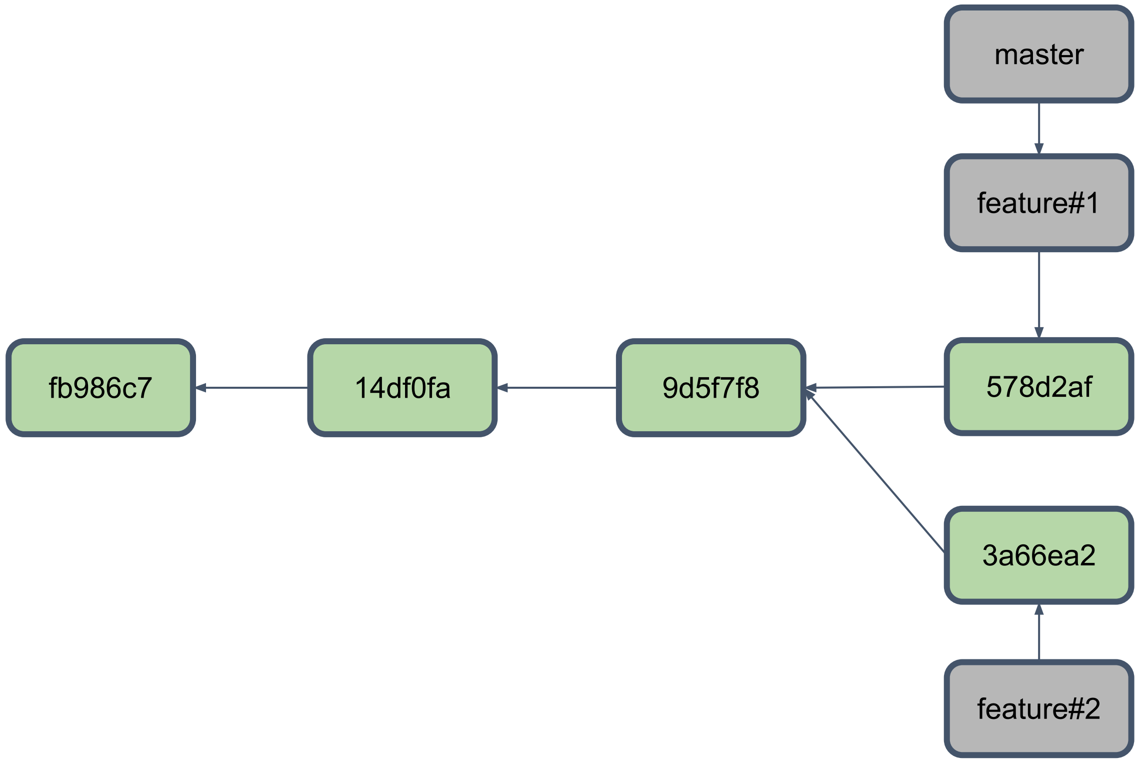
Fast forwarding



Fast forwarding



Fast forwarding



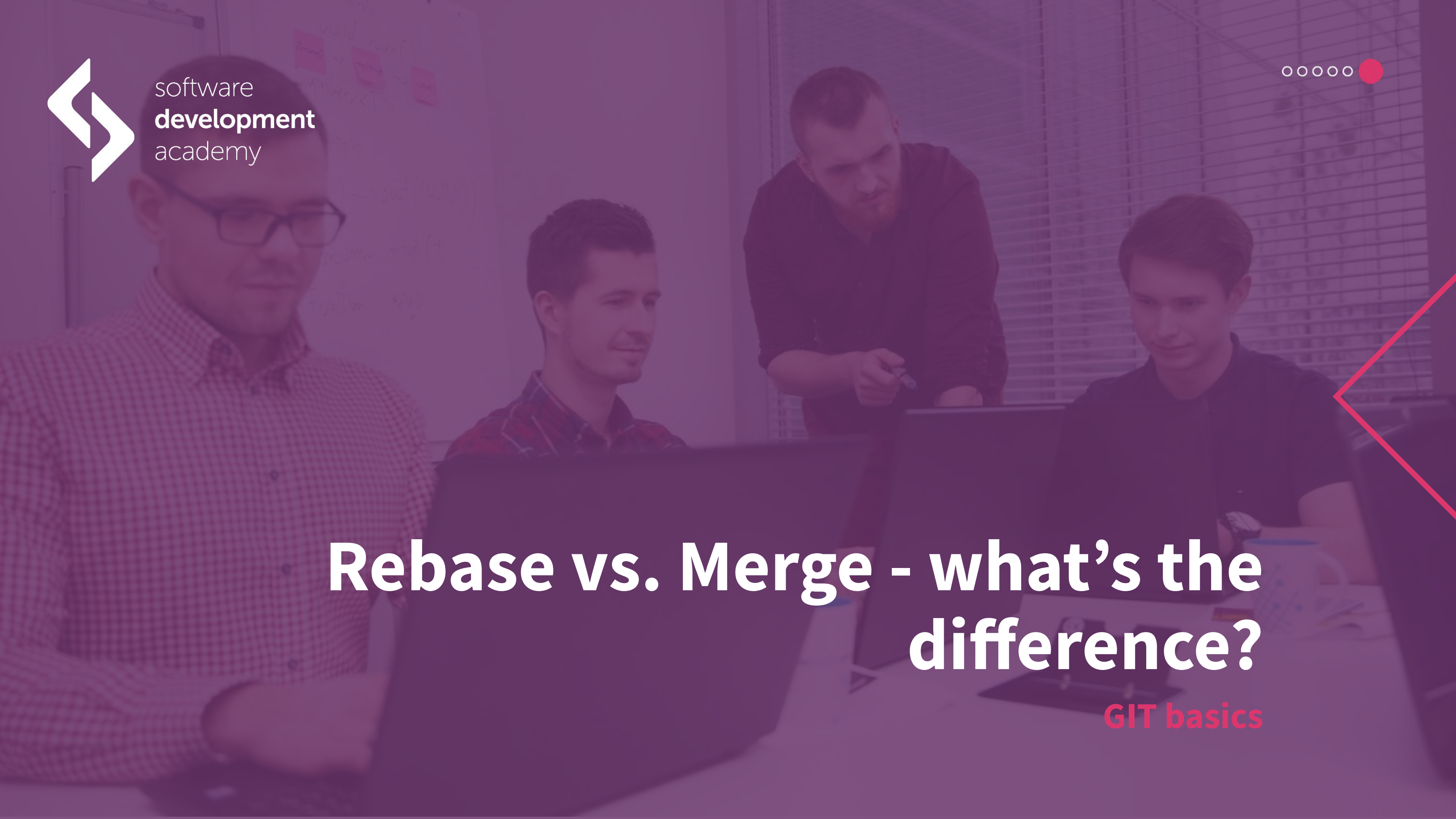
After watching this video:

- You know what fast forwarding is.

software **development** academy

Fast Forwarding- demo

GIT basics



software **development** academy

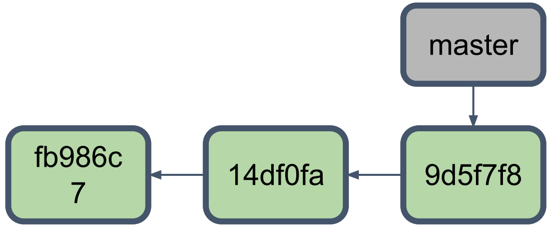
Rebase vs. Merge - what’s the difference?

GIT basics

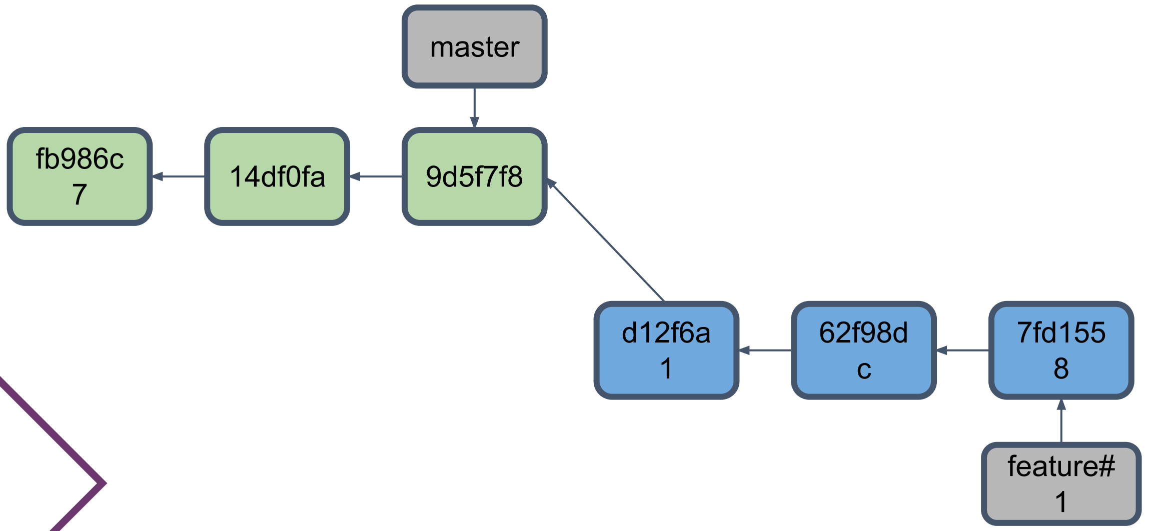
In this video you will learn:

- What is the difference between a regular merge and a rebase?

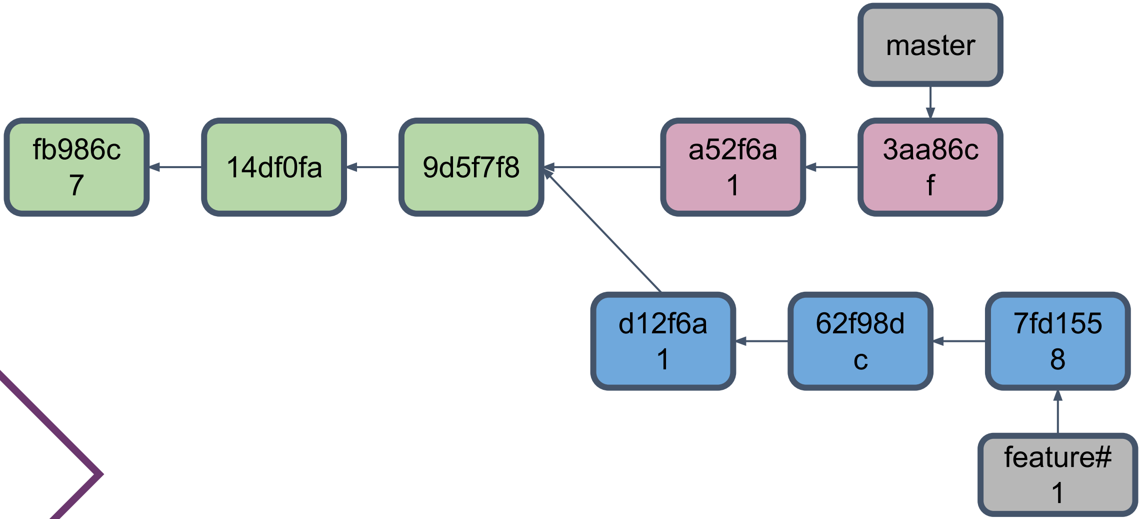
Merge vs. Rebase



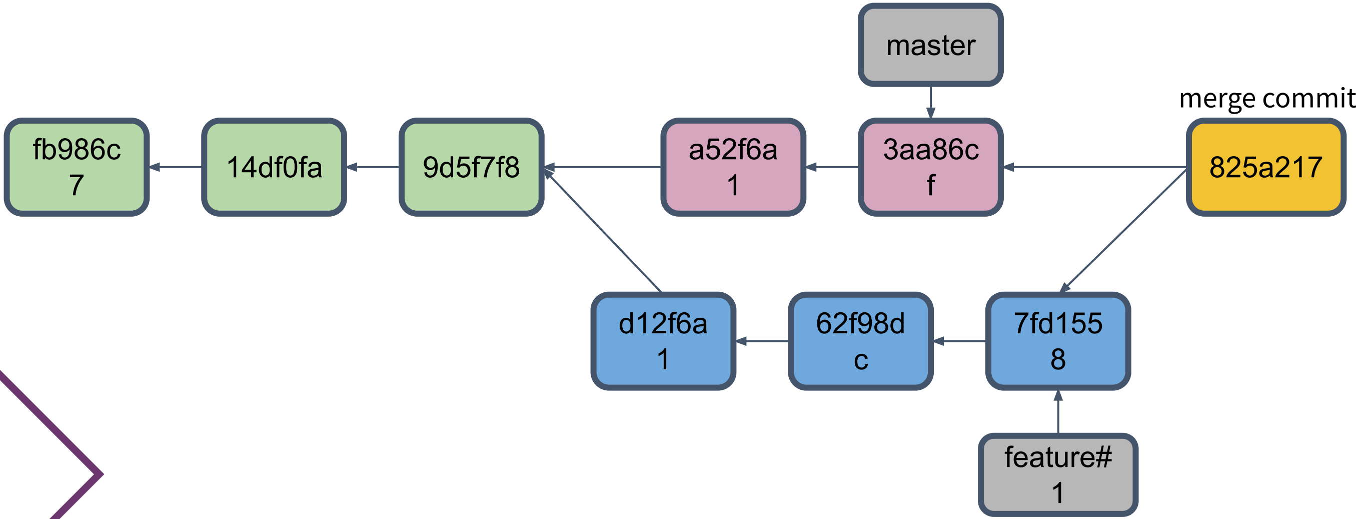
Merge vs. Rebase



Merge vs. Rebase



Merge


master
merge commit
fb986c 7
14df0fa
9d5f7f8
a52f6a 1
3aa86c f
825a217
d12f6a 1
62f98d c
7fd155 8
feature# 1

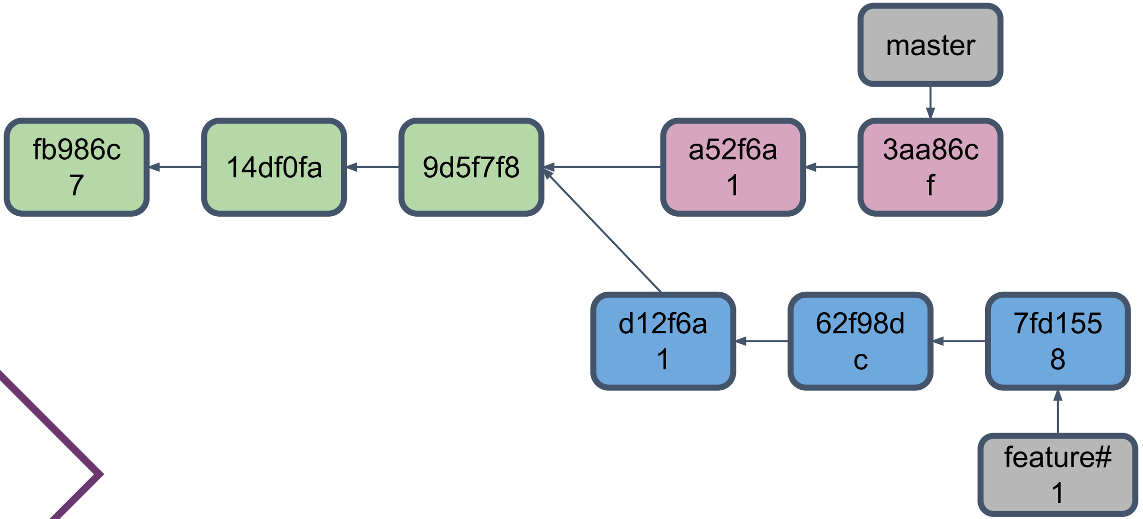
Rebase

Command

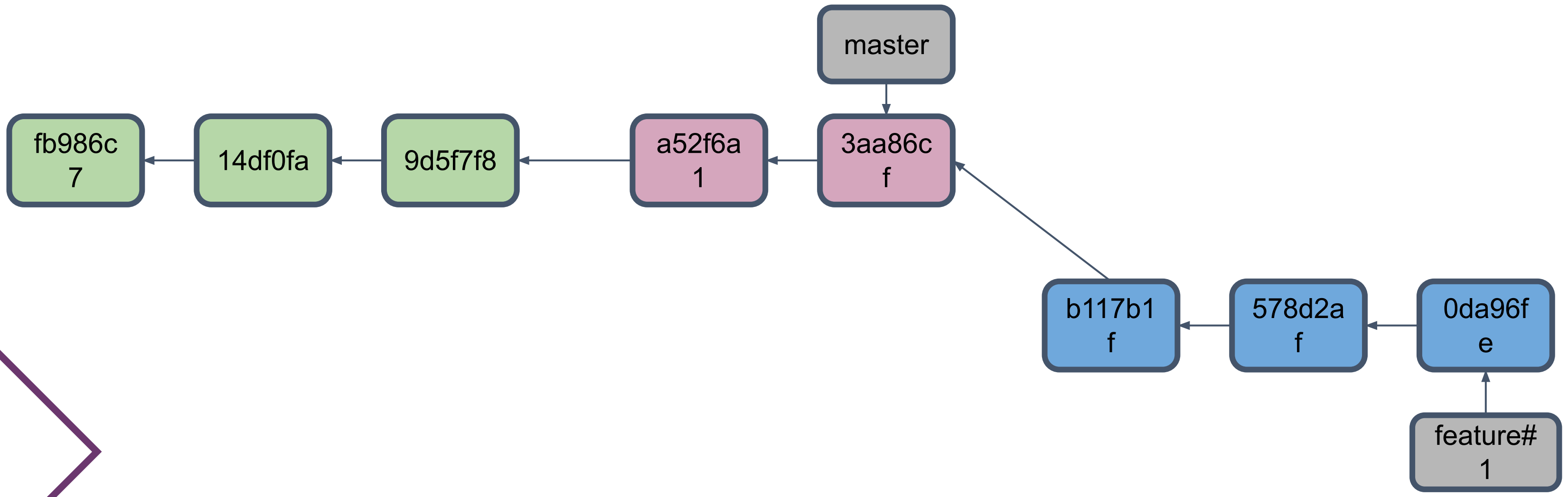
git rebase <branch_name>

allows you to change the base of the branch we are on to the branch given in the command.

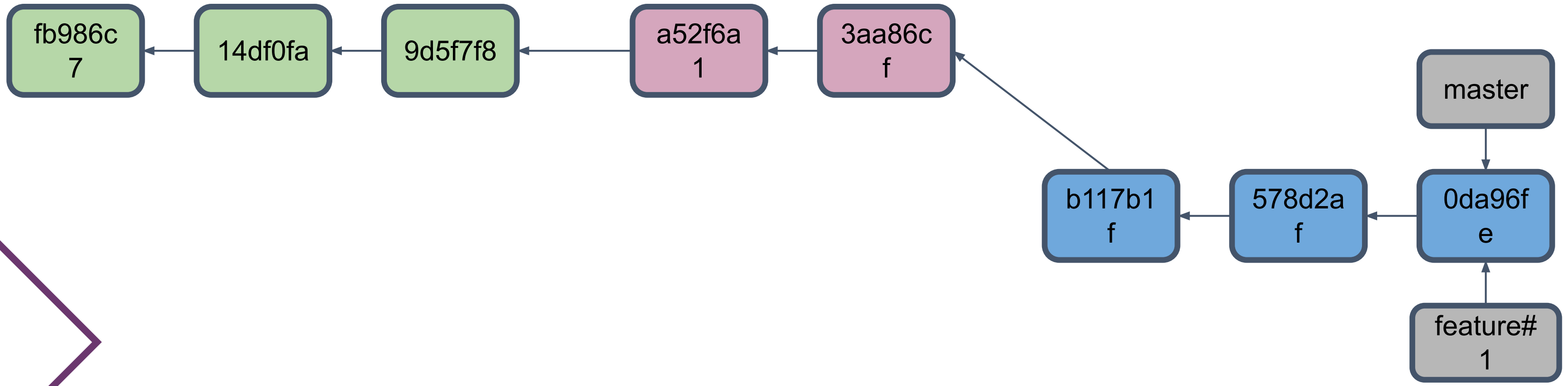
Rebase


master
fb986c 7
14df0fa
9d5f7f8
a52f6a 1
3aa86c f
d12f6a 1
62f98d c
7fd155 8
feature# 1

Rebase


master
fb986c
7
14df0fa
9d5f7f8
a52f6a
1
3aa86c
f
b117b1
f
578d2a
f
0da96f
e
feature#
1

Rebase



fb986c7
14df0fa
9d5f7f8
a52f6a1
3aa86cf
master
b117b1f
578d2af
0da96fe
feature#1

Interactive rebase

Command

git rebase-i <branch_name>

launches the interactive rebase creator. You can change commits in it, including joining them together, changing their messages.

Advantages and disadvantages of rebase

Advantages:

- Simplifies the potentially complex story.
- Manipulating a single commit is easier.
- Prevents merge commits from appearing in the repository.

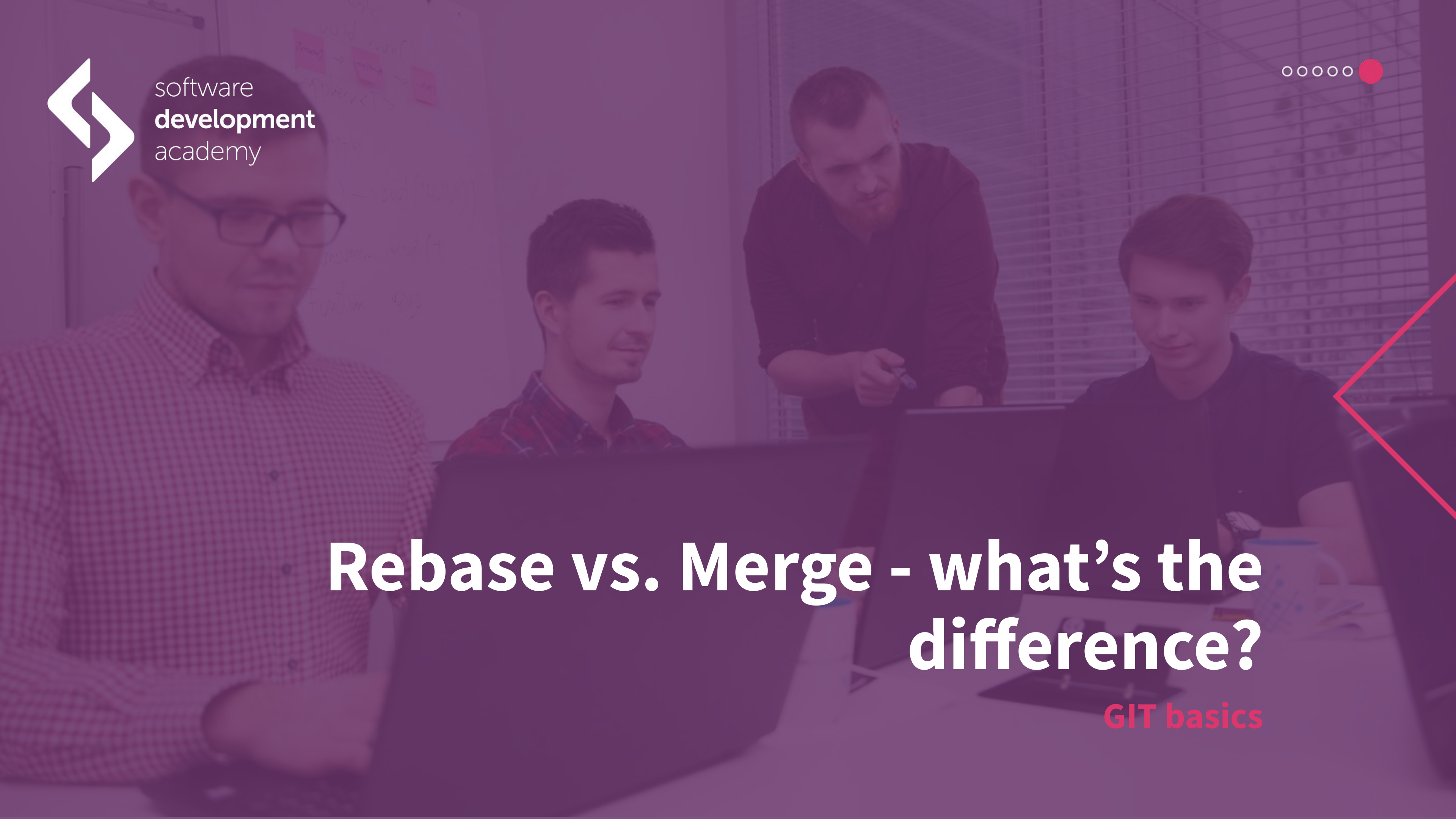
Advantages and disadvantages of rebase

Disadvantages:

- Collecting several commits into one can distort the context of work.
- Performing a rebase in a public repository can be dangerous if you work as a team.
- A little more work - you need to use rebase to keep branch updated.

After watching this video:

- You know the differences between merge and rebase.
- You have learned the pros and cons of rebase.



Rebase vs. Merge - what's the difference?

GIT basics

Stash

GIT basics

In this video you will learn:

- What is the stash and how to use it?

Stash

Command	
git stash	allows you to 'put' introduced changes aside, without having to commit them.
git stash list	lets you see a list of saved changes in the stash.
git stash apply	allows you to re-apply the changes you have recently stashed. This option only integrates changes, they will still be listed in the stash list.
git stash pop	applies recently stashed changes and then removes them from the stash list.

After watching this video:

- You know what a stash is.
- You learned about the stash commands.

software
development
academy

Stash - demo

GIT basics