

Mentoring (التوجيه)

Session (1)

04/02/2021

7:30 PM (Cairo Time)



Technical Skills (DevOps)

Arabic (عربي)

Mohamed Radwan





Overview of the program

“

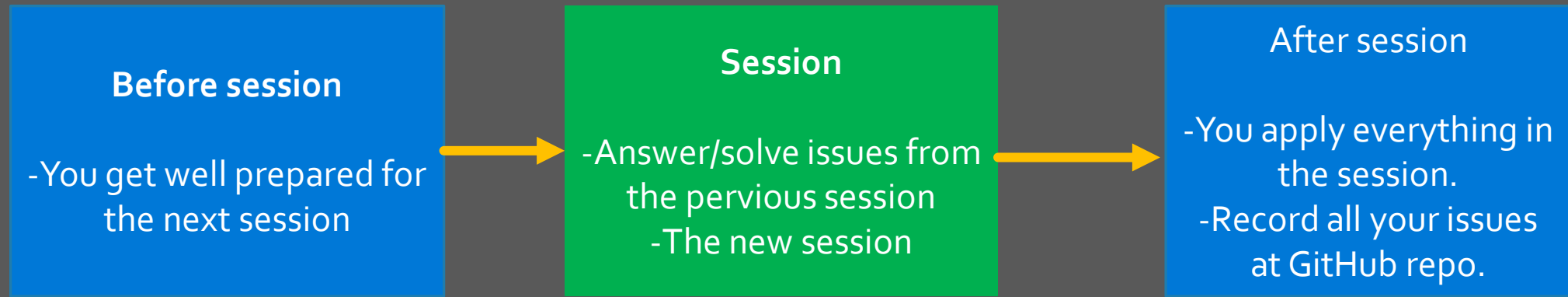
Reading **1000 books** about how to drive a car will **never** make you a driver.

Nothing can beat experience.

”

Mohamed Radwan

Overview of the program



Next session:?

- 2 weeks
- 1 month
- 2 months
- More

Record your issues/questions on GitHub

The screenshot displays the GitHub interface for the repository **MohamedRadwan-DevOps / DevOps-step-by-step-arabic**. The **Issues** tab is selected in the top navigation bar. A red box highlights the repository name (1). In the top right, the **Watch** (2) and **Star** (3) buttons are highlighted. The **Issues** tab in the navigation bar is highlighted with a red box (4). Below the navigation bar, a message states: "Label issues and pull requests for new contributors. Now, GitHub will help potential first-time contributors discover issues labeled with **good first issue**." Below this, the **Filters** dropdown is set to "is:issue is:open" (5). To the right of the filters, the **Labels** (9) and **Milestones** (0) buttons are visible. The **New issue** button is highlighted with a red box (6). The main content area shows a "Welcome to issues!" message, explaining that issues are used to track todos, bugs, feature requests, and more, and that they will appear in a searchable and filterable list. To get started, users are encouraged to [create an issue](#).

MohamedRadwan-DevOps / DevOps-step-by-step-arabic

Watch 0 Star 0 Fork 0

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Label issues and pull requests for new contributors
Now, GitHub will help potential first-time contributors discover issues labeled with **good first issue**

Dismiss

Filters is:issue is:open Labels 9 Milestones 0 New issue

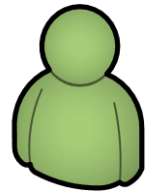
Welcome to issues!

Issues are used to track todos, bugs, feature requests, and more. As issues are created, they'll appear here in a searchable and filterable list. To get started, you should [create an issue](#).

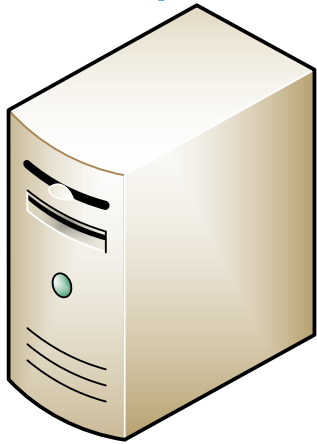
GitHub



Why Version Control ?



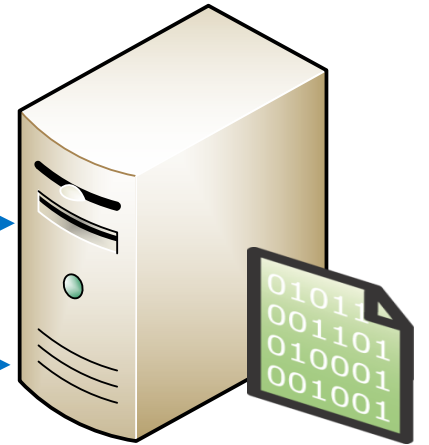
Seif



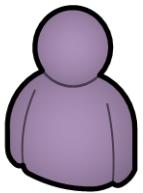
```
Function1 ()  
{  
  //code  
}
```



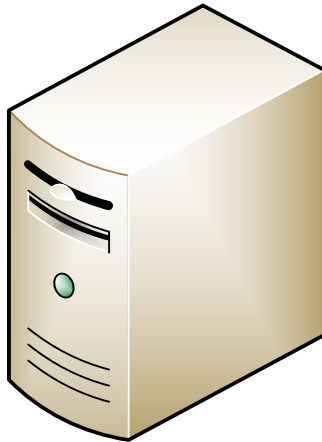
Add



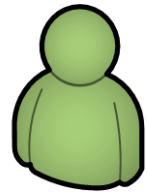
Version Control



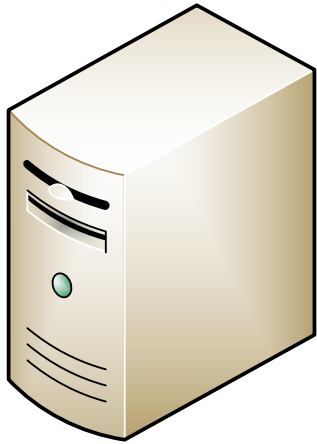
Lara



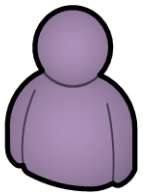
```
Function2 ()  
{  
  //code  
}
```



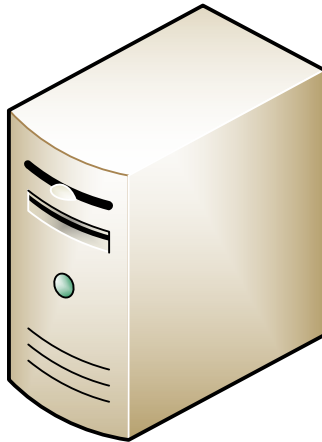
Seif



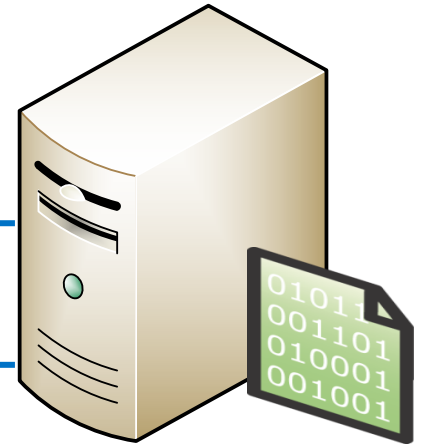
```
Function1 ()  
{  
  //code  
}  
Function2 ()  
{  
  //code  
}
```



Lara



```
Function1 ()  
{  
  //code  
}  
Function2 ()  
{  
  //code  
}
```



Version Control

Why Version Control?

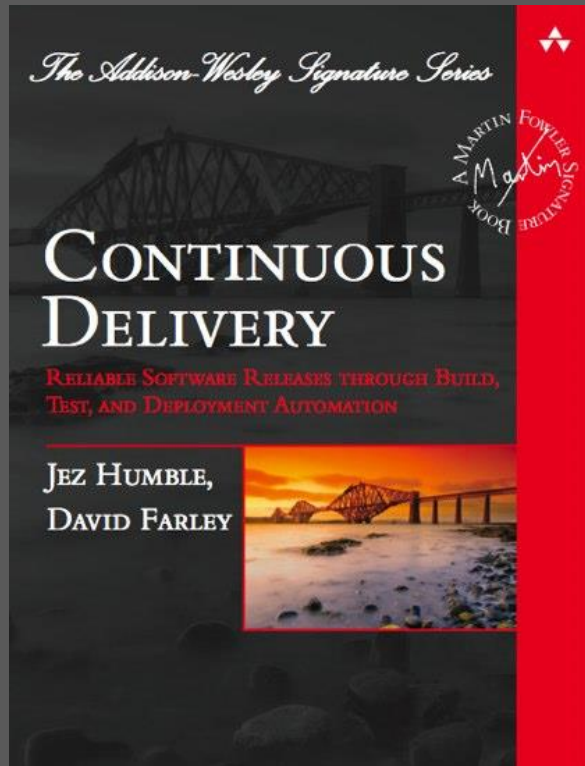
- Teamwork (collaboration)
- Versioning and history
- Responsibility (blame)
- Revert to any version
- Full audit





Is Version Control only for
developers?

Principles and Practices of Continuous Delivery



- 8 Principles of Continuous Delivery
- 4 Practices of Continuous Delivery

8 Principles of Continuous Delivery

1. .

2. .

3. .

4. Keep everything in source control

5. .

6. .

7. .

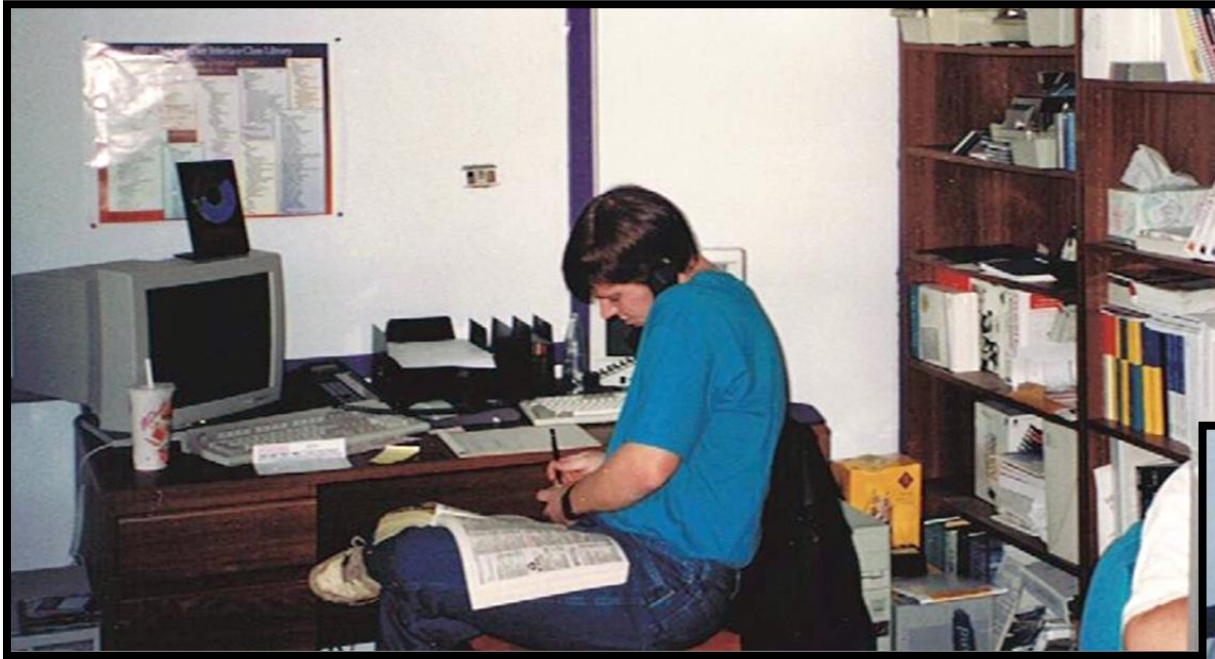
8. .



History of Version Control

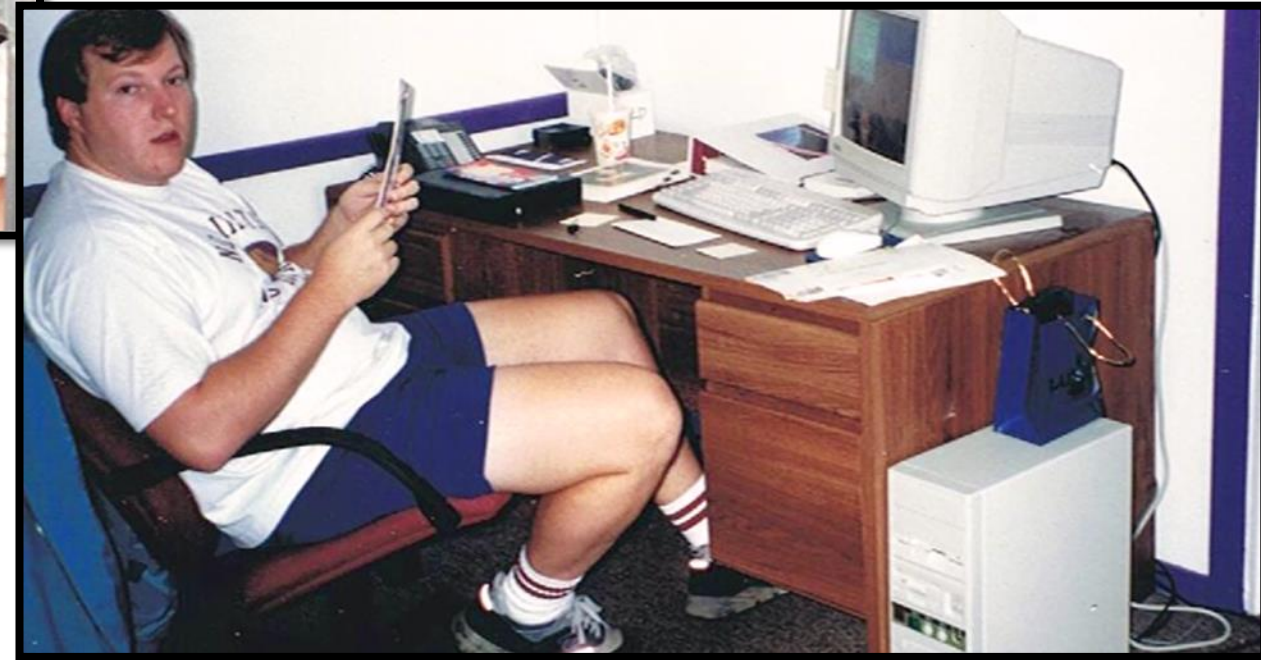


Brian Harry
Creator of Visual SourceSafe
and Team Foundation Server




90's

A small start-up (One Tree Software)



Visual SourceSafe (VSS)

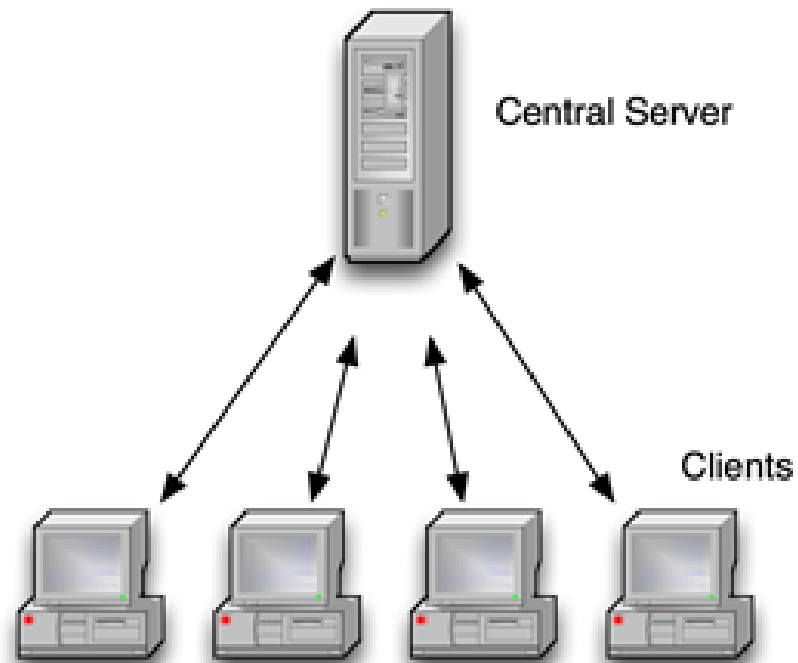


In 1994, Microsoft acquired
One Tree Software,
a small start-up company
which produced SourceSafe

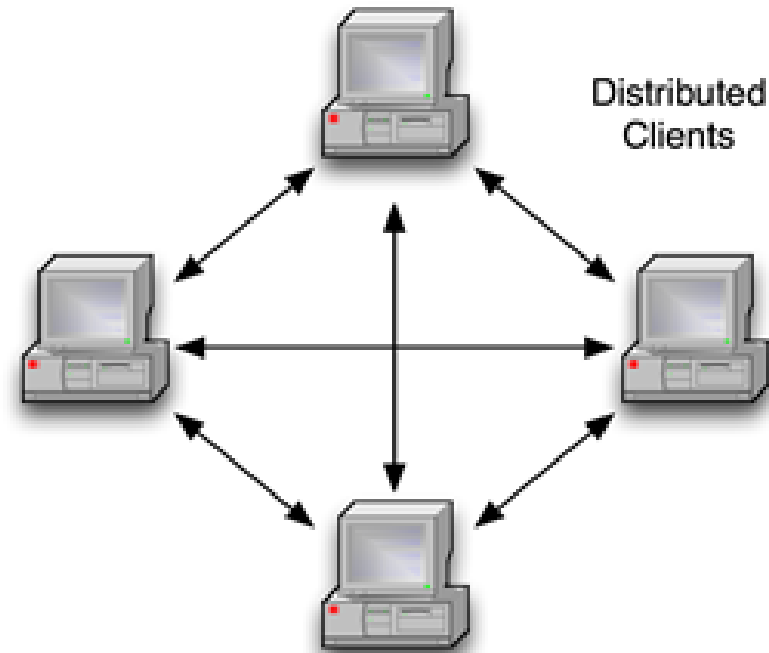
History of Version Control

- VSS (1994)
- SVN (2000)
- TFS->TFVC (2005)
- Git (2005)
- TFS-> Git (2013-2014)

Version control architecture



Other SC/VC



Git

In 2018 Brian Harry decided to take a break for 1 year



After more than 23 years at Microsoft, I've decided it's time for me to take a break. Starting March 12th 2018, I'll be taking a leave of absence for a year. Deciding to do this has been one of the most gut-wrenching decisions of my life. As someone who.....

TFS/VSTS -> Rebranded as Azure DevOps (2018)

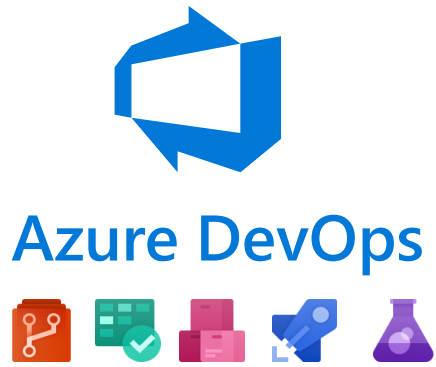
In 2018 Microsoft acquired GitHub for \$7.5 billion





Git is not GitHub!

Example of Git Services



Azure DevOps



Azure Boards

Deliver value to your users faster using proven agile tools to plan, track, and discuss work across your teams.



Azure Test Plans

Test and ship with confidence using manual and exploratory testing tools.



Azure Pipelines

Build, test, and deploy with CI/CD that works with any language, platform, and cloud. Connect to GitHub or any other Git provider and deploy continuously.



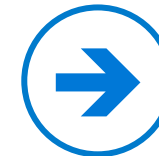
Azure Artifacts

Create, host, and share packages with your team, and add artifacts to your CI/CD pipelines with a single click.



Azure Repos

Get unlimited, cloud-hosted private Git repos and collaborate to build better code with pull requests and advanced file management.



<https://azure.com/devops>





Is Version Control only for
developers?



“ Version Control is a **must** for
everyone! ”

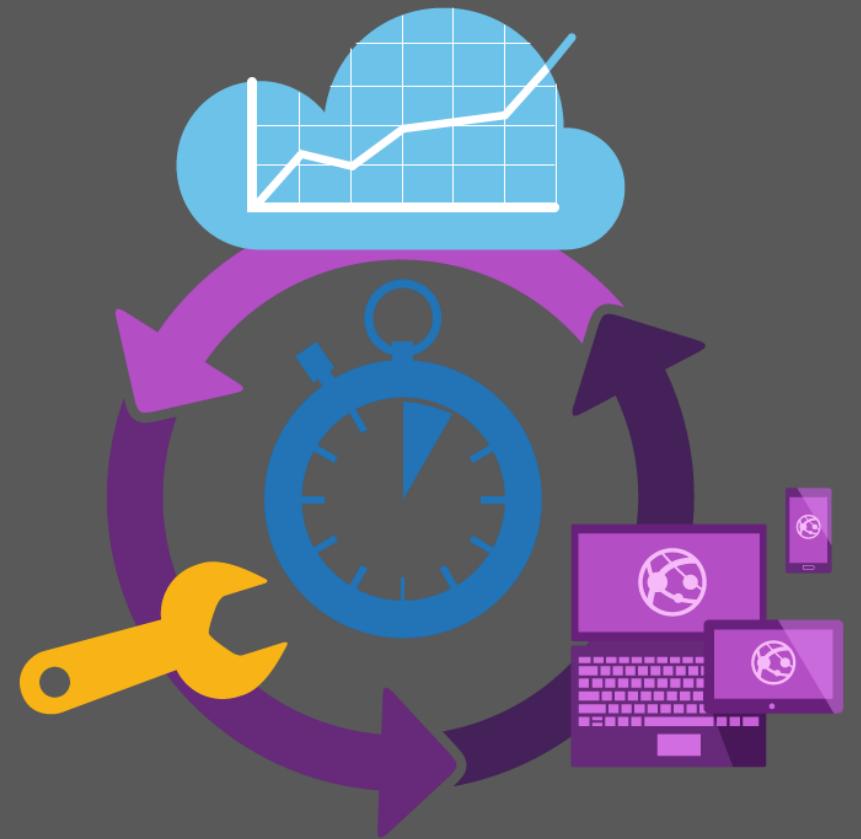
Mohamed Radwan

Keep everything in source control

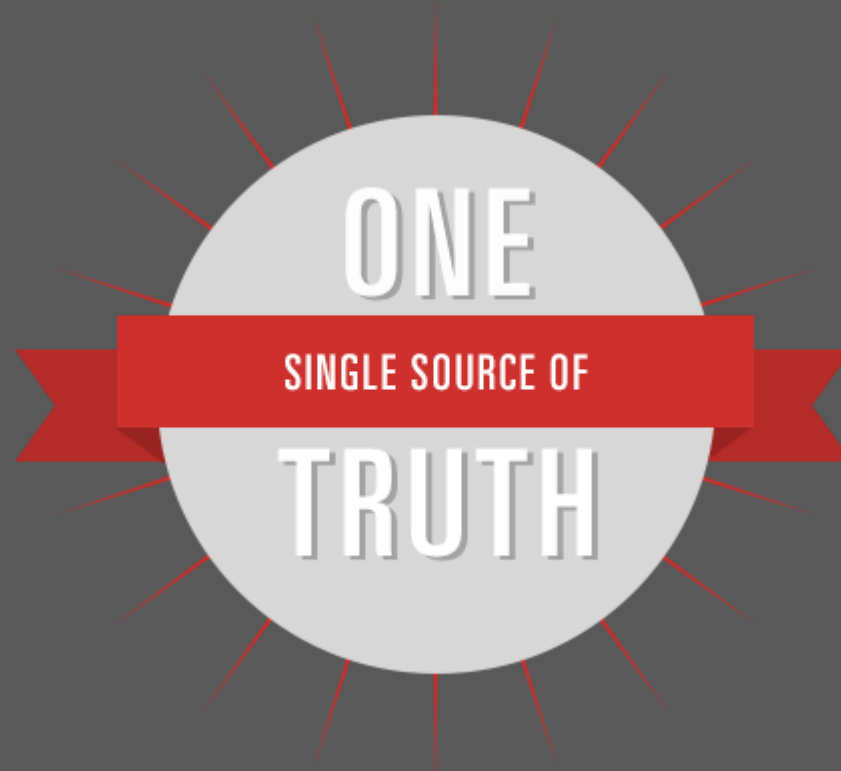
- Application Code —————> Developers
- Testing Code —————> Testers/QA
- Infrastructure as Code —————> IT Admins
- Configuration as Code —————> IT Operation/SRE
- Docs as Code —————> Everyone

Demo

Docs as Code



Single Source of Truth (SSOT)



In information systems design and theory, single source of truth is the practice of structuring information models and associated data schema such that every data element is mastered in only one place.

GitOps



gitops

Demo

Install git Ubuntu / Windows



Types of files for SC

- Binaries ✗
- Text ✓
 - Generated text files ✗
 - Manual text files ✓

This topic is huge and has many details!

Demo

Manual text files Vs. Generated text files



Changing (transformation) the form of files



This topic is huge and has many details!

Ignore files (.gitignore)

*.xxx

.vs/

Demo

Reduce mistakes for undesired files for SC



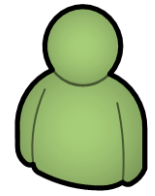
Types of changes

- Add a new file
- Delete an existing file
- Edit an existing file
- Rename (delete/add)

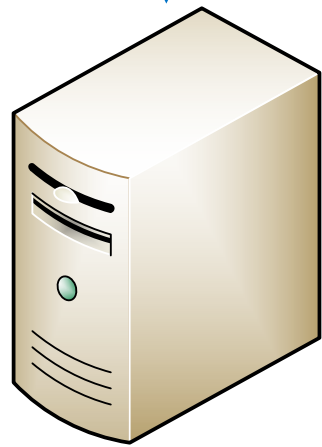
Edit an existing file changes

- Edit existing lines by one user
- Edit existing lines by more than one users (conflict)

Edit existing lines by one users



Seif



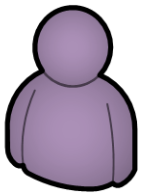
```
Function1 ()  
{  
  //code change 1  
}  
Function2 ()  
{  
  //code  
}
```



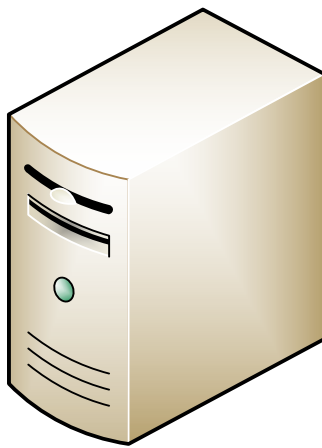
Merge



Version Control

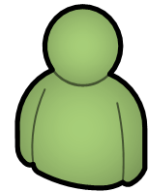


Lara

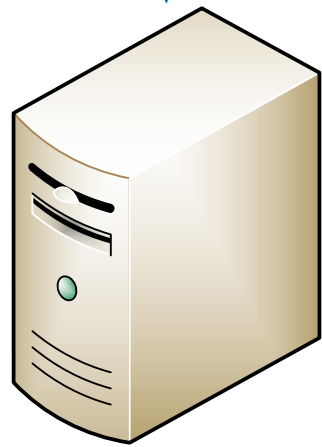


```
Function1 ()  
{  
  //code  
}  
Function2 ()  
{  
  //code change 2  
}
```

Edit existing lines by more than one users



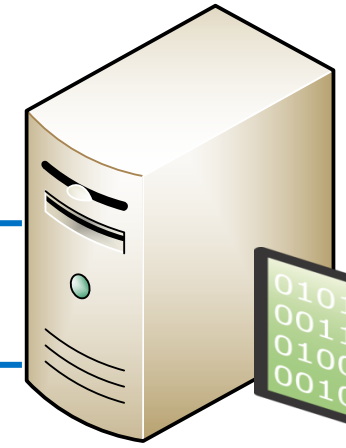
Seif



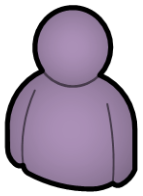
```
Function1 ()  
{  
  //code  
}  
Function2 ()  
{  
  //code change 1  
}
```



Conflict



Version Control



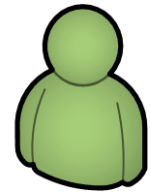
Lara



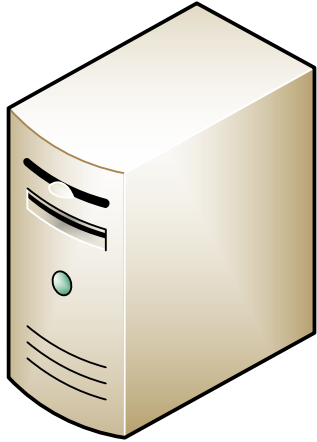
```
Function1 ()  
{  
  //code  
}  
Function2 ()  
{  
  //code change 2  
}
```

Conflict can occur in two places

- On your side ✓
- On target side (branch/remote) ✗

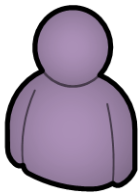


Seif

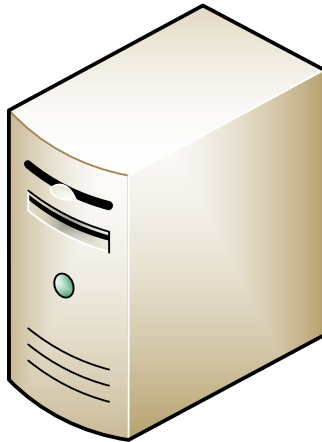


Conflict

```
Function1 ()  
{  
  //code  
}  
Function2 ()  
{  
  //code  
}
```



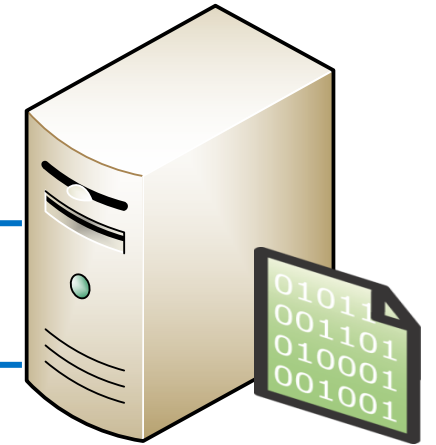
Lara



```
Function1 ()  
{  
  //code  
}  
Function2 ()  
{  
  //code  
}
```

Conflict place

Conflict



Version Control

Demo

Edit an existing file changes (Win Merge)



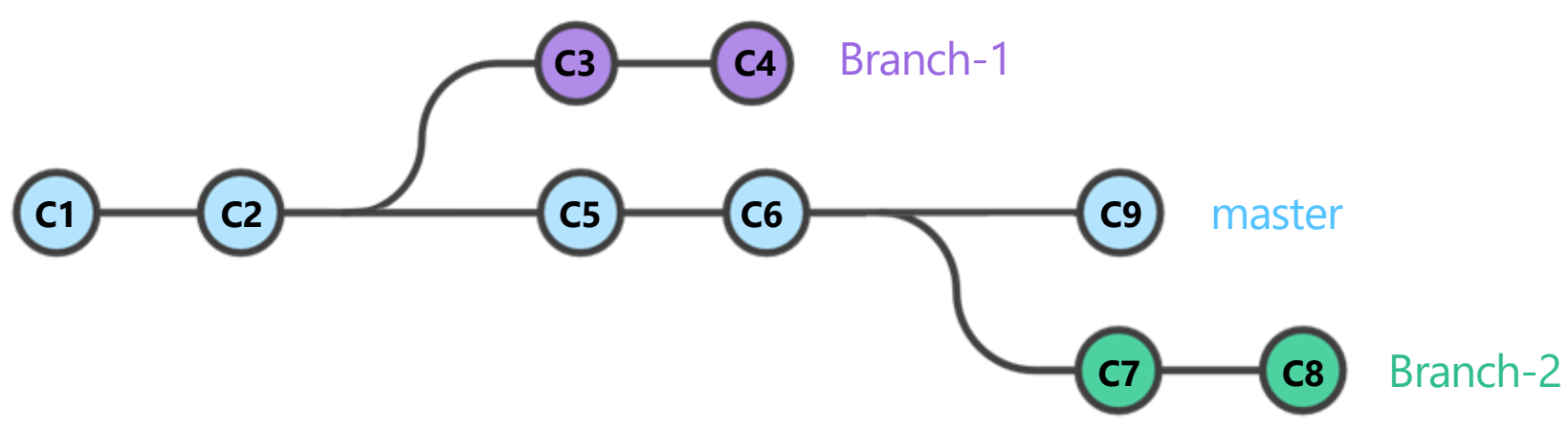
Commit



The commit is a group of changes.

It is the delta that has only the new changes, this is why each commit points to parent commit(s), or which commit(s) are based on.

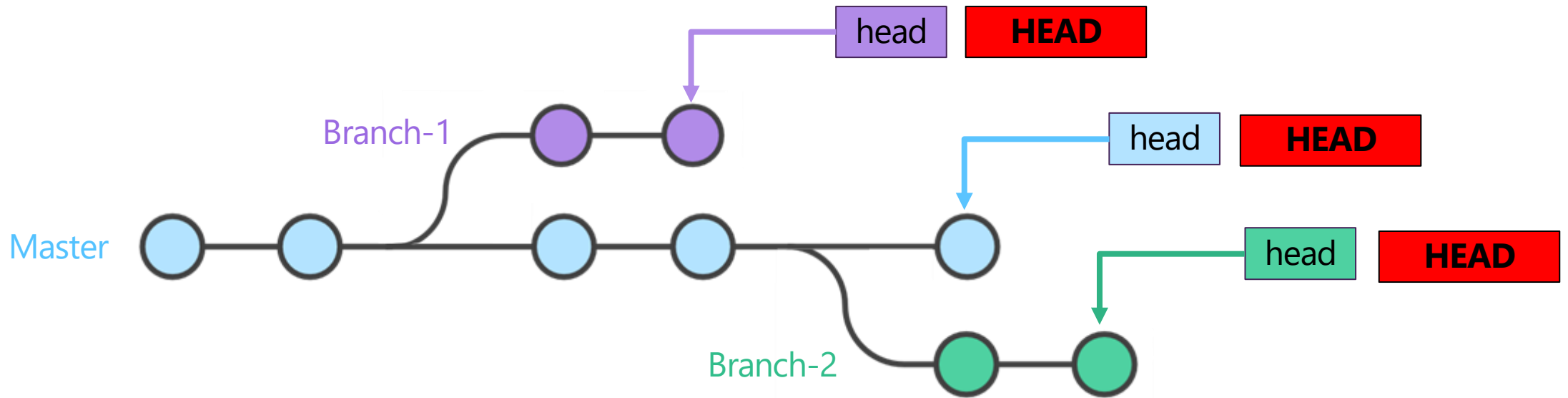
Branch



Branch is the duplication of all commits as an object under version control. Each object (branch) can thereafter be modified separately and in parallel so that the objects become different.

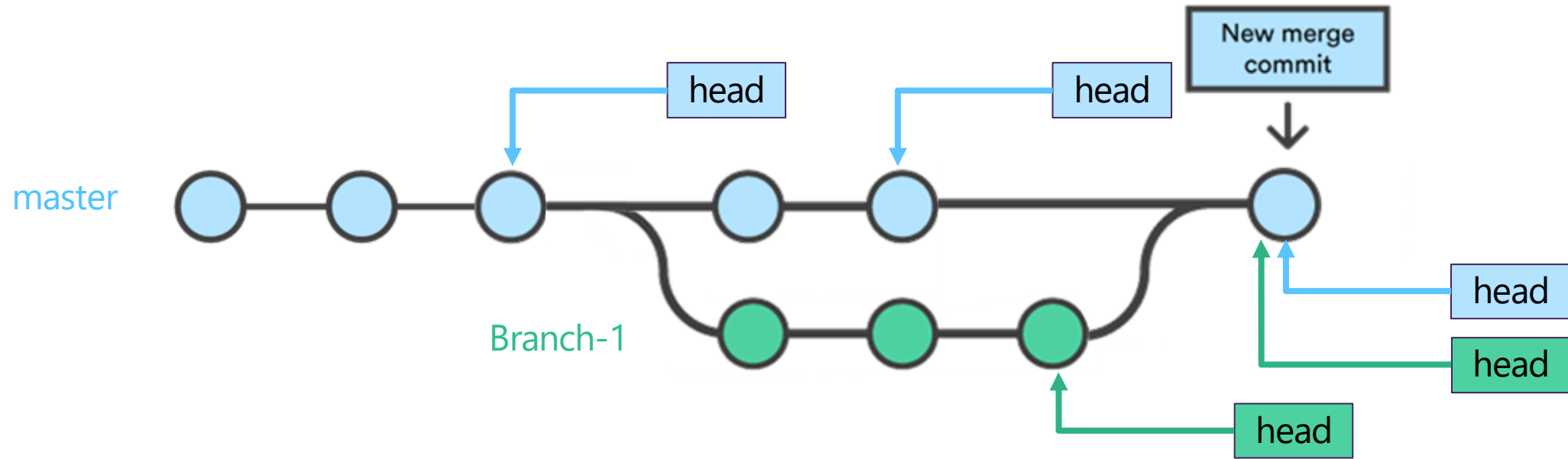
Branch isolates new commit(s) while keep pointing to parent commit(s) before the isolation

Git head and HEAD



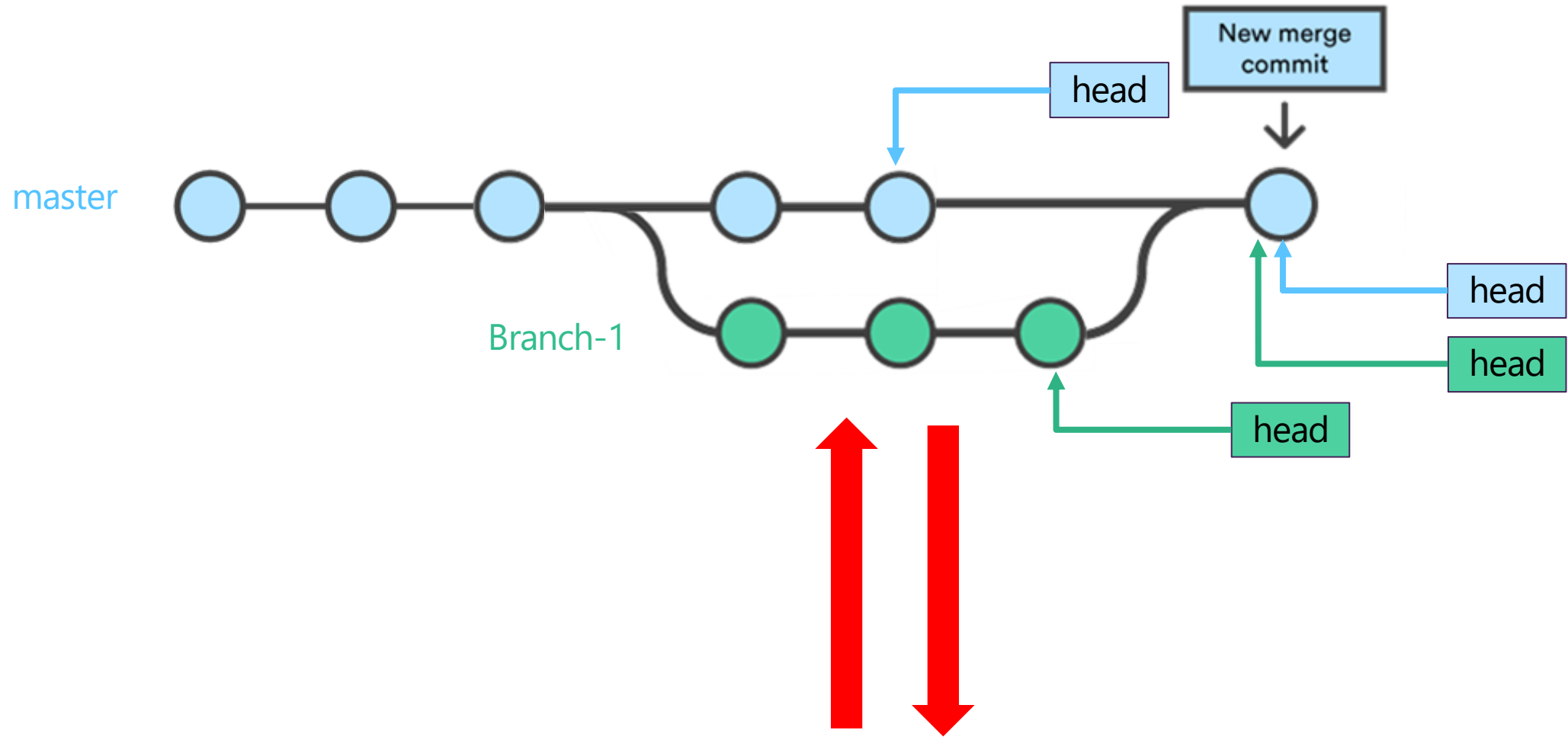
- A head is a reference to a commit object.
- Each head has a name (branch name or tag name, etc).
- By default, there is a head in every repository called master.
- At any given time, one head is selected as the "current head." This head is aliased to HEAD, always in capitals".
- Note this difference: a "head" (lowercase) refers to any one of the named heads in the repository; "HEAD" (uppercase) refers exclusively to the currently active head.

Merge

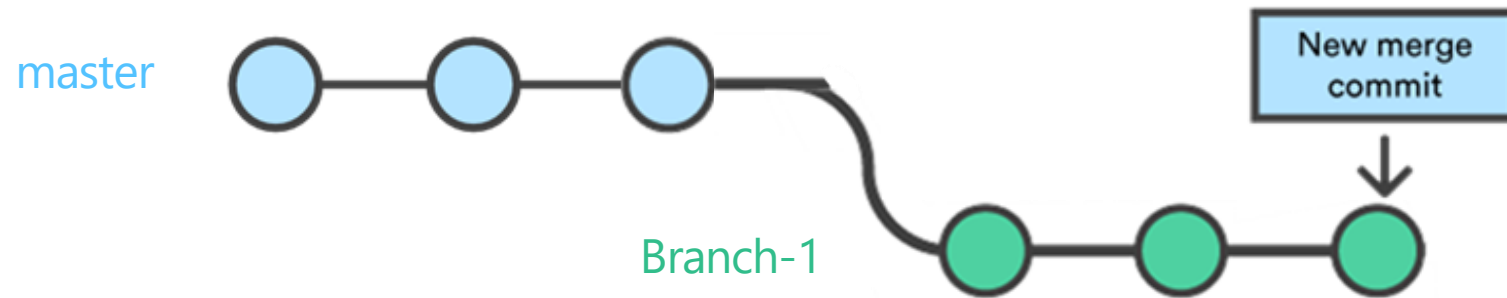


Merge (no FF) is a new special commit that has 2 parents commits

Merge (direction)



Merge (Fast-Forward) & Rebase

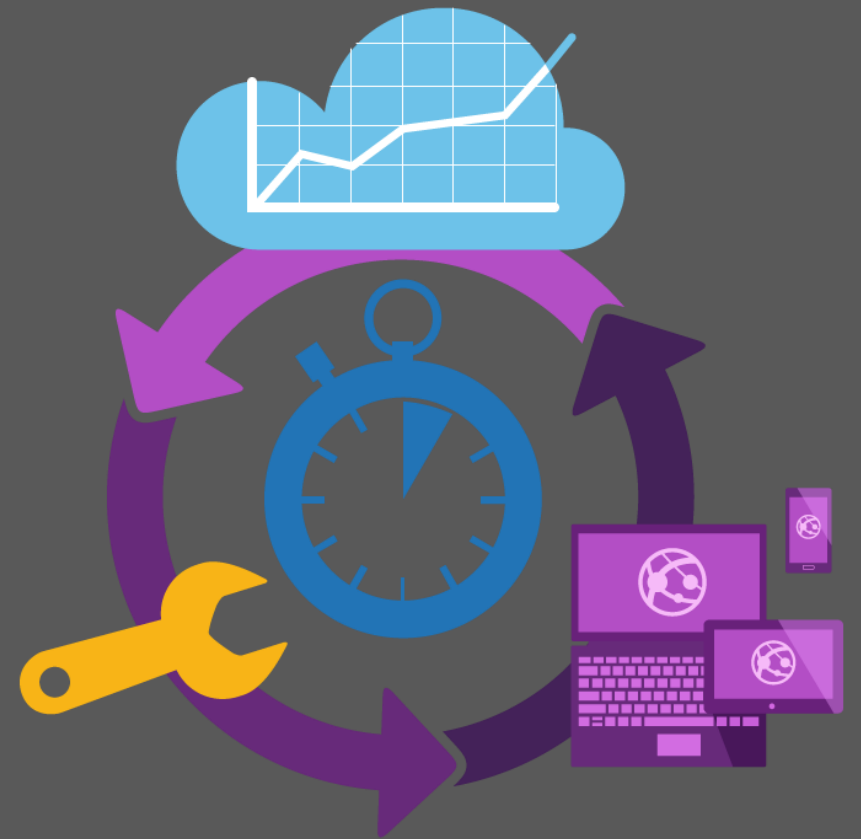


Merge (FF) is just moving the HEAD to the last commit

Rebase is reapplying the new commits from the commit that starts the isolation (linear commits instead of multiple parents)

Demo

Git visualization



Demo

Git visualization merge on both sides





Types of merge

Types of merge



Squash

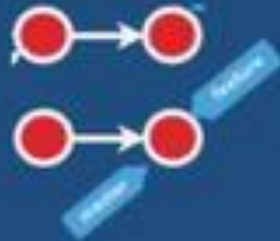


Merge (No FF)

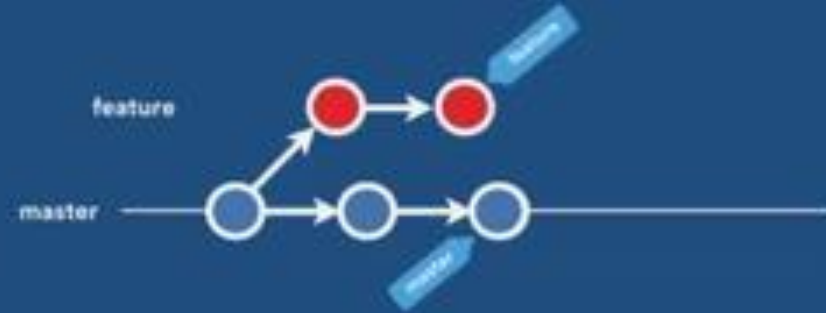


Rebase

Types of merge



Fast-Forward Merge



Merge

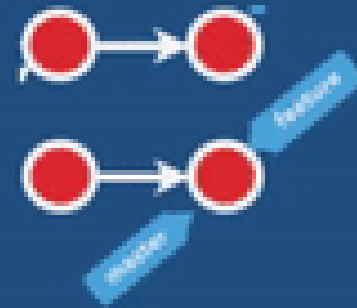


Squash

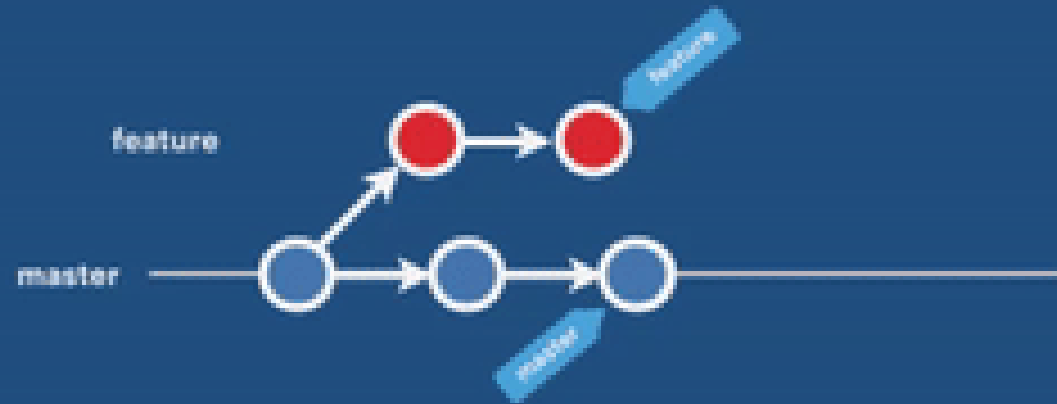


Rebase

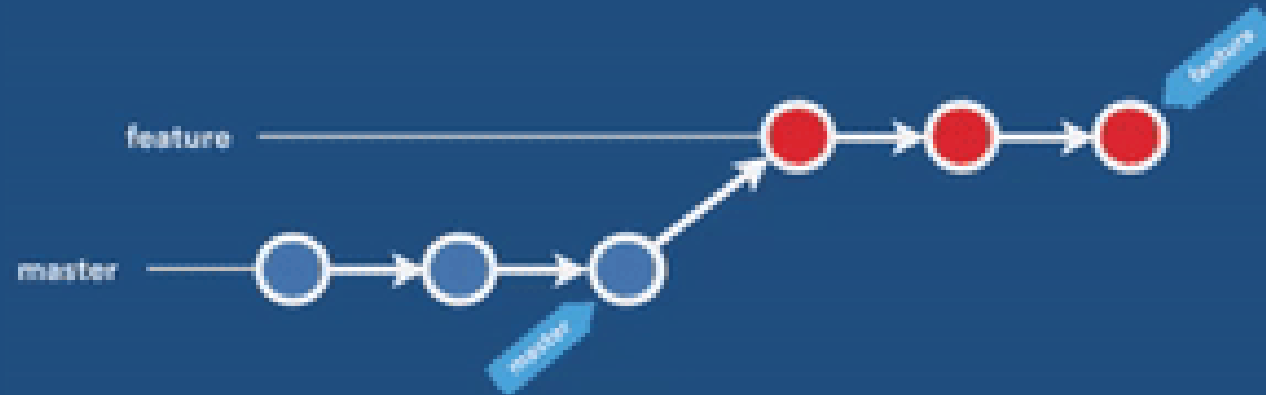
Fast-Forward Merge



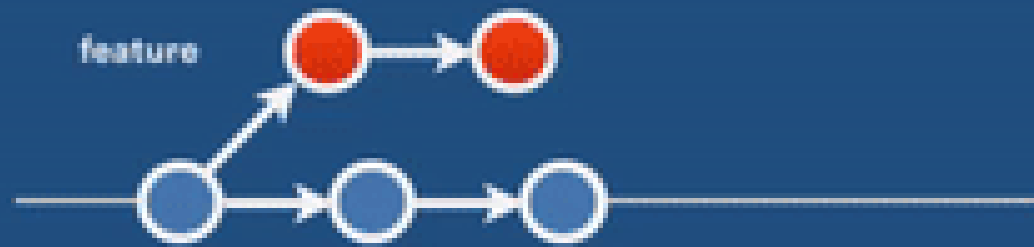
Merge (No FF)



Squash



Rebase



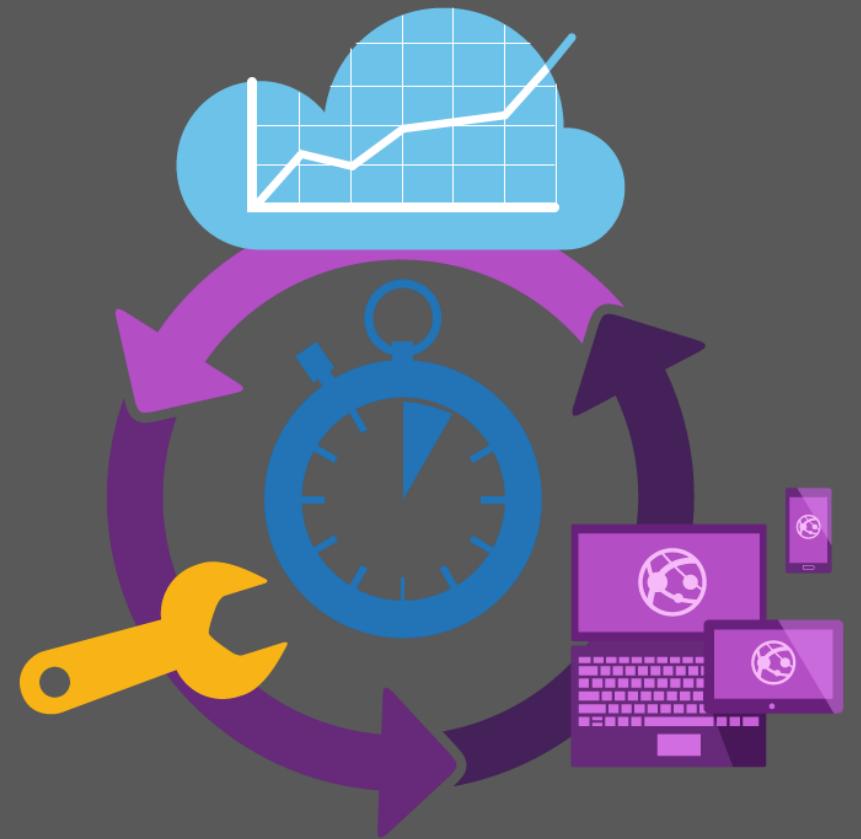
Demo

Git visualization for FF and no FF



Demo

Git rebase



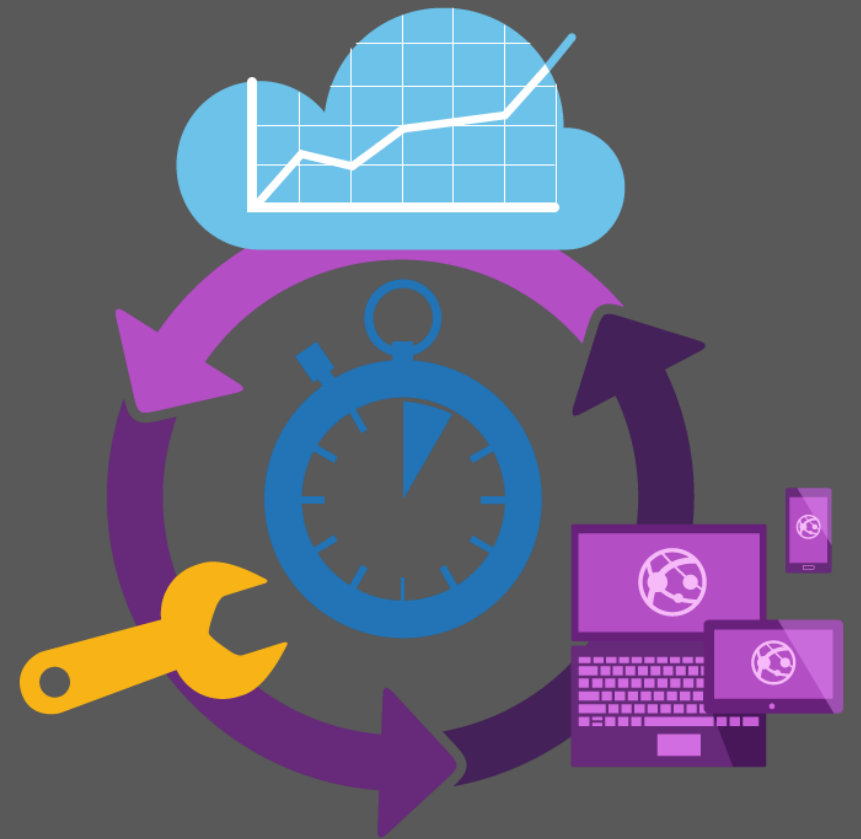
Demo

Git cherry-pick



Demo

Git HEAD



Demo

Git HEAD + cherry-pick



Demo

Git commands



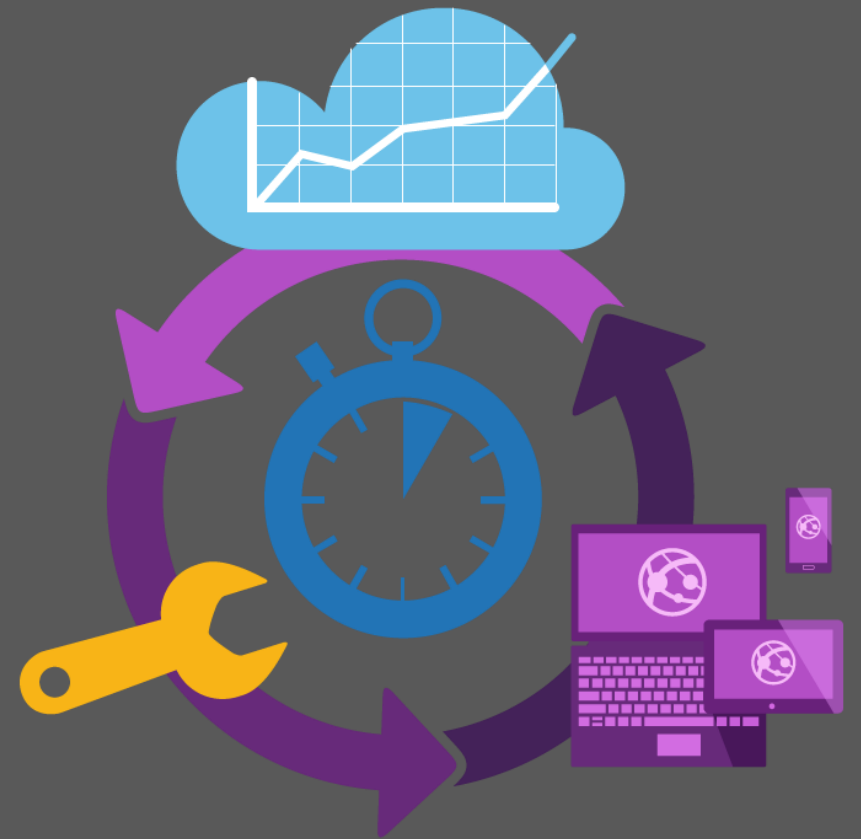
Demo

Gitignore



Demo

Git UI



Demo

Git remote (add/clone)



Demo

GitHub



Demo

Git conflict



“

Reading **1000 books** about how to drive a car will **never** make you a driver.

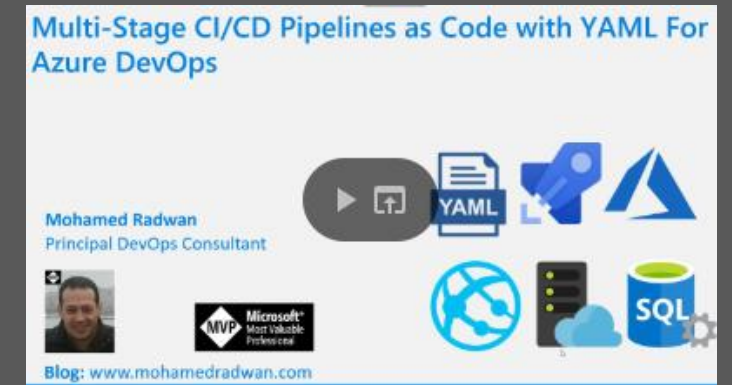
Nothing can beat experience.

”

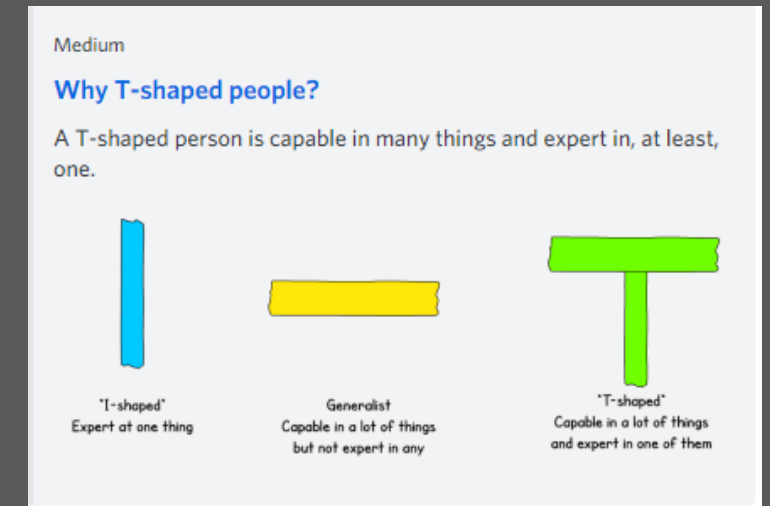
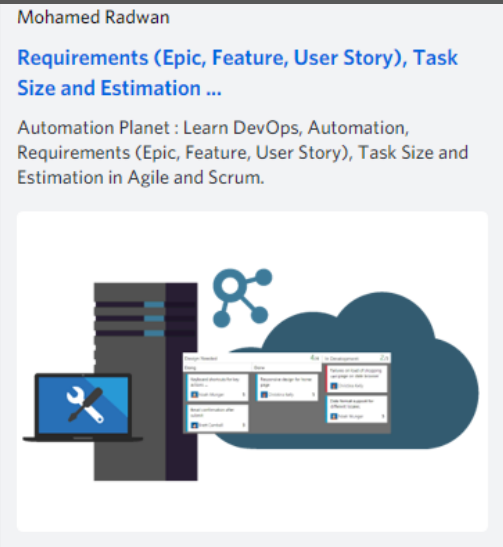
Mohamed Radwan

Next session material

Videos



Posts



Next session checklist

- Apply and practice everything you learned in this session. ✓
- Any questions/problems, report them on GitHub as issues. ✓
- Well prepare/study the material of the next session. ✓
- Well prepare your questions/problems of the next session. ✓

لو استفادتوا من المحاضرة



Like



Share



Comment



Subscribe



Activate
Notification

Mentoring (التوجيه)

Mohamed Radwan



Soft Skills
Arabic (عربي)

Amateurs Anonymous

جمعية الهواة المجهولين

مرحبا بك في جمعية الهواة المجهولين
(AmateursAnonymous.org)

الأسئلة الشائعة	معلومات أخرى	أمثلة على المبادئ	الاجتماعات	نبذة عن الجمعية	مبادئ الجمعية
					



Thank you!

شكرا