

Live Demo

1. Adding, Staging & Committing

- create `poems.md` :

My Poems Collection

=====

Eduard Mörike: Er ist's

Frühling lässt sein blaues Band
Wieder flattern durch die Lüfte;
Süße, wohlbekannte Düfte
Streifen ahnungsvoll das Land.
Veilchen träumen schon,
Wollen balde kommen.
- Horch, von fern ein leiser Harfenton!
Frühling, ja du bist's!
Dich hab ich vernommen!

- stage and commit `poems.md`
- log
- create `main.c` :

```
int add(int a, int b){  
    return a + b;  
}
```

```
int getAnswerToTheUltimateQuestionOfLifeTheUniverseAndEverything(void){  
    return 42;  
}
```

- create `README.md` :

Git Basics

=====

Repository Contents

My Repository contains

- Poems in `poems.md`
 - Mörike
- Code in `code.c`
 - Addition
 - The Answer

I like both,
but the best is of course: <to be defined>

- status
- diff
- add code.c
- status
- diff
- add .
- status
- diff
- diff --staged
- git restore --staged
- git status
- commit
- log
- status
- add
- commit

- log
- status
- log --stat
- log --patch

2. Branching

- branch
- create branch add_code
- branch
- create branch add_poem
- branch
- switch add_poem
- log
- add Silesius to authors in README.md
- add and commit poem

```
Angelus Silesius: Die Ros' ist ohn warum ...  
-----
```

```
Die Ros' ist ohn warum, sie blühet, weil sie blühet,  
Sie acht nicht ihrer selbst, fragt nicht, ob man sie siehet.
```

- switch add_code
- add multiplication to code in README.md
- create and commit code

```
int mult(int a, int b){  
    return a * b;  
}
```

- log
- switch code/poem/master

3. Merging

- switch master
- ff-merge poems
- log
- normal merge code
- log
- show merged files
- show that `README.md` merged correctly with changes from both branches
- best is: code vs poems on both branches
- merge poems
- merge with conflict code
- resolve merge conflict

4. Remotes

- IN CENTRAL repo: `git config receive.denyCurrentBranch warn`
- Clone repo alice
- Clone repo bob
- ALICE: create `more_poem`
- create

A Pangram

The quick brown fox jumps over the lazy dog.

- clone repo bob
- `git remote -v`
- BOB: create branch more code
- create

```
int sub(int a, int b){  
    return a - b;  
}
```

- commit and push -> fetch first, merge -> push