

# m1ke - local vcs

`m1ke` - это примитивная система контроля версий, аналог `git` который умеет работать только локально

`m1ke` НЕ запускается через JAR файл. Он работает по принципу `JAVA_HOME` доступен с любой папки как `javac`

Для того чтобы запустить `m1ke` нужно выполнить команду

```
m1ke init
```

К примеру у нас есть папка `C:\project`. Для того чтобы просканировать папку на файлы и уже сделанные изменения нужно выполнить:

```
m1ke integrate %FOLDER_NAME%
```

После этой команды `m1ke` открывает конкретную юзер ветку (branch) которая там сохранилась (если она там не одна, то открывается та ветка, в которой произошли последние изменения). Если в этой папке ничего никогда со стороны `m1ke` не запускалось то запускается первичная ветка `master`. Юзер должен об этом предупрежден сообщением.

```
...
m1ke found there was no branch here
'master' branch choosed

>C:\project
```

Если не знаете что такое ветка (branch) в `VCS` (изучите как ветки работают в `GIT`)

Допустим у нас в этой папке есть текстовый файл, который вы только что создали. К

примеру, он называется 1.txt

```
1.txt
```

```
Hello!
```

Мы хотим закоммитить эти изменения - используем специальную команду

```
...  
>C:\project m1ke save -m "Saving hello message"
```

-m команда сохранить с комментарием

Когда мы комитим, мы сохраняем в специальную скрытую папку **само изменение (дельту изменений)** на ваше усмотрение

А теперь представьте что в этой папке в отдельной **другой** вашей ветке есть такой же файл но с другими изменениями.

Создать ветку (чтобы был не master к примеру):

```
m1ke create-branch someBranchName
```

Выбрать ветку:

```
m1ke get-branch someBranchName
```

Когда вы выбираете ветку - вы автоматически подтягиваете изменения которые вы там же и сохранили с помощью `m1ke save`

Если вы переходите на другую ветку, но вы уже сделали какие то изменения, то **вы эти изменения теряете.**

Удалить ветку:

```
m1ke remove-branch someBranchName
```

Выйти из m1ke:

m1ke quit

