Git添加与提交文件与查看

- 一、Git添加
- 二、Git提交文件
- 三、查看Git仓库的提交历史

一、Git添加

1、在终端或命令提示符中,导航到你的Git项目所在的目录,使用 cd 命令切换目录。

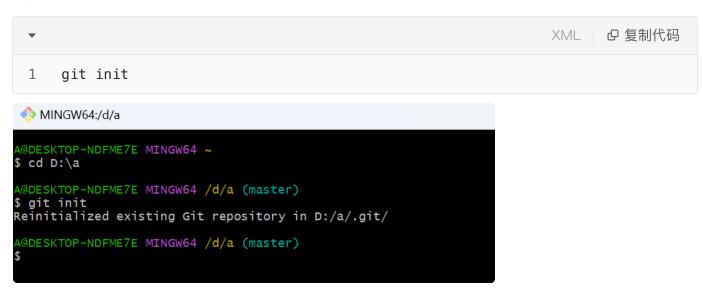
例如: (推荐linux) cd d/目录/目录; 或 (windows) cd D:\目录\目录;

```
MINGW64:/d/a

A@DESKTOP-NDFME7E MINGW64 ~ (master)
$ cd D:\a

A@DESKTOP-NDFME7E MINGW64 /d/a
$
```

2、在目标目录中,运行以下命令来初始化一个新的Git仓库,这将在当前目录下创建一个名为.git的隐藏文件 夹,它是Git仓库的核心部分。



3、使用以下命令将文件添加到Git的暂存区(Staging Area),这样会将所有新创建或修改的文件添加到暂存区,(需要新建一个文件,里面有内容)。

▼ XML □ 复制代码 1 git add 文件名

```
MINGW64:/d/a

A@DESKTOP-NDFME7E MINGW64 ~

$ cd D:\a

A@DESKTOP-NDFME7E MINGW64 /d/a (master)

$ git init
Reinitialized existing Git repository in D:/a/.git/

A@DESKTOP-NDFME7E MINGW64 /d/a (master)

$ git add one

A@DESKTOP-NDFME7E MINGW64 /d/a (master)
```

如果你有多个文件需要添加,可以使用以下命令来一次性添加多个文件:

▼ XML □ 复制代码 1 git add 文件名1 文件名2 ...

如果要添加整个目录中的所有文件,可以使用以下命令:

▼ XML 口复制代码 1 git add.

4、确保你已经添加了所有要提交的文件,可以使用以下命令来查看当前文件的状态,这会显示哪些文件已添加 到暂存区以及哪些文件还未添加。

▼ XML ②复制代码 1 git status

这表示你已经成功地将名为 one/新建文件.txt 的文件添加到Git仓库的暂存区。

如果没有任何未提交的更改,输出将显示 "nothing to commit, working tree clean",当前没有任何未提交的更改,工作区是干净的。

```
MINGW64:/d/a

A@DESKTOP-NDFME7E MINGW64 /d/a (master)

$ git status
On branch master
nothing to commit, working tree clean
```

二、Git提交文件

1、使用以下命令将暂存区的文件提交到Git仓库,将**提交说明**替换为对提交的简要描述,例如修复了某个Bug、添加了新功能等。

```
▼ XML □ 复制代码

1 git commit -m "提交说明"
```

```
A@DESKTOP-NDFME7E MINGW64 /d/a (master)
$ git commit -m "新建了一个文件"
[master (root-commit) 7fa55de] 新建了一个文件
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 "one/\346\226\260\345\273\272 \346\226\207\346\234\254\346\
26\207\346\241\243.txt"

A@DESKTOP-NDFME7E MINGW64 /d/a (master)
$ |
```

三、查看Git仓库的提交历史

1、git log命令用于显示Git仓库的提交历史。在终端中执行以下命令,这将显示最近的提交记录,包括每个提交的作者、提交日期和提交信息等。

XML P 复制代码

1 git log

MINGW64:/d/a

```
A@DESKTOP-NDFME7E MINGW64 /d/a (master)
$ git log
commit 7fa55de79105acda8cdbd8fc5cc983431dc9e06f (HEAD -> master)
Author: jiubana1 <26582737240@qq.com>
Date: Wed Jun 21 23:32:02 2023 +0800
新建了一个文件
```

2、git log命令的一些常用选项:

- (1) git log --oneline:以简洁的单行形式显示提交记录。
- (2) git log -n <数量>: 只显示最近指定数量的提交记录。
- (3) git log --author=<作者>: 只显示指定作者的提交记录。
- (4) git log --grep=<关键词>: 只显示包含指定关键词的提交记录。
- (5) git log --since=<日期>: 只显示指定日期之后的提交记录。
- (6) git log --until=<日期>: 只显示指定日期之前的提交记录。