

## گزارش دستورات مهم GitHub

### 1. Git init:

مقدار دهی اولیه repository

```
git init
```

### 2. Git clone:

آخرین ورژن یک برنامه را کپی می کند.

```
git clone < address>
```

### 3. Git status :

وضعیت فعلی برنامه ها و به روز بودنشان در گیت را بررسی می کند.

```
git status
```

### 4. Git commit:

با این دستور هر کامیت ذخیره می شود و می توان پیشرفت پروژه را زیر نظر داشت.

```
git commit -m "<message text>"
```

### 5. Git Commit log :

تاریخچه کامیت ها را نشان می دهد.

```
git log
```

### 6. Git branch:

با استفاده از برنج ها (شاخه ها) توسعه دهندگان به طور موازی می توانند روی پروژه کار کنند.

دستور ساخت برنج جدید :

```
git branch <branch name>
```

دستور ساخت برنج local :

```
git push -u <remote> <branch name>
```

مشاهده برنج:

```
git banch
```

```
git branch --list
```

پاک کردن:

```
git branch -d < branch name>
```

### 7. Git checkout :

از برنچی که هستیم به برنج مورد نظر می رویم.

```
git branch <current-branch>
```

```
switch to branch <designated-branch>
```

### 8. Git push :

اعمال تغییرات به سرور

```
git push <remote> <branch name>
git push -u origin <branch name>
```

## 9. Git pull:

آپدیت های جدید را میگیرد و وارد ریپازیتوری می کند.

```
git pull <remote>
```

## 10. Git revert :

تغییراتی که داده شده است را باز میگرداند به حالت قبل.

```
git log --online
```

```
git revert (change.Number )
```

```
Shift + q
```

## 11. Git merge :

هنگامی که توسعه سازی برنج کامل شد، آن را با برنج والد مرج می کنیم .

```
git merge
```

### Common git commands

Clone	Clone a repository into a new directory
init	Create an empty Git repository or reinitialize an existing one
add	Add file contents to the index
mv	Move or rename a file, a directory, or a symlink
restore	Restore working tree files
rm	Remove files from the working tree and from the index
bisect	Use binary search to find the commit that introduced a bug
diff	Show changes between commits, commit and working tree, etc
grep	Print lines matching a pattern
log	Show commit logs
show	Show various types of objects
status	Show the working tree status
branch	List, create, or delete branches
commit	Record changes to the repository
merge	Join two or more development histories together

rebase	Reapply commits on top of another base tip
reset	Reset current HEAD to the specified state
switch	Switch branches
tag	Create, list, delete or verify a tag object signed with GPG
fetch	Download objects and refs from another repository
pull	Fetch from and integrate with another repository or a local branch
push	Update remote refs along with associated objects

**Short cuts:**

1. New branch : command +shift+ N
2. Edit file : command + shift + A
3. Push : command + P
4. Open a pull request : command + R