



## تنصيب Git

نظام GitHub يوفر برنامج عميل لسطح المكتب يشمل واجهة مستخدم رسومية لنشاطات المستودعات الأكثر شيوعاً وتحديث نسخة سطر أوامر Git تلقائياً للحالات المتقدمة.

### GitHub لنظام تشغيل وندوز

[windows.github.com](https://windows.github.com)

### GitHub لنظام تشغيل ماك

[mac.github.com](https://mac.github.com)

توزيعات Git لأنظمة لينكس وPosix متوفرة في موقع Git SCM الرسمي.

### Git لكل أنظمة التشغيل

[git-scm.com](https://git-scm.com)

## إعادة تكوين الأدوات

إعداد معلومات المستخدم لجميع المستودعات المحلية

```
$ git config --global user.name "[الاسم]"
```

إعداد الاسم المراد إدراجه في إجراءات تنفيذ أو "ارتكاب" التغييرات

```
$ git config --global user.email "[البريد الالكتروني]"
```

إعداد البريد الالكتروني المراد إدراجه في إجراءات تنفيذ أو "ارتكاب" التغييرات

## صنع المستودعات

بدء مستودع جديد أو الحصول على مستودع موجود مسبقاً عن طريق وصلة

```
$ git init [اسم المشروع]
```

صنع مستودع جديد بالمسمى المعطى

```
$ git clone [رابط المشروع]
```

تحميل مشروع بماضيه الشامل لكل النسخ الحالية والسابقة

## إحداث تغييرات

مراجعة التغييرات وتشكيل إجراء "ارتكاب" التغييرات

```
$ git status
```

إدراج جميع الملفات الجديدة أو التي تم تغييرها وتنتظر أن يتم ارتكابها

```
$ git diff
```

إظهار كل التغييرات في الملفات المتغيرة التي لم يتم إضافتها لقائمة الترحيل

```
$ git add [اسم الملف]
```

إضافة الملف لقائمة الترحيل للتحضير ليتم ارتكابها

```
$ git diff --staged
```

إظهار كل التغييرات في الملفات المتغيرة في قائمة الترحيل والمختلفة عن الموجودة في المستودع حالياً

```
$ git reset [اسم الملف]
```

إزالة الملف من قائمة الترحيل مع الحفاظ على المحتوى الجديد دون تغيير.

```
$ git commit -m "[تعليق توضيحي]"
```

ارتكاب التغييرات الموجودة في قائمة الترحيل وحفظها في ماضي النسخ

## التغييرات الجماعية

تسمية مجموعة من التغييرات المرتكبة وجمع الجهود المكتملة

```
$ git branch
```

إدراج قائمة بكل الفروع المحلية للمستودع الحالي

```
$ git branch [اسم الفرع]
```

تكوين فرع جديد

```
$ git switch -c [اسم الفرع]
```

التغيير إلى فرع معين وتحديث المسار الحالي

```
$ git merge [اسم الفرع]
```

دمج الفرع الحالي بفرع آخر ماضي

```
$ git branch -d [اسم الفرع]
```

حذف فرع معين

## إزالة وتغيير الملفات

تحريك وإزالة الملفات التي تتم متابعة نسخها

```
$ git rm [اسم الملف]
```

حذف الملف وإضافة عملية الحذف لقائمة الترحيل

```
$ git rm --cached [اسم الملف]
```

إلغاء عملية تتبع نسخ الملف مع الحفاظ على الملف محلياً (الملف موجود محلياً لكنه لا يظهر على Github)

```
$ git mv [اسم الملف الجديد] [اسم الملف الأصلي]
```

تغيير اسم الملف والتحضير لارتكاب التغييرات

## قمع تتبع الملفات

استثناء الملفات والمجلدات المؤقتة

```
*.log build/ temp-*
```

استخدام ملف نصي بمسمى `.gitignore` يمنع تتبع الملفات والمجلدات الغير مرغوبة بتحديد أنماط تسمية هذه الملفات

```
$ git ls-files --others --ignored --exclude-standard
```

سرد قائمة بكل الملفات التي تم تجاهلها في المشروع الحالي

## حفظ المتغيرات الصغيرة

التخزين الجانبي واستعادة المتغيرات غير المكتملة

```
$ git stash
```

تخزين مؤقت لكل الملفات المتغيرة التي تتم متابعتها

```
$ git stash pop
```

استعادة آخر الملفات المتغيرة التي تم تخزينها مؤخراً بشكل مؤقت

```
$ git stash list
```

سرد قائمة بكل عمليات التخزين المؤقتة التي لم يتم استعادتها بعد

```
$ git stash drop
```

التخلص من آخر الملفات المتغيرة التي تم تخزينها مؤخراً بشكل مؤقت

## مراجعة ماضي التغييرات

تصفح وتفحص تطور ملفات المشروع

```
$ git log
```

سرد قائمة بماضي النسخ للفرع الحالي

```
$ git log --follow [اسم الملف]
```

سرد قائمة بماضي نسخ ملف معين (يشمل تغيير الاسم)

```
$ git diff [الفرع الأول]...[الفرع الثاني]
```

عرض اختلافات المحتوى من فرع لآخر

```
$ git show [عملية ارتكاب]
```

عرض البيانات الوصفية وتغييرات المحتوى لعملية ارتكاب معينة

## التراجع عن عمليات الارتكاب

مسح الأخطاء واستبدال جزء من الماضي

```
$ git reset [عملية ارتكاب]
```

إلغاء جميع عمليات ارتكاب التغييرات بعد [عملية ارتكاب] مع الحفاظ على التغييرات محلياً

```
$ git reset --hard [عملية ارتكاب]
```

إلغاء جميع عمليات ارتكاب التغييرات واستعادة الملفات إلى وضعها السابق وقت عملية ارتكاب التغييرات

# مزامنة التغييرات

تسجيل علامة مرجعية للمستودع وتبادل ماضي النسخ

```
$ git fetch [علامة مرجعية]
```

تحميل كل ماضي المستودع من العلامة المرجعية

```
$ git merge [علامة مرجعية]/[فرع]
```

دمج فرع لعلامة مرجعية مع الفرع المحلي الحالي

```
$ git push [فرع] [alias]
```

رفع كل عمليات ارتكاب التغييرات للفرع المحلي على Github

```
$ git pull
```

تحميل ماضي العلامة المرجعية ودمج التغييرات

Made with ❤️ by 🐙s and friends



COMMUNITY

© 2024 GitHub, Inc. Jekyll & Primer.