

# Basic Github



Git คือ ?

Version Control แบบ Distributed ตัวหนึ่ง

เป็นระบบที่ใช้จัดเก็บและควบคุมการเปลี่ยนแปลงที่เกิดขึ้นกับไฟล์ชนิดใดก็ได้ Version control คืออะไร และทำไมคุณถึงต้องแคร์? Version control คือระบบที่จัดเก็บการเปลี่ยนแปลงที่เกิดขึ้นกับไฟล์หนึ่งหรือหลายไฟล์เพื่อที่คุณสามารถเรียกเวอร์ชันใดเวอร์ชันหนึ่งกลับมาดูเมื่อไรก็ได้

หนังสือเล่มนี้จะยกตัวอย่างจากไฟล์ที่เป็นซอร์สโค้ดของซอฟต์แวร์

แต่ขอให้เข้าใจว่าจริง ๆ แล้วคุณสามารถใช้ version control กับไฟล์ชนิดใดก็ได้

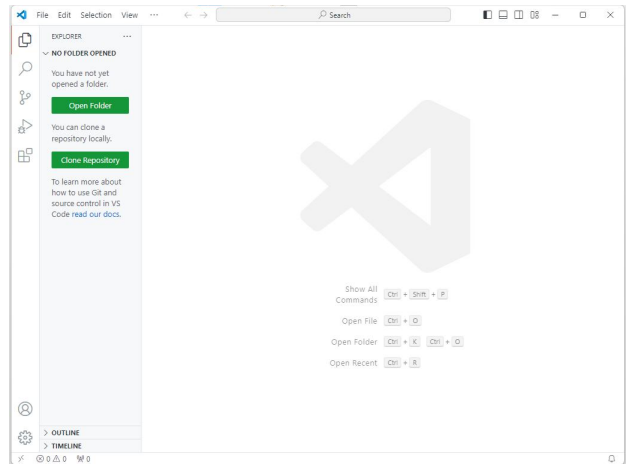
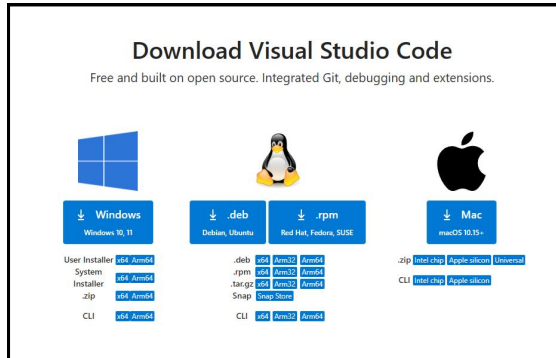
แล้วทำไมเราถึงต้องใช้ git

- Source Code ไม่มีวันหาย ย้อนกลับได้ตลอดเวลา
- ช่วยให้เวลาพัฒนางานเป็น team เป็นไปอย่างง่ายขึ้น
- สามารถดูได้ว่าส่วนไหนใครเป็นคนทำ
- สามารถทำงานรวมกันผ่าน Git Hosting ได้ (ตัวนี้ตอบโจทย์มาก)

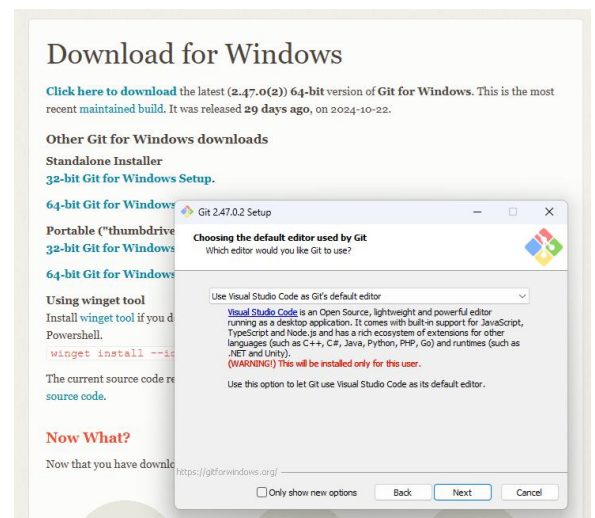
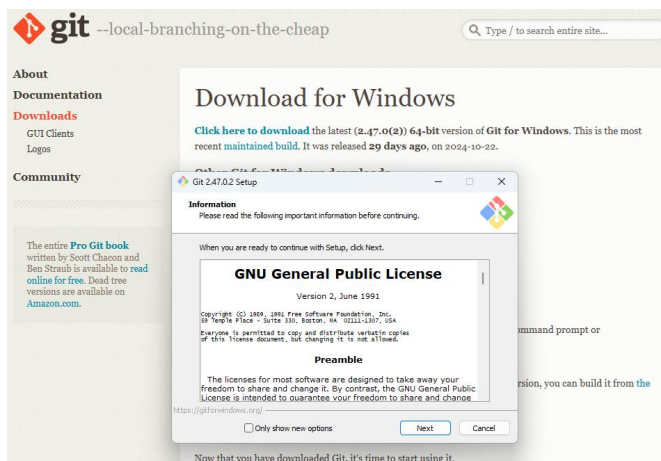
## basic command

เครื่องมือ

- ลงทะเบียนGit hub + Download git + Down VS code

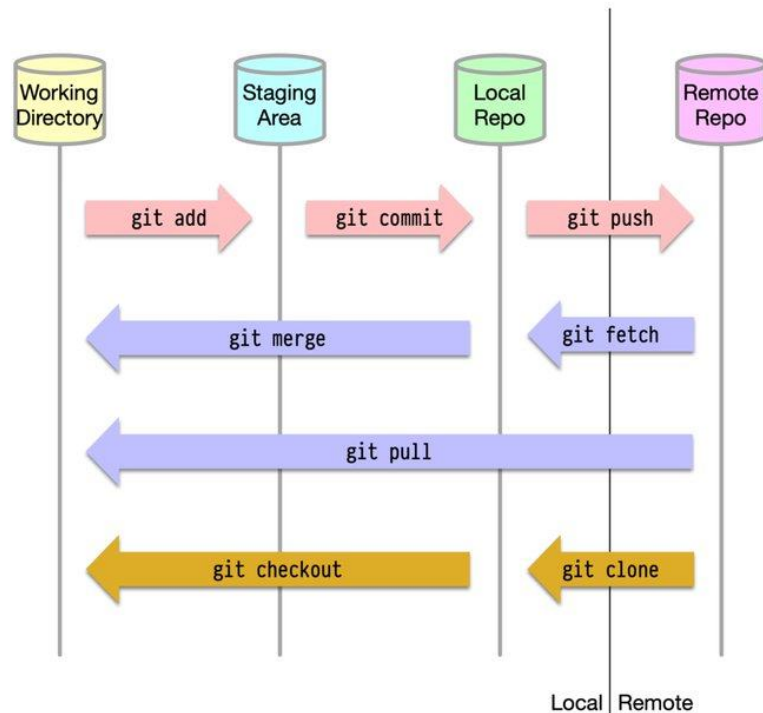


## VS Code

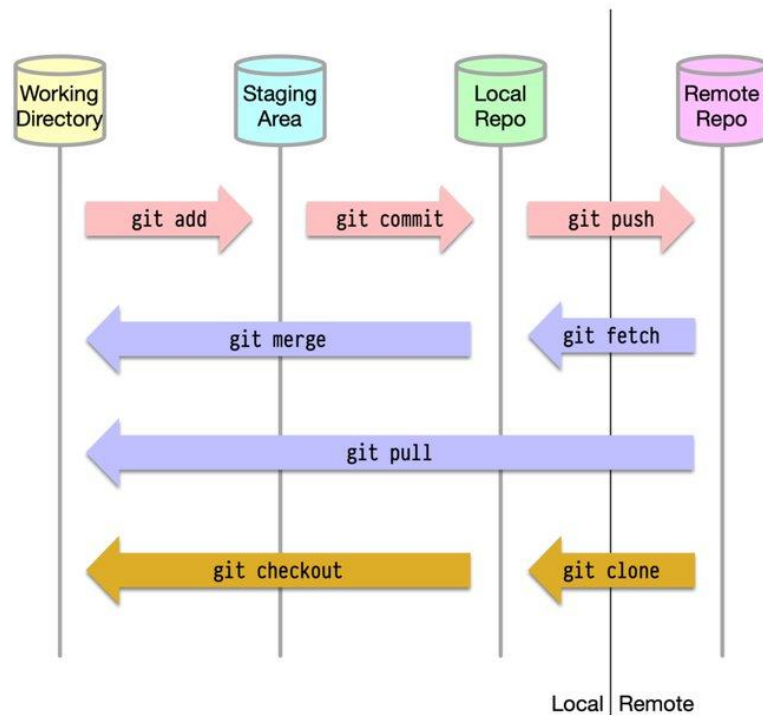


เลือก Use VS code  
as git default editor

## GitHub

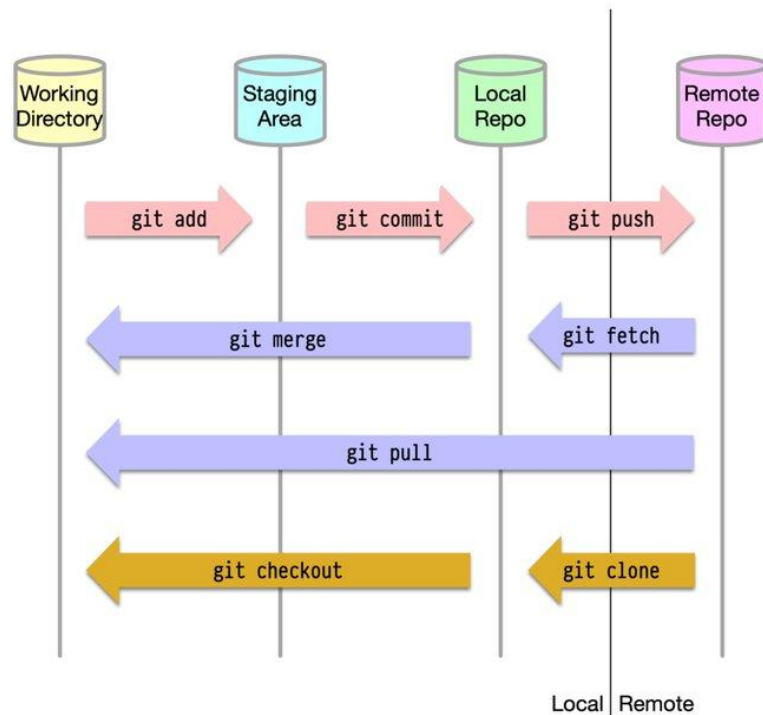


1. **Working Directory** The working directory is where your current project files reside. This is your main workspace where you create, edit, and delete files.
2. **Staging Area** The staging area is a temporary storage for changes that are meant to be included in the next commit. Think of it as a preview of your next commit snapshot.
3. **Local Repository** Your local repository is where all your project's commits are stored. This is your personal copy of the project history, which you can manipulate and manage independently of others.
4. **Remote Repo** "Remote" refers to a version of your repository that is hosted on a server and is typically used to share your work with others.



### สถานะของ Source Code

- Untracked เป็นสถานที่ Source Code ถูกเพิ่มเข้ามาใหม่และยังไม่ได้ถูกเก็บไว้ในระบบ
- Working Directory เป็นสถานะที่กำลังมีการเปลี่ยนแปลงหรือแก้ไข Source Code
- Staged เป็นสถานะที่ Source Code กำลังเตรียมที่จะ Commit เพื่อยืนยันการเปลี่ยนแปลงก่อนที่จะเก็บลงในสถานะ Local repository
- Local Repository เป็นสถานะที่มีการเก็บบันทึกข้อมูลการเปลี่ยนแปลง Source Code ลงใน git Repository ที่เป็น Local เครื่องตัวเอง
- Remote Repository เป็นสถานะที่มีการเก็บข้อมูลการเปลี่ยนแปลงของ Source Code ลงใน git Repository ที่เป็น host




### สถานะของ Source Code

- Untracked เป็นสถานที่ Source Code ถูกเพิ่มเข้ามาใหม่และยังไม่ได้ถูกเก็บไว้ในระบบ
- Working Directory เป็นสถานะที่กำลังมีการเปลี่ยนแปลงหรือแก้ไข Source Code
- Staged เป็นสถานะที่ Source Code กำลังเตรียมที่จะ Commit เพื่อยืนยันการเปลี่ยนแปลงก่อนที่จะเก็บลงในสถานะ Local repository
- Local Repository เป็นสถานะที่มีการเก็บบันทึกข้อมูลการเปลี่ยนแปลง Source Code ลงใน git Repository ที่เป็น Local เครื่องตัวเอง
- Remote Repository เป็นสถานะที่มีการเก็บข้อมูลการเปลี่ยนแปลงของ Source Code ลงใน git Repository ที่เป็น host

# การ Fork คือการ Copy Repository ของคนอื่นมาเป็นของตัวเอง


11 [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#)

 **python-qrcode** Public










[Watch](#) 121 [Fork](#) 680 [Star](#)

Fork your own copy of lincolnloop/python-qrcode

[main](#) 4 Branches 37 Tags  [Add file](#) [Code](#)

 **maribedran** Preparing release 8.0 ✓

456b01d · 2 months ago 443 Commits

	Fix quote issue	3 months ago
	Preparing release 7.4.2	last year
	Raise error if embedded image is provided and error correcti...	2 months ago
	Generic editorconfig	3 months ago
	Add ruff format to git ignore rev file.	3 months ago
	Poetry base files	3 months ago
	Preparing release 8.0	2 months ago
	Add license, readme and setup.py module	13 years ago
	Documentation updates and bring Poetry config to the sam...	3 months ago

About

Python QR Code image gen

[pypi.python.org/pypi/qrcode](#)

[Readme](#)

[View license](#)

[Activity](#)

[Custom properties](#)

4.4k stars

121 watching

680 forks

[Report repository](#)

Releases

37 tags

## Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Required fields are marked with an asterisk (\*).

Owner \*

 keeta-ka

Repository name \*

python-qrcode

✓ python-qrcode is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

Python QR Code image generator

☒ Copy the **main** branch only

Contribute back to lincolnloop/python-qrcode by adding your own branch. [Learn more.](#)

 You are creating a fork in your personal account.


Create fork

## Top repositories

 New

Find a repository...

 keeta-ka/picoFinance

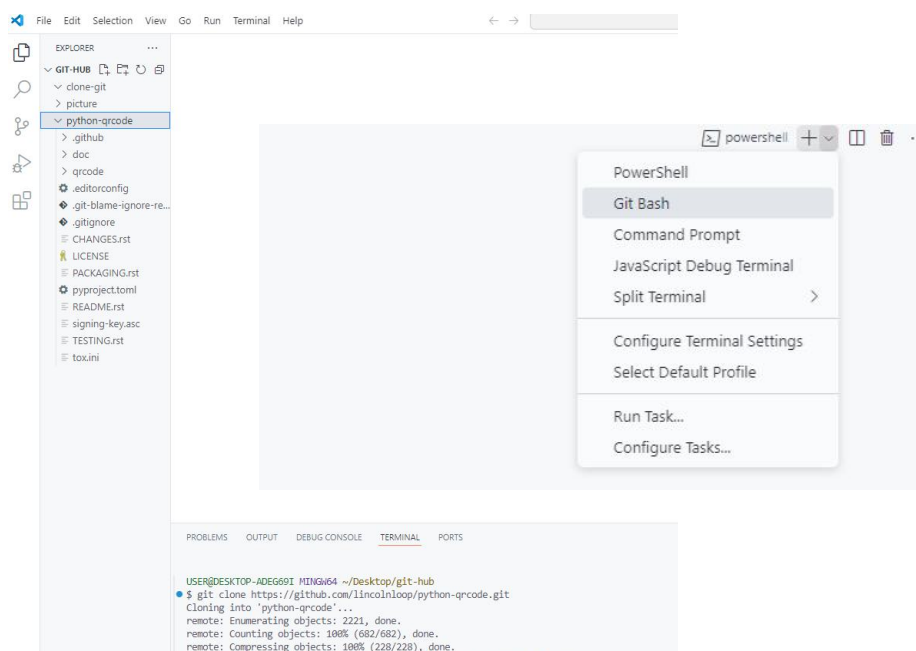
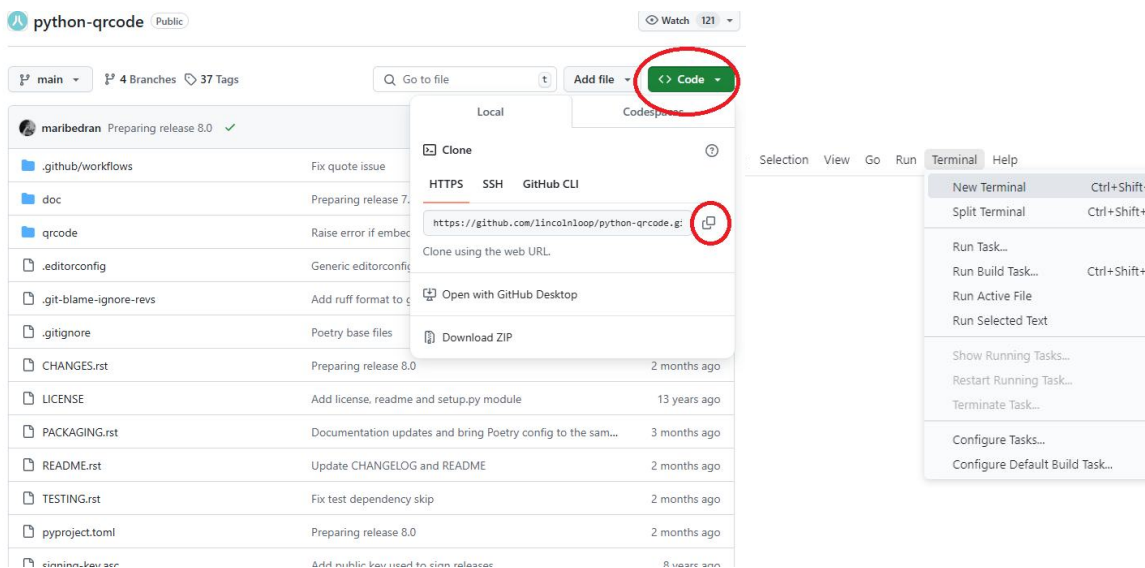
 keeta-ka/python-qrcode

## basic command

### คำสั่งพื้นฐาน 1

การดึง code จาก git hub มาใช้ศึกษา

- git clone xxxxxxxxxx Source จะอยู่ในชั้น Local Repostiory



git clone เป็นคำสั่งที่จะเป็นคำสั่งที่ใช้สำหรับ Download Source code ที่เราเอาไปเก็บไว้บน git host มาลงไว้ในเครื่องของเรา คำสั่งนี้จะคุณหาคูณตาคันมากสำหรับสายที่ชอบไปโหลด code คนอื่นมาลอง รนท ดู



## 1. git config

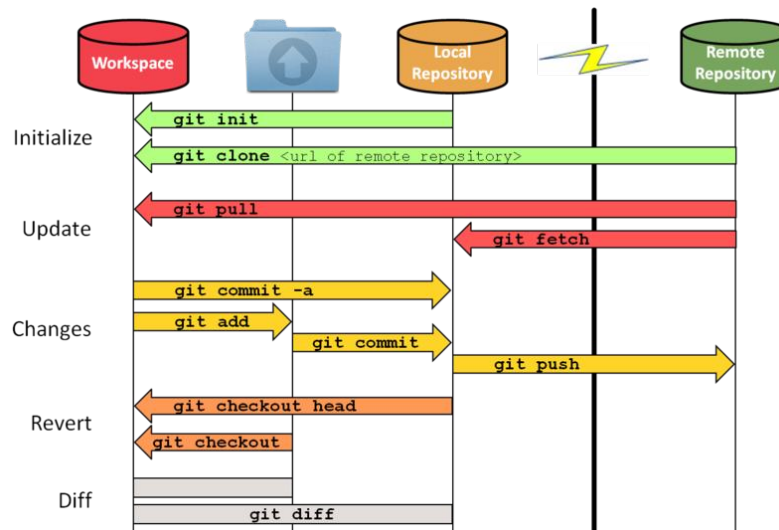
git config เป็นคำสั่งที่ใช้ไว้สำหรับกำหนดข้อมูลผู้ใช้เพื่อระบุตัวตน ของ git

ก่อนเริ่มใช้งานต้องตั้งค่าเบื้องต้นก่อน

git config --global user.name "xxxx"

git config --global user.email 'xxx@xxx.com'

git config --global --list



2. git init เป็นคำสั่งที่ใช้สำหรับสร้าง Folder ชื่อว่า .git เพื่อกำหนดว่า workspace ที่ทำงานนั้นให้เป็น Local repository

## git status

git status จากที่เราได้เพิ่ม File แล้วเราจะรู้ได้ยังไงว่าเราเพิ่ม File อะไรเข้าไปเป็นสถานะ Staged บ้าง เราจะใช้คำสั่ง git status เพื่อที่จะดูสถานะของแต่ละ File ว่าได้ถูก เพิ่มไปหรือยัง

# Home

<> Start writing code

## Start a new repository for keeta-ka

A repository contains all of your project's files, revision history, and collaborator discussion.

### Repository name \*

☐ Public

Anyone on the internet can see this repository


☒ Private

You choose who can see and commit to this repository

Create a new repository

รูปที่ 1

## Quick setup — if you've done this kind of thing before

 Set up in Desktop or ☐ HTTPS ☒ SSH



Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

## ...or create a new repository on the command line



```
echo "# test" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/keeta-ka/test.git
git push -u origin main
```

## ...or push an existing repository from the command line



```
git remote add origin https://github.com/keeta-ka/test.git
git branch -M main
git push -u origin main
```

รูปที่ 2

4. คำสั่งพื้นฐาน git Add ใช้สำหรับเพิ่มเข้าไปใน stage เพื่อรวบรวมไฟล์หรือไฟล์เดอร์สำหรับการ commit ครั้งต่อไป

git add main.html

git add . (ทั้งไฟล์เดอร์)

5. git commit เป็นการยืนยัน File ของเราที่เราทำการแปลงเป็นสถานะ Staged เพื่อที่จะเอาไปเก็บ Backup ไว้ใน Local Repository

6. คำสั่ง git push

ส่งไฟล์ที่ Commit ขึ้น remote repository

7. คำสั่ง git status

ตรวจสอบว่ามีไฟล์ใดที่ังไม่ add หรือมีการเปลี่ยนแปลงแต่ไม่ถูกจัดการบ้าง

## คำสั่งเพิ่มเติม

git fetch

git fetch เป็นคำสั่งที่เอาข้อมูลล่าสุดที่อยู่บน Remote Repository ของเรามาถึง Local Repository จะยังไม่ได้ Merge code จาก Remote Repository ลงมานะ

```
# รับการเปลี่ยนแปลงทุก Branch จาก Remote Repository
$ git fetch --all
# รับการเปลี่ยนแปลง Branch master จาก Remote Repository
# ที่ชื่อ origin
$ git fetch origin master
# แต่ทุกทีผมอยู่ไหน Branch ไหน ก็ใช้แค่คำสั่ง สั้นๆ ก็พอ
$ git fetch
```

git pull

git pull เป็นคำสั่งที่เอาข้อมูลล่าสุดที่อยู่บน Remote Repository ของเรามาถึง Local Repository แต่จะทำการ Merge code จาก Remote Repository ของเรามาลงใน Local

```
# แต่ทุกทีผมอยู่ไหน Branch ไหน ก็ใช้แค่คำสั่ง สั้นๆ ก็พอ
$ git pull
# รับการเปลี่ยนแปลง Branch master จาก Remote
# Repository ที่ชื่อ origin
$ git pull origin master
```

git rm

git rm เราได้พูดถึง git add ที่เปลี่ยน status file ให้เป็น Staged แล้วเราก็จะมาเรียนรู้เรื่องการทำ Unstaged กันบ้าง ใช้คำสั่ง git rm ซึ่งถ้าชื่อเต็มก็คือ git remove นั้นแหละ แต่ concept มันก็คล้าย git add นั้นแหละ คือ git rm ก็คือไม่ได้ลบ file แต่แค่เปลี่ยนสถานะให้เป็น unstaged เหมือนเดิมแค่นั้นเอง

```
# git unstaged เฉพาะ file
$ git rm --cached < file >
```