

【Git】checkout 與 switch 比較

我來幫你整理 `git switch` 和 `git checkout` 的差異，並用表格讓你快速比較。

1 歷史背景

- `git checkout` 是 Git 早期就有的指令，功能很多，但語意太複雜，既可以切換分支、也可以檢出檔案，容易讓初學者混淆。
- `git switch` (Git 2.23 之後加入) 是為了讓「切換分支」這個動作更直觀而出現的，專注於分支切換，不負責檔案檢出。

2 功能比較表

功能	<code>git checkout</code>	<code>git switch</code>
切換到已存在分支	<code>git checkout branch_name</code>	<code>git switch branch_name</code>
建立並切換到新分支	<code>git checkout -b new_branch</code>	<code>git switch -c new_branch</code>
檢出特定 commit / tag	<code>git checkout commit_hash</code>	不支援
檢出特定檔案版本	<code>git checkout commit_hash -- file.txt</code>	不支援
語意單純性	功能太多易混淆	專注於切換分支
推薦使用情境	老專案 / 習慣舊指令	新專案 / 想簡化操作

3 使用範例

切換到現有分支

```
# 傳統方式
git checkout dev

# 新方式 (推薦)
git switch dev
```

建立並切換到新分支

```
# 傳統方式
git checkout -b feature/login

# 新方式 (推薦)
git switch -c feature/login
```

4 為什麼官方推薦 `git switch`

- 讓指令語意更明確，不再把「檔案檢出」和「分支切換」混在一起。
- 搭配 `git restore` (處理檔案版本還原) 更符合直覺：
 - `git switch` → 專注切換分支
 - `git restore` → 還原檔案內容

簡單記：

- 切換分支 → 用 `git switch`
- 還原檔案 → 用 `git restore`

- 需要舊版本的行為 → 用 `git checkout`

太好了！以下是 `git switch` 和 `git restore` 如何取代舊指令 `git checkout` 的完整對照表，讓你更容易理解並開始使用新指令：

☐ `git checkout` 對照表：對應的新指令用法

原本 <code>git checkout</code> 用法	對應的新指令	功能說明
<code>git checkout <branch></code>	<code>git switch <branch></code>	切換到已存在的分支
<code>git checkout -b <new-branch></code>	<code>git switch -c <new-branch></code>	建立並切換到新分支
<code>git checkout <commit></code>	☐ (建議保留 <code>checkout</code>)	切換到某個 <code>commit</code> 狀態 (暫不取代)
<code>git checkout <file></code>	<code>git restore <file></code>	還原工作區中的單一檔案
<code>git checkout -- <file></code>	<code>git restore <file></code>	同上
<code>git checkout <commit> -- <file></code>	<code>git restore --source <commit> <file></code>	還原某個 <code>commit</code> 中的檔案
<code>git checkout .</code>	<code>git restore .</code>	還原所有工作區的檔案
<code>git checkout HEAD <file></code>	<code>git restore --source=HEAD <file></code>	還原最新 <code>commit</code> 的某個檔案
<code>git checkout HEAD^</code>	☐ (建議保留 <code>checkout</code>)	切換到上一個 <code>commit</code> (建議保留舊指令)

☐ 建議做法

任務	建議用法
切換分支	<code>git switch dev</code>
建新分支並切過去	<code>git switch -c feature/new-ui</code>
還原單一檔案	<code>git restore src/App.tsx</code>
還原某 <code>commit</code> 的檔案	<code>git restore --source=abc123 file.txt</code>
還原整個工作目錄	<code>git restore .</code>

☐ Git 版本要求

- `git switch` 和 `git restore` 需 Git 2.23 或以上
- 使用 `git --version` 檢查目前版本

🔄 修訂版本 #2

★ 由 treeman 建立於 7 🏆K🏆🏆 2025 15:54:28

🔪 由 treeman 更新於 5 🏆T🏆🏆 2026 10:51:46