**1. Nếu dự án thuộc tài khoản cá nhân (Personal Repository)**

1. **Vào repo trên GitHub.**
2. **Nhấn Settings (tab ở trên cùng).**
3. **Trong menu bên trái, chọn Collaborators.**
4. **Nhấn nút Add people.**
5. **Nhập tên tài khoản GitHub hoặc email của người bạn muốn thêm.**
6. **Chọn quyền (quyền mặc định là Write để có thể push code).**
7. **Nhấn Add <username> to repository.**

**👉 Người được mời sẽ nhận email/notification, họ phải Accept lời mời mới truy cập được repo.**

**🔹 2. Nếu dự án thuộc tổ chức (Organization)**

1. **Vào repo trên GitHub.**
2. **Nhấn Settings.**
3. **Chọn Manage Access.**
4. **Nhấn Invite a team or person.**
5. **Nhập tên người dùng GitHub hoặc team trong tổ chức.**
6. **Chọn quyền (Read, Write, Admin, Maintain, Triage).**
7. **Nhấn Invite.**

**🔑 Các mức quyền trong GitHub:**

* **Read → chỉ xem code, issues, pull requests.**
* **Triage → quản lý issues/pull requests (nhưng không được push code).**
* **Write → push code, tạo branch, quản lý issues.**
* **Maintain → quản lý repo (labels, settings cơ bản) nhưng không xóa repo.**
* **Admin → toàn quyền (bao gồm xóa repo, thêm/xóa thành viên).**

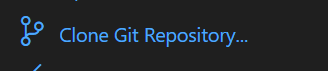
**CÁCH TẠO REPOSITORY (PROJECT) MỚI TRÊN GITHUB**

1. **Đăng nhập GitHub** (bạn đã đăng nhập rồi).
2. Ở góc trên bên trái (menu bên trái), bấm nút **New** (nút màu xanh lá, ngay chỗ "Top repositories").
   * Hoặc bạn có thể bấm vào dấu **+** (ở góc trên cùng bên phải cạnh avatar) → chọn **New repository**.
3. Điền thông tin cho repository:
   * **Repository name**: tên project (ví dụ: warehouse-management).
   * **Description (optional)**: mô tả ngắn về dự án.
   * **Public / Private**: chọn Public (ai cũng thấy) hoặc Private (chỉ bạn và người được cấp quyền).
   * Tick chọn **Add a README file** (nếu muốn có sẵn file README.md).
   * Có thể chọn **.gitignore** và **License** nếu cần.
4. Bấm **Create repository**.

**Sau khi tạo xong**

* Bạn sẽ được chuyển đến trang repository mới.
* GitHub sẽ hướng dẫn bạn **clone repository** về máy tính hoặc **upload code trực tiếp**.
* Nếu bạn đang code trên máy:
  + git init
  + git remote add origin https://github.com/<username>/<repo>.git
  + git add .
  + git commit -m "First commit"
  + git push -u origin main

TRÊN GIT HUB TẠO PROJECT MỚI:  
1. Liên kết project với Visual Studio:

Copy link project Git Hub => Mở => page link Git Hub vào tạo thư mục lưu Git Hub nhấn save

2. Tạo file code nhỏ ở Visual Studio:

Ấn newfile => Code => Save.

Khi thêm, sửa, xóa xong code thì:   
git add a1.cs \_ thêm file code lên gitbub  
git commit -m "Thêm nội dung khi sửa" \_ nội dung sửa sẽ hiện lên ở gitbub  
git push origin main\_ Đẩy nó lên github.  
git pull origin main \_ Tải các phiên bản mới nhất từ git hub về.

CÁCH ĐẨY DANH MỤC LÊN GITHUB:

Mở danh mục cần đẩy lên Visual rồi mở termital sau đó thao tác:

* PS C:\Users\Admin\Desktop\clean code> git remote add origin <https://github.com/NTHoa-NTH/abc.git> => THÊM DM LÊN GITHUB
* PS C:\Users\Admin\Desktop\clean code> git add . => THÊM
* PS C:\Users\Admin\Desktop\clean code> git commit -m "thêm dm clean code lên git hub" => VIẾT GỢI Ý HIỂN THỊ KHI THAO TÁC
* PS C:\Users\Admin\Desktop\clean code> git branch -M main TẠO BRANCH CHÍNH CHO BÀI
* PS C:\Users\Admin\Desktop\clean code> git push -u origin main => PUSH LÀ ĐẨY LÊN/ PULL LÀ TẢI VỀ

**CÁCH TẠO BRANCH NHỎ TRONG BÀI:**

* git pull origin main # cập nhật main mới nhất từ GitHub
* TẠO TÊN BRANCH: git checkout -b ten-branch
* ĐẨY BRANCH LÊN: git push -u origin ten-branch
* KIỂM TRA BRANCH HIỆN TẠI: git branch

**SAU KHI TẠO SỬA NỘI DUNG ĐẨY LẠI NỘI DUNG LÊN GITHUB:**

* SỬA NỘI DUNG Ở VSC
* # Lấy branch về
  + git checkout ten-branch
* # Code & sửa file
* # Kiểm tra thay đổi
  + git status => kiểm tra thay đổi
  + git diff => xem những đoạn nào đã sửa => nhấn q để thoát git diff
* # Stage + commit
  + git add .
  + git commit -m "Update chức năng login"
* # Push lên Git server
  + git push origin feature/login

**KHÔNG NHẤT THIẾT PHẢI PUSH BRANCH LÊN NGAY SAU KHI TẠO CÓ THẾ SỬA TRỰC TIẾP LUÔN**

* # Tạo branch nhỏ từ main
  + git checkout -b feature1
* # Sửa file Vd1.cs trong branch feature1
* # Lưu lại rồi commit
  + git add Vd1.cs
  + git commit -m "Sửa code trong feature1"
* # Vẫn chưa đẩy lên GitHub nhưng branch feature1 trên máy đã có thay đổi

**NẾU MUỐN CODE CÓ CẢ TRÊN MAIN**

**🔹 Cách 1: Merge bằng Git trên máy bạn**

# Đang ở branch cleanvd1

* git checkout main # chuyển về branch main
* git pull origin main # cập nhật main mới nhất từ GitHub
* git merge cleanvd1 # gộp code từ cleanvd1 vào main
  + nhấn esc để thoát
  + ghi “:wq” + enter để lưu
* git push origin main # đẩy lên GitHub

**LÀM VIỆC TRONG BRANCH NHỎ → MERGE VÀO MAIN → RỒI PUSH LÊN GITHUB**

* Bạn tạo branch nhỏ (feature1 chẳng hạn).
* Sửa code, commit nhiều lần trong branch nhỏ, nhưng **chưa push lên GitHub**.
* Khi xong việc:
  + git checkout main # về branch main
  + git pull origin main # đồng bộ main với GitHub cho chắc
  + git merge feature1 # gộp branch nhỏ vào main
  + git push origin main # đẩy main đã gộp lên GitHub

XÓA 1 FILE NHỎ TRÊN GIT

* # Chuyển sang branch main
  + git checkout main
* # Đồng bộ code từ GitHub về (tránh xung đột)
  + git pull origin main
* # Xóa file trong main
  + git rm clean1.cs
  + git commit -m "Xóa file clean1.cs trong branch main"
* # Đẩy lên lại GitHub
  + git push origin main