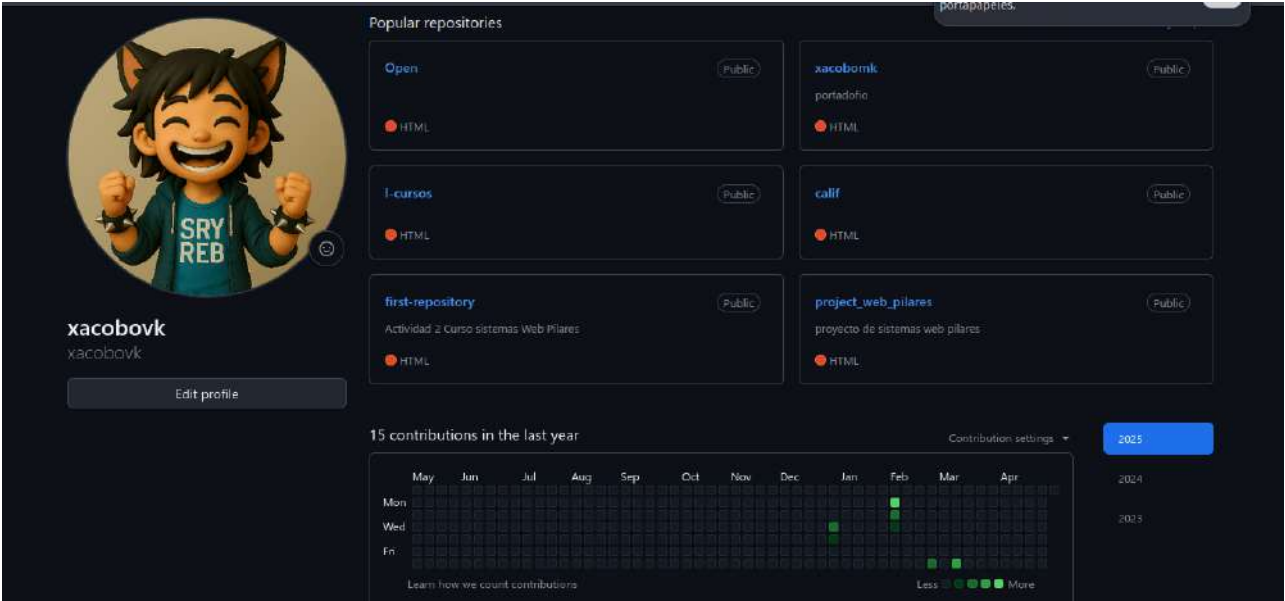
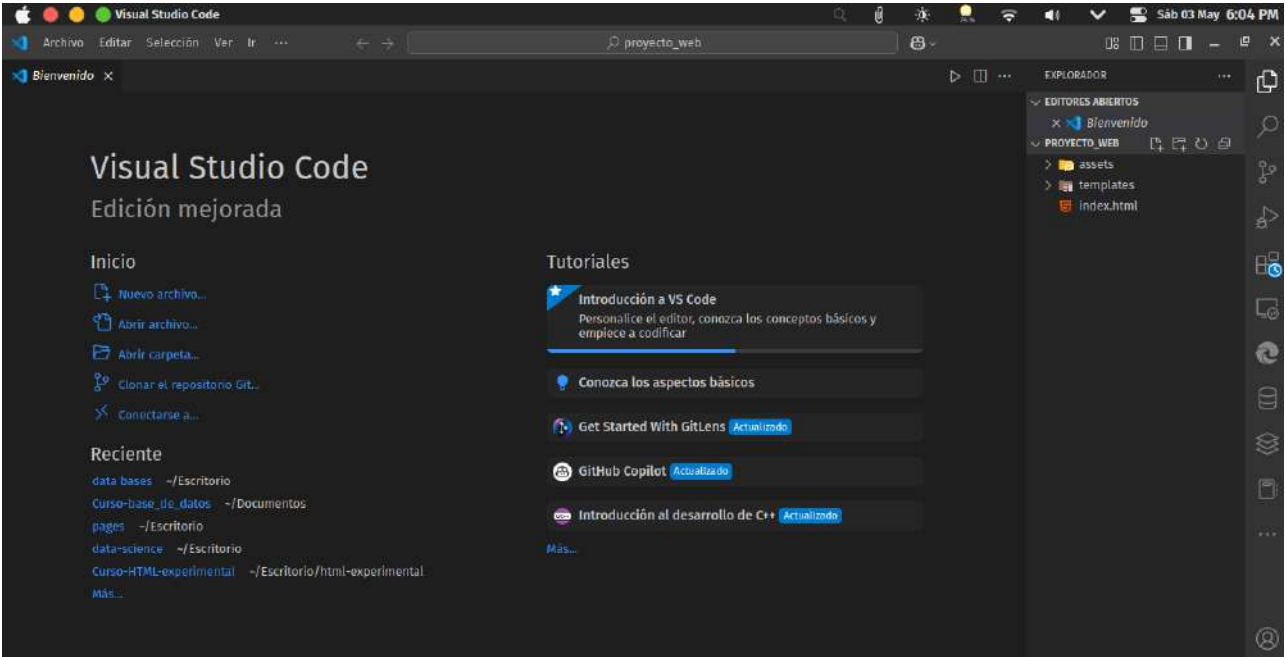


1. crear una cuenta de github



2. descargar proyecto



### 3. crear repositorio local

```

yuta@yuta in ~/Documentos/Curso-Git/semana 1/proyecto_web
$ git init
hint: Using 'master' as the name of the initial branch. This name is predeterminado
hint: esta sujeto a cambios. Para configurar el nombre de la rama inicial para usar en todos
hint: de sus nuevos repositorios, reprimiendo esta advertencia, llama a:
hint:
hint:   git config --global init.defaultBranch <nombre>
hint:
hint: Los nombres comúnmente elegidos en lugar de 'master' son 'main', 'trunk' y
hint: 'development'. Se puede cambiar el nombre de la rama recién creada mediante este comando:
hint:
hint:   git branch -m <nombre>
Iniciado repositorio Git vacío en /home/yuta/Documentos/Curso-Git/semana 1/proyecto_web/.git/

yuta@yuta in repo: proyecto_web on master [?] took 0s
$ git branch -M main

yuta@yuta in repo: proyecto_web on main [?] took 0s

```

### 4. Crear el archivo .gitignore

- Abre una terminal o visual studio code en la raíz de tu proyecto:
- Crea o abre el archivo .gitignore:

#### Desde la terminal

```
nano .gitignore
```

```

yuta@yuta in repo: proyecto_web on main
$ nano .gitignore

```

- Escribe los nombres de los archivos o carpetas que deseas ignorar, uno por línea. Por ejemplo:

```

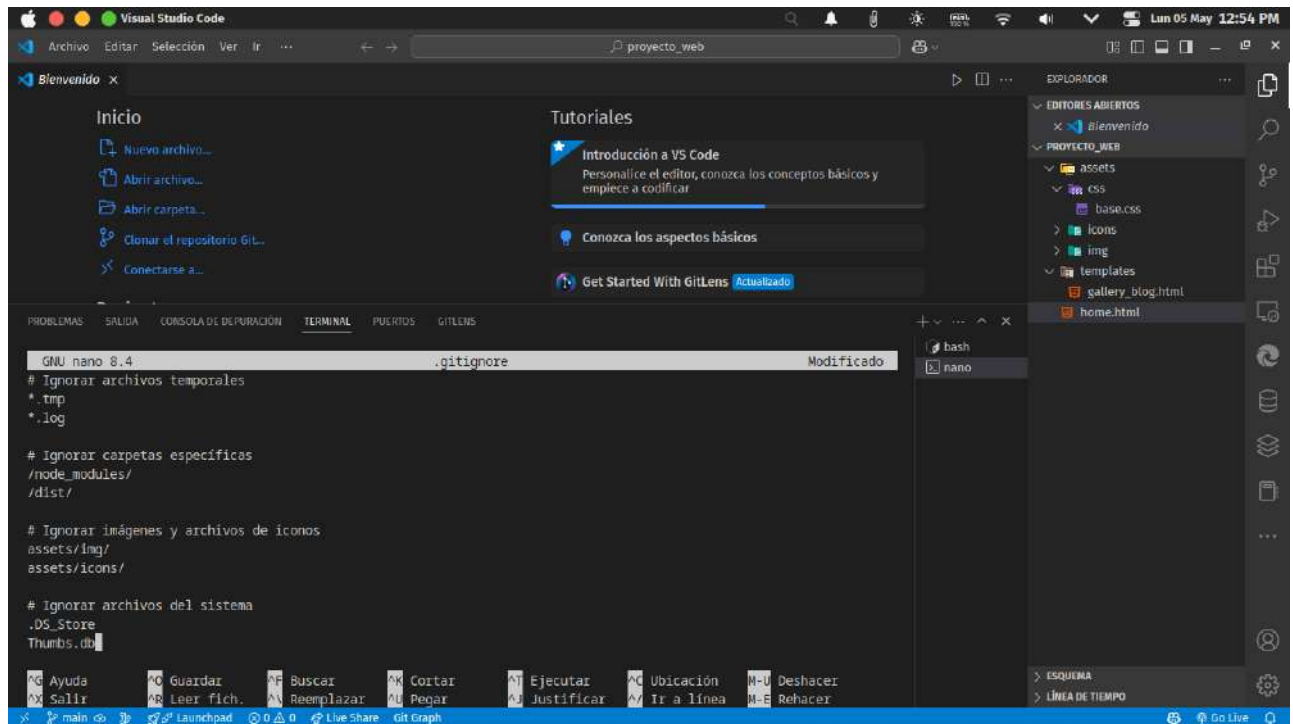
# Ignorar archivos temporales
*.tmp
*.log

```

```
# Ignorar carpetas específicas
/node_modules/
/dist/

# Ignorar imágenes y archivos de iconos
assets/img/
assets/icons/

# Ignorar archivos del sistema
.DS_Store
Thumbs.db
```

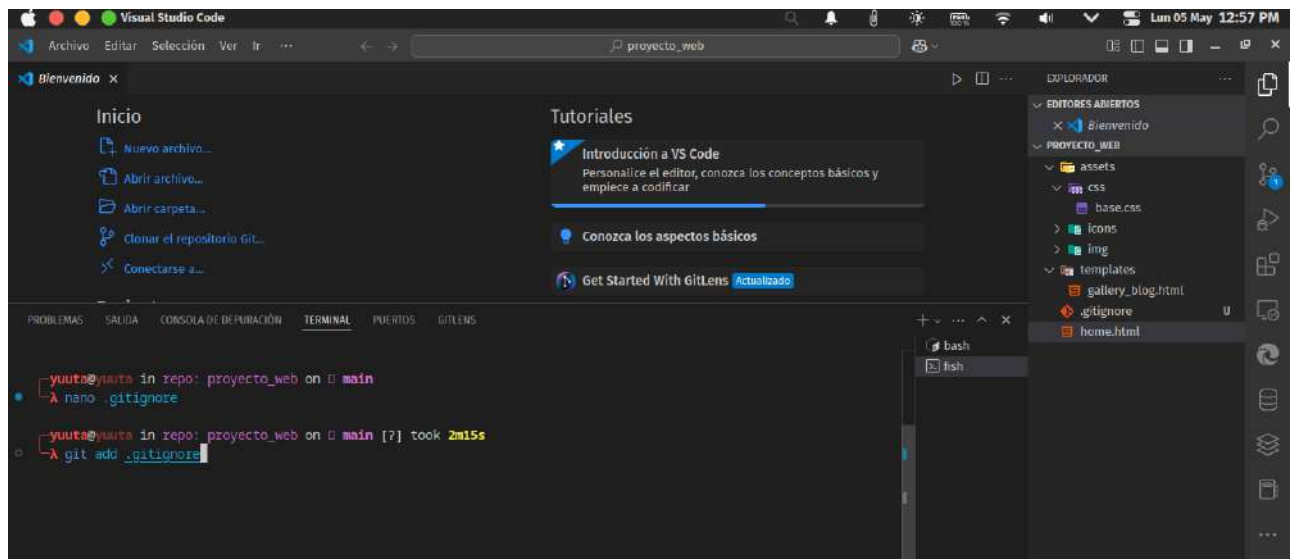


- Guarda el archivo y cierra el editor (en **nano**, presiona **CTRL+O**, luego **Enter**, y finalmente **CTRL+X**).

## 5. Confirmar el archivo **.gitignore** en el repositorio

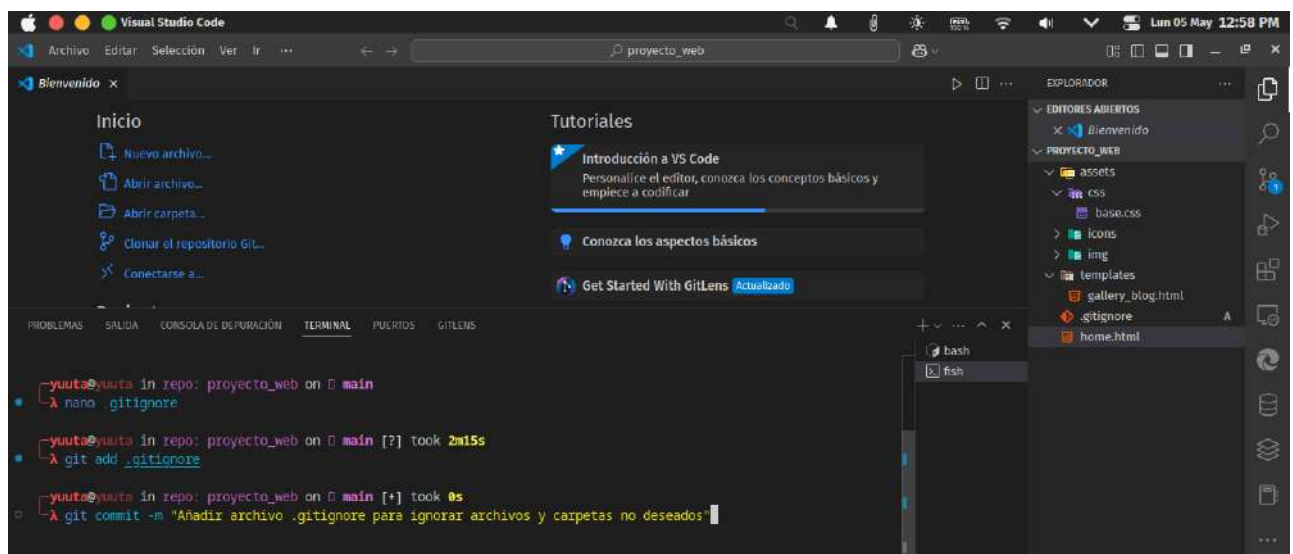
- Añade el archivo **.gitignore** al área de preparación:

```
git add .gitignore
```



- Realiza una confirmación:

```
git commit -m "Añadir archivo .gitignore para ignorar archivos y carpetas no  
deseados"
```



## 6. Verificar que los archivos ignorados no se rastrean

- Usa el comando `git status` para verificar que los archivos o carpetas especificados en `.gitignore` no aparecen en el área de preparación:

```
git status
```

```
• λ git add .gitignore

[yuuta@yuuta in repo: proyecto_web on □ main [+] took 0s]
• λ git commit -m "Añadir archivo .gitignore para ignorar archivos y carpetas no deseados"
[main 1b7bcac] Añadir archivo .gitignore para ignorar archivos y carpetas no deseados
1 file changed, 15 insertions(+)
create mode 100644 .gitignore

[yuuta@yuuta in repo: proyecto_web on □ main took 0s]
• λ git status
En la rama main
nada para hacer commit, el árbol de trabajo está limpio
```