

Proyecto final

Una vez que haya resuelto cada uno de los conjuntos de problemas del curso, es hora de implementar su proyecto final, ¡un programa Python propio! El diseño y la implementación de su proyecto depende totalmente de usted, aunque sujeto a estos requisitos:

- Su proyecto debe implementarse en Python.
- Tu proyecto debe tener una `main` función y tres o más funciones adicionales. Al menos tres de esas funciones adicionales deben ir acompañadas de pruebas que puedan ejecutarse con `pytest`.
 - Su `main` función debe estar en un archivo llamado `project.py`, que debe estar en la "raíz" (es decir, la carpeta de nivel superior) de su proyecto.
 - Sus 3 funciones personalizadas requeridas, además de, `main` también deben estar `project.py` y definirse en el mismo nivel de sangría `main` (es decir, no anidadas en ninguna clase o función).
 - Sus funciones de prueba deben estar en un archivo llamado `test_project.py`, que también debe estar en la "raíz" de su proyecto. Asegúrese de que tengan el mismo nombre que sus funciones personalizadas, precedido por `test_` (`test_custom_function`, por ejemplo, donde `custom_function` está una función que ha implementado `project.py`).
 - Le invitamos a implementar clases y funciones adicionales como mejor le parezca más allá del requisito mínimo.

```
def function_n():  
  
if __name__ == "__main__":  
    main()
```

`test_project.py`

```
def test_function_1():  
    ...  
  
def test_function_2():  
    ...  
  
def test_function_n():  
    ...
```

Eres bienvenido, pero no obligatorio, a colaborar con uno o dos compañeros de clase en tu proyecto. ([¡Es posible que desee colaborar con Live Share \(https://code.visualstudio.com/learn/collaboration/live-share\)](https://code.visualstudio.com/learn/collaboration/live-share) !) Pero un proyecto de dos o tres personas debería implicar el doble o el triple del tiempo y esfuerzo requerido por un proyecto de una sola persona.

Tenga en cuenta que el personal de CS50 audita las presentaciones a CS50P, incluido este proyecto final. Los estudiantes que violen **la política de Honestidad Académica** serán retirados del curso y considerados no elegibles para un certificado. A los estudiantes que

- En el mundo del software, implementar casi todo lleva más tiempo de lo esperado. Por eso, no es raro lograr en un período de tiempo fijo menos de lo esperado. ¿Qué podría considerar un buen resultado para su proyecto? ¿Un mejor resultado? ¿El mejor resultado?

Considere establecer objetivos hitos para mantenerse encaminado.

Cómo enviar

¡Debes completar los tres pasos!

Paso 1 de 3

Crea un vídeo corto (que no dure más de 3 minutos) en el que presentes tu proyecto al mundo. Su video **debe** comenzar con una sección de apertura que muestre:

- el título de su proyecto;
- Su nombre;
- sus nombres de usuario de GitHub y edX;
- tu ciudad y país;
- y la fecha en que grabó este video.

Si no está familiarizado con la sintaxis de Markdown, puede que le resulte útil [la sintaxis básica de escritura y formato \(https://docs.github.com/en/free-pro-team@latest/github/writing-on-github/basic-writing-and-formatting-syntax\)](https://docs.github.com/en/free-pro-team@latest/github/writing-on-github/basic-writing-and-formatting-syntax) de GitHub . Si está utilizando CS50 Codespace y se le solicita "Abrir en CS50 Lab", simplemente presione `cancel` para abrir en el Editor. También puede obtener una vista previa de su `.md` archivo haciendo clic en el ícono de 'vista previa' como se explica aquí: [Vista previa de Markdown en vscode \(https://code.visualstudio.com/docs/languages/markdown#_markdown-preview\)](https://code.visualstudio.com/docs/languages/markdown#_markdown-preview) . Los proyectos de software estándar `README` a menudo pueden tener miles o decenas de miles de palabras; El tuyo no tiene por qué ser tan largo, ¡pero al menos debe contener varios cientos de palabras que describan las cosas en detalle!

Su `README.md` archivo debe tener una extensión mínima de varios párrafos y debe explicar cuál es su proyecto, qué contiene y qué hace cada uno de los archivos que escribió para el proyecto, y si debatió ciertas opciones de diseño, explicando por qué las hizo. Asegúrese de dedicar suficiente tiempo y energía a redactar un `README.md` documento que documente su proyecto en profundidad. ¡Estar orgulloso de ello! Es probable que unas `README.md` 500 palabras sean suficientes para describir su proyecto y todos los aspectos de su funcionalidad. Si no puede alcanzar ese umbral, probablemente significa que su proyecto no es lo suficientemente complejo.

Ejecute el `submit50` siguiente comando desde su `~/project` directorio (o desde cualquier directorio que contenga `README.md` el archivo y el código de su proyecto, que también debe

