

Publicar tutoriales de LEARNR con GitHub

1. Creamos un repositorio público en GitHub con un archivo README: el nombre del repositorio será el nombre del paquete. Para la prueba he usado *PIDlearnr*. Luego podrían ser, por ejemplo, *TecClearnr*.
2. Copiamos la URL del repositorio usando el botón verde *Clone*. En nuestro caso: <https://github.com/rnoremias/PIDlearnr.git>. La estructura será siempre la misma, cambiará el nombre del repositorio.
3. Abrimos RStudio y creamos un nuevo proyecto Git siguiendo la ruta *File* → *New Project* → *Version Control* → *Git* y pegamos la URL del repositorio anterior. Especifico también la carpeta donde guardar el proyecto (en mi caso he creado una carpeta en el escritorio llamada *PIDlearnr*: C:\Users\Usuario\Desktop\PIDlearnr). Para poder crear el proyecto Git en RStudio previamente hay que instalarlo:
 - a. Seguir las instrucciones de la siguiente web que me ha sugerido el propio RStudio: <https://docs.posit.co/ide/user/ide/guide/tools/version-control.html>.
 - b. En <https://git-scm.com/downloads/win> descargo la versión para Windows (a día de hoy, Git-2.47.0-64-bit.exe) y la instalo (con las opciones por defecto).
 - c. El siguiente paso es activarlo en RStudio: en *Tools* → *Global Options* → *Git/SVN* activamos *Enable versión control interface for RStudio* y le damos el camino en *Git executable* (en mi caso, C:/Program Files/Git/bin/git.exe).
 - d. Si fuese necesario, añadirlo también al PATH de Windows (en mi caso no lo ha sido).
4. Una vez creado el proyecto cargamos los paquetes: *devtools*, *usethis*, *roxygen2*, *learnr*.
5. Creamos la estructura básica del paquete usando la carpeta en la que se ha creado el proyecto: *usethis::create_package("C:\\Users\\Usuario\\Desktop\\PIDlearnr")*. Si nos preguntaran si queremos sobrescribir el proyecto preexistente, decirle que no.
6. A continuación se abrirá una nueva sesión de RStudio, podemos cerrar la primera.
7. Vamos a crear el tutorial:
 - a. Ejecuto *usethis::use_tutorial("TC1learnr", "TC1", open=interactive())*. El primer argumento es el nombre del archivo que contiene el tutorial de learnr sin la extensión .Rmd; el segundo argumento es el título del tutorial que verán los usuarios.
 - b. Esto crea un archivo base de learnr que guarda dentro de la carpeta *inst* creada en C:\Users\Usuario\Desktop\PIDlearnr.
 - c. Podemos editar este archivo sin problemas, yo he copiado y pegado lo que ya tenía creado. También podríamos copiar el archivo .Rmd creado en la carpeta directamente. También podemos añadir en la carpeta aquellos datos que sean necesarios usar en el tutorial (aunque esto me da fallo) y puede ser el momento para modificar el archivo README (se escribe como en cualquier markdown).
 - d. Los pasos anteriores los puedo realizar tantas veces como tutoriales vaya a crear. Yo por ejemplo, para probar he creado un segundo tutorial: *usethis::use_tutorial("TC2learnr", "TC2", open=interactive())*.
8. Una vez creados todos los tutoriales que veamos oportunos, generamos el paquete: en la parte superior derecha de RStudio nos centramos en la pestaña *Build* y presionamos sobre *Install*. Si hubiese algún problema usar la opción *Clean and Install*.
9. Automáticamente aparecerán los tutoriales en la pestaña *Tutorial* de la parte superior derecha de RStudio. Se puede ejecutar desde dicha pestaña o, me gusta más, usando el código siguiente:
 - a. *learnr::run_tutorial("TC1learnr", package= "PIDlearnr")*.
 - b. *learnr::run_tutorial("TC1learnr", package= "PIDlearnr")*.
10. El siguiente paso es publicar el paquete creado en GitHub, para ello seleccionamos *commit* en la pestaña *Git* de la parte superior derecha de RStudio. En la ventana que sale, seleccionamos *Push* (flecha verde).
 - a. Esto no he conseguido que me funcione, así que lo que he hecho es subir manualmente (pinchando en el + al lado del botón verde *Code*) el contenido de la carpeta *PIDlearnr* creada en el tercer paso en el repositorio creado en el primer paso.
11. Cada vez que modifiquemos algo hay que repetir el proceso desde el paso 8.