



Actividad:

Instrucciones primera parte:

- 1.- Crea un nuevo repositorio en GitHub con el título “ProyectoHerramientasProductividad”. Aprovecha para crear un token de acceso personalizado, similar al de la sesión anterior.
- 2.- Guarda el token en un archivo llamado “token.txt” dentro de tu carpeta de proyecto.
- 3.- Copia la URL del repositorio remoto.
- 4.- Abre la terminal en RStudio y asegúrate de que estás en el directorio del proyecto actual.
- 5.- Inicializa el repositorio local con **git init ProyectoHerramientasProductividad**.
- 6.- Conecta tu repositorio local con el remoto usando git remote add origin **<https://github.com/tuUsuario/ProyectoHerramientasProductividad>**. Sustituye "tuUsuario" por tu nombre de usuario y utiliza el token de acceso que generaste.
- 7.- Escribe el reporte siguiendo las indicaciones del archivo **Sesion3_Proyecto.Rmd**, que incluye una lista de actividades sugeridas y ejemplos de código.
- 8.- Crea un script de shell **Sesion3_Script.sh** para automatizar la creación y carga del reporte en GitHub. Inicia el script con **#!/bin/bash/** para indicar que es un archivo ejecutable.



Actividad:

Instrucciones segunda parte:

- 9.- Establece la carpeta de tu proyecto como una variable de entorno y utiliza **read** para introducir el estado de los datos que deseas revisar.
- 10.- Ejecuta el script **limpiar_datos.R**, incluido en los archivos de la sesión, usando la variable **\$estado** como input para la limpieza de datos.
- 11.- Utiliza **find** para identificar archivos CSV modificados recientemente y **ls -lh** para listarlos de manera legible.
- 12.- Genera el documento de reporte ejecutando **Rscript -e con Sesion3_Proyecto.Rmd** y especifica el archivo de salida **reporte_proyecto.md**.
- 13.- Inicializa el repositorio con **git init** (si ya estaba inicializado, no habrá conflictos).
- 14.- Añade el archivo de reporte al área de preparación con **git add**, confirma los cambios con **git commit -m** y sube los cambios con **git push**.
- 15.- Confirma la carga con un mensaje de éxito usando **echo**.
- 16.- Ejecuta el script con **bash Sesion3_script.sh**.
- 17.- Verifica en GitHub que el archivo de reporte se haya cargado correctamente.