

Introducción General

El conjunto de scripts tiene como objetivo:

- Automatizar la **búsqueda y asignación de códigos CFR** a nombres de buques.
 - Facilitar la **modificación masiva de archivos Excel** con datos de buques.
 - Cargar los datos procesados en **tablas PostgreSQL**: `info_buques` y `licencias_cfr`.
-

Script 1: Autocompletado de CFR

Funciones Clave

1. `verificar_entorno_python()`

- Verifica si Python y las librerías necesarias (`selenium`, `webdriver-manager`) están disponibles.
- Intenta instalar automáticamente los paquetes si no existen.

✅ **Salida esperada:** Mensajes de confirmación o error claro si falta Python o pip.

⚠️ **Errores comunes:**

- Python no en PATH → ❌ Python no está instalado o no se encuentra en el PATH.
 - pip no disponible → `pip install selenium` falla silenciosamente.
 - Instalación bloqueada por firewalls o permisos.
-

2. `obtener_cfr(nombre_buque)`

- Llama al script Python `buscar_cfr.py` y devuelve el mejor CFR asociado al nombre de buque.

- Aplica lógica para elegir entre múltiples resultados (prefiere buques **activos**).

✓ **Salida esperada:** Código CFR, “consultar manualmente” o “CFR no encontrado”.

⚠ **Errores comunes:**

- Script Python falla → retorno: "error en python, revisar consola".
 - JSON mal formado desde Python → retorno: "Barco no encontrado".
 - Resultados ambiguos (múltiples CFR activos) → "consultar manualmente".
-

3. `procesar_buques(input_path)`

- Acepta archivo `.csv` o `.txt` con nombres de buques.
- Devuelve un `data.frame` con columnas `Buque` y `CFR`.

⚠ **Errores comunes:**

- Archivo incorrecto o malformado → `Formato de archivo no soportado`.
 - Campos vacíos o mal escritos → CFR no encontrado.
-

4. `completar_cfr_y_modificar_excel(hoja = 1, guardar = TRUE)`

- Permite completar los CFR faltantes en un archivo Excel y guardar los cambios.
- Detecta columnas automáticamente y actualiza directamente.

✓ **Salida esperada:** Archivo modificado con CFRs rellenados.

⚠ **Errores comunes:**

- Columnas `Buque` o `CodigoCfr` no existen o mal escritas → ✗ **Error: La hoja [...] debe contener columnas [...].**
- Valores `NA` o vacíos → no se procesan.
- Si `guardar = TRUE`, sobrescribe el archivo original sin confirmación adicional.

Script 2: Carga a PostgreSQL – **info_buques**

Este script:

1. Lee un archivo Excel con información técnica de buques.
2. Formatea fechas y columnas.
3. Inserta registros en la tabla **info_buques** de una base PostgreSQL.

Flujo del Script

- Conversión automática de **.xlsb** a **.xlsx** con **LibreOffice**.
- Mapeo de columnas esperadas → formato compatible con PostgreSQL.
- Conexión a NeonDB (PostgreSQL) y escritura en tabla temporal.
- Inserción final usando **ON CONFLICT DO NOTHING**.

Errores comunes:


- **LibreOffice** no instalado → conversión **.xlsb** falla.
- Columnas faltantes en Excel → celdas vacías en PostgreSQL.
- Credenciales incorrectas → **dbConnect()** falla.
- Nombres mal mapeados → columnas ignoradas al insertarse.
- Duplicados → se ignoran por la cláusula **ON CONFLICT**.

Script 3: Carga a PostgreSQL – **licencias_cfr**

Muy similar al script anterior, pero apunta a otra tabla: **licencias_cfr**.

Errores comunes similares:

- Conversión de `.xlsb`.
- Columnas mal escritas.
- Fallo en conexión o escritura.

 Recomendado:

- Validar nombre de columnas antes de ejecutar.
- Verificar tabla temporal antes del `INSERT`.

Requisitos del Sistema

R Packages

Instalables con:

```
install.packages(c("readxl", "writexl", "openxlsx", "dplyr", "tools", "DBI", "RPostgres",  
"lubridate"))
```

Python y Librerías

- Python ≥ 3.6
- pip
- selenium
- webdriver-manager

Instalación:

```
pip install selenium webdriver-manager
```


Software adicional

- LibreOffice (para convertir `.xlsb` \rightarrow `.xlsx`)
- PostgreSQL con acceso remoto habilitado (ej. NeonDB)

Conclusión

Este sistema está diseñado para ser modular y automatizable. Su principal fortaleza está en:

- Evitar errores humanos al completar CFR.
- Estándar robusto de procesamiento de Excel.
- Compatibilidad con bases de datos PostgreSQL.

 **Recomendación:** Siempre verificar visualmente los archivos antes y después de su uso para evitar sobrescrituras accidentales o entradas mal parseadas.