

# Acceso a Datos

Miguel Equihua

Xalapa, Ver., 9 de julio, 2024

## Tabla de contenidos

Obtención de Datos . . . . .	1
Manejo de claves confidenciales . . . . .	1

### Obtención de Datos

Las fuentes de datos, especialmente hoy, pueden ser muy variadas. Desde los datos que conseguimos directamente en campo a partir de mediciones directas o encuestas, hasta los datos que podemos obtener de *fuentes de datos abiertos*. Considero que será de tu interés explorar las distintas formas de interacción que las fuentes de datos implican para nuestros procesos de producción *científica reproducible*.

### Manejo de claves confidenciales

Un tema importante a cuidar es preservar la confidencialidad de claves, **tokens** y otras formas de identificación personal que puede implicar el proceso de acceso a datos en línea. Así que veremos eso como primer asunto. Queda claro que debemos evitar por todos los medios evitar poner esa información en carpetas o código que pueden acabar siendo registradas en Github en nuestro repositorio público. Haberte vacunado con `usethis::git_vaccinate()` ayuda en gran medida, pero desde luego no es remplazo a estar atentos a lo que estamos haciendo. La estrategia de registro de datos confidenciales que te propongo es la biblioteca **keyring**. Esta biblioteca accede al sistema de almacenamiento de credenciales de tu máquina desde *R*. La describen como una **API Independiente de la Plataforma** para acceder al depósito de credenciales del sistema operativo de tu máquina. [Este sitio explica que es una API \(Application Programming Interface\)](#). Actualmente **keyring** soporta: **Keychain** en *macOS*, **Credential Store** en *Windows*, **the Secret Service API** en *Linux*, soluciones simples (sin plataforma específica) desarrollados con variables de sistema o archivos encriptados e incluso ofrece la posibilidad de desarrollar algunas soluciones propias con sencillez. En nuestro

caso, básicamente usaremos dos funciones de esta biblioteca. Primero y desde la pantalla de **Consola** ejecuta:

```
key_set(service = "[dale un nombre]", username = "[el que desees]")
```

Esto hará el registro de tus credenciales en tu máquina, fuera de la vista. A partir de ese momento y mientras no elimines el registro explícitamente, estarán disponibles los datos que hayas registrado y los podrás obtener con:

```
key_get(service = "[dale un nombre]", username = "[el que desees]")
```

Esta última línea recupera los datos confidenciales, así que deberás cuidar usar las credenciales de inmediato y procurar no guardarlas y menos desplegarlas o habilitar medios para mostrarlas, durante el proceso.

Veamos un primer ejemplo con **INEGI**. Te sugiero ir a [https://www.inegi.org.mx/servicios/api\\_indicadores.html](https://www.inegi.org.mx/servicios/api_indicadores.html), en donde **INEGI** nos muestra un ejemplo de como acceder a los datos abiertos que *compilan, mantienen y custodian*. Deberás obtener un **token** personal, el mismo sitio de **INEGI** te dirá como obtenerlo. El ejemplo muestra como obtener datos de la **serie histórica** del indicador de la **Población total** de los Estados Unidos Mexicanos, en idioma español, en formato *JSON*. Una vez que los obtengamos calcularemos el promedio.

Lo primero que haremos es preparar el acceso a los datos con el **token** confidencial y obtenemos los datos, sin haber registrado la *URL* de acceso, pues como viste arriba, incluye tu **token**, así que habrá que manejarla con seguridad. El resultado de este *código* es una estructura de datos que ya no contiene información confidencial.

```
library(httr)
library(jsonlite)
library(rjson)
```

```
Attaching package: 'rjson'
```

The following objects are masked from 'package:jsonlite':

fromJSON, toJSON

```
library(keyring)
library(stringr)
suppressWarnings(suppressMessages(library(tidyverse)))

# Llamado al API de INEGI
url <- "https://www.inegi.org.mx/app/api/indicadores/desarrolladores/jsonxml/INDICADOR/1002000001"

# Obtiene los datos (usa los datos confidenciales sin mostrarlos ni guardarlos)
datosGenerales <- content(GET(str_replace(url,
                                           "\\[Aquí va tu Token\\]",
                                           key_get(service = "inegi_api",
                                                    username = "miguel_inegi"))),
                           "text")
```

La siguiente tarea que haremos ahora es simplemente arreglar los datos y ponerlos de la manera que requiero. Los datos son actualmente una base de datos *json*, que es una estructura parecida a un diccionario que tiene una etiqueta seguida de los datos que le corresponden.

```
prettify(datosGenerales)
```

```
{
  "Header": {
    "Name": "Datos compactos BISE",
    "Email": "atencion.usuarios@inegi.org.mx"
  },
  "Series": [
    {
      "INDICADOR": "1002000001",
      "FREQ": "7",
      "TOPIC": "123",
      "UNIT": "96",
      "UNIT_MULT": "",
      "NOTE": "1398",
      "SOURCE": "2,3,343,487,781,1668,1669,1670,1671,1672,1677,1678,1679,1714,3001",
      "LASTUPDATE": "24/01/2021 12:00:00 a. m.",
      "STATUS": null,
      "OBSERVATIONS": [
```

```

{
  "TIME_PERIOD": "1910",
  "OBS_VALUE": "15160369.0000000000000000000000",
  "OBS_EXCEPTION": null,
  "OBS_STATUS": "3",
  "OBS_SOURCE": "",
  "OBS_NOTE": "",
  "COBER_GEO": "0700"
},
{
  "TIME_PERIOD": "1921",
  "OBS_VALUE": "14334780.0000000000000000000000",
  "OBS_EXCEPTION": null,
  "OBS_STATUS": "3",
  "OBS_SOURCE": "",
  "OBS_NOTE": "",
  "COBER_GEO": "0700"
},
{
  "TIME_PERIOD": "1930",
  "OBS_VALUE": "16552722.0000000000000000000000",
  "OBS_EXCEPTION": null,
  "OBS_STATUS": "3",
  "OBS_SOURCE": "",
  "OBS_NOTE": "",
  "COBER_GEO": "0700"
},
{
  "TIME_PERIOD": "1940",
  "OBS_VALUE": "19653552.0000000000000000000000",
  "OBS_EXCEPTION": null,
  "OBS_STATUS": "3",
  "OBS_SOURCE": "",
  "OBS_NOTE": "",
  "COBER_GEO": "0700"
},
{
  "TIME_PERIOD": "1950",
  "OBS_VALUE": "25791017.0000000000000000000000",
  "OBS_EXCEPTION": null,
  "OBS_STATUS": "3",
  "OBS_SOURCE": "",
  "OBS_NOTE": "",

```

```

        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "1960",
        "OBS_VALUE": "34923129.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "1970",
        "OBS_VALUE": "48225238.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "1980",
        "OBS_VALUE": "66846833.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "1990",
        "OBS_VALUE": "81249645.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "1995",
        "OBS_VALUE": "91158290.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",

```

```

        "OBS_SOURCE": "",
        "OBS_NOTE": "422,425",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "2000",
        "OBS_VALUE": "97483412.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "9,425",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "2005",
        "OBS_VALUE": "103263388.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "49,425",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "2010",
        "OBS_VALUE": "112336538.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "115,425",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "2015",
        "OBS_VALUE": "119938473.0000000000000000000000",
        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "8230",
        "COBER_GEO": "0700"
    },
    {
        "TIME_PERIOD": "2020",
        "OBS_VALUE": "126014024.0000000000000000000000",

```

```

        "OBS_EXCEPTION": null,
        "OBS_STATUS": "3",
        "OBS_SOURCE": "",
        "OBS_NOTE": "",
        "COBER_GEO": "0700"
      }
    ]
  }
]
}

```

```

flujoDatos <- paste(datosGenerales, collapse = " ")

#Obtención de la lista de observaciones
flujoDatos <- fromJSON(flujoDatos)
flujoDatos <- flujoDatos$Series
flujoDatos <- flujoDatos[[1]]$OBSERVATIONS

#Generación del promedio de la lista de observaciones
datos<-0;
for (i in 1:length(flujoDatos))
{
  datos[i]<-flujoDatos[[i]]$OBS_VALUE
}

datos<-as.numeric(datos)
print(mean(datos))

```

```
[1] 64862094
```