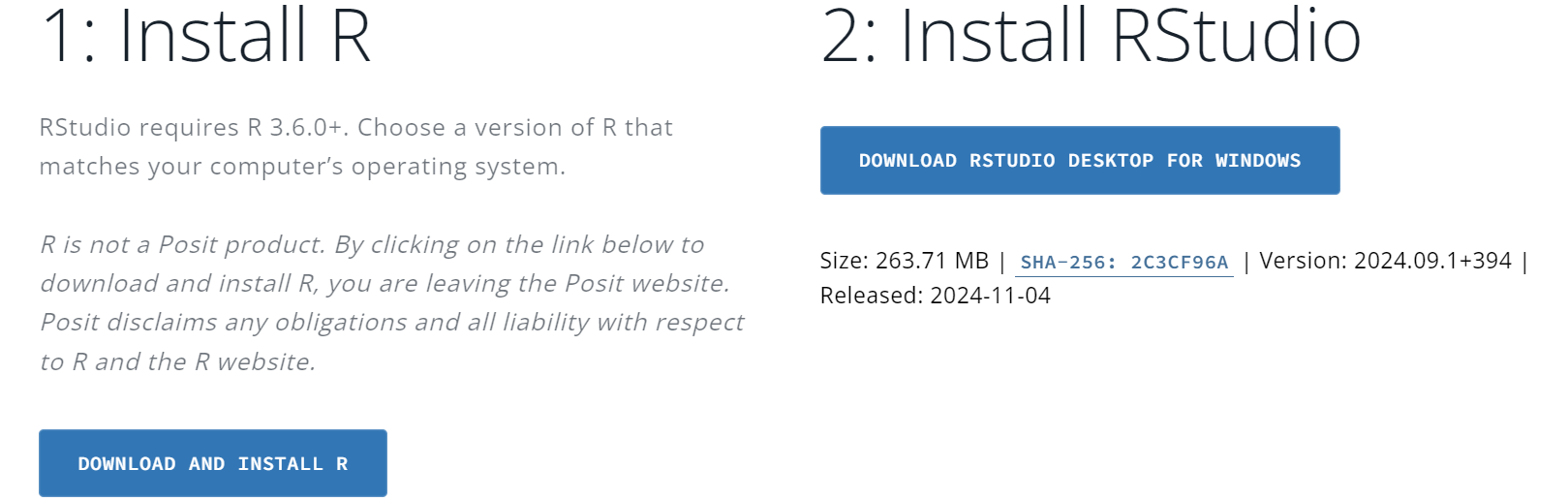
Análisis de datos con R

**Introducción**

R es un lenguaje de programación y entorno utilizado principalmente para análisis estadístico, modelado de datos y visualización. Es especialmente popular en la comunidad científica y de investigación.

**Instalación de R y RStudio**

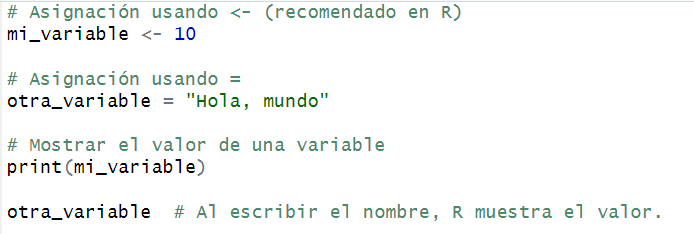
* [**RStudio Desktop - Posit**](https://posit.co/download/rstudio-desktop/)

****

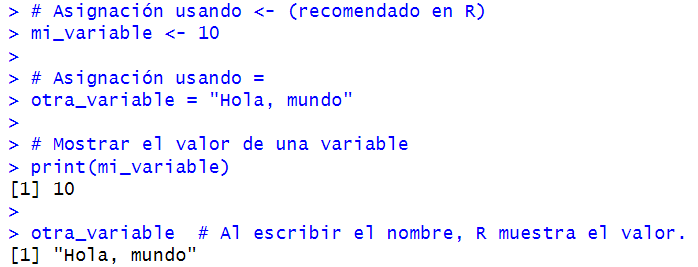


**Creación de Variables y Tipos de Datos**

En R, puedes crear variables asignándoles valores mediante el operador <- o el signo igual =.



Resultado en la consola:



Tipos de datos básicos

**Numéricos:** Para números con o sin decimales.

numero\_entero <- 42

numero\_decimal <- 3.14

**Texto (cadenas):** Almacenan caracteres o palabras.

texto <- "Hola Mundo

**Lógico (booleano):** Representa valores TRUE o FALSE.

logico <- TRUE

otro\_logico <- FALSE

**Vectores:** Colecciones de valores del mismo tipo.

vector\_numerico <- c(1, 2, 3, 4)

vector\_texto <- c("R", "es", "genial")

**Factores:** Datos categóricos.

categorias <- factor(c("bajo", "medio", "alto"))

La función **class()** para verificar el tipo de una variable.

class(calificaciones) # Devuelve "numeric"

class(numero\_entero) # Devuelve "logical"

Operaciones entre datos

**Operaciones Relacionales**

Se utilizan para comparar valores y producen resultados lógicos (TRUE o FALSE).

|  |  |
| --- | --- |
| < | Menor que |
| > | Mayor que |
| <= | Menor o igual que |
| >= | Mayor o igual que |
| == | Mayor o igual que |
| ¡= | Distinto de |

**Ejemplo:**

# Comparaciones numéricas

5 > 3 # Devuelve TRUE

5 < 3 # Devuelve FALSE

5 == 5 # Devuelve TRUE

5 != 4 # Devuelve TRUE

# Comparaciones de texto

"R" == "R" # Devuelve TRUE

"R" != "Python" # Devuelve TRUE

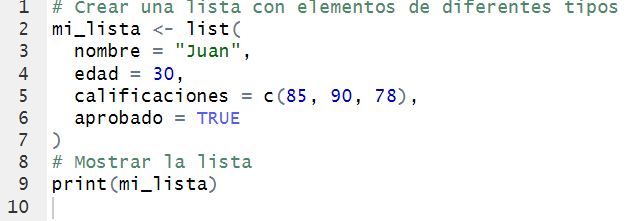
Listas

**¿Qué son las listas?**

Una lista en R es una estructura de datos que puede contener elementos de **diferentes tipos**, como vectores, matrices, funciones, y hasta otras listas. Es muy flexible y útil para almacenar **información heterogénea**.

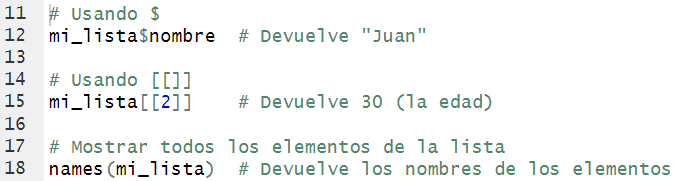
**Cómo crear una lista**

Se usa la función list() para crear una lista.

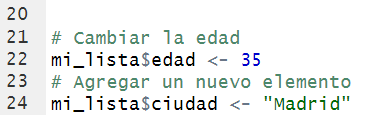


**Acceder a elementos de una lista**

Se accede a los elementos de una lista mediante el operador $ o corchetes dobles [[]].

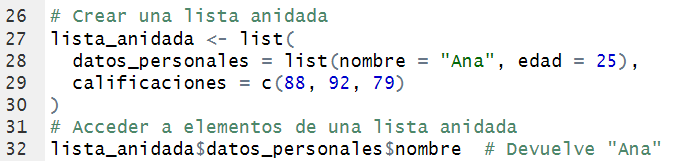


**Modificar elementos de una lista**



**Anidación de listas**

Se pueden incluir listas dentro de otras listas.



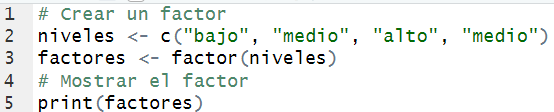
Factores

**¿Qué son los factores?**

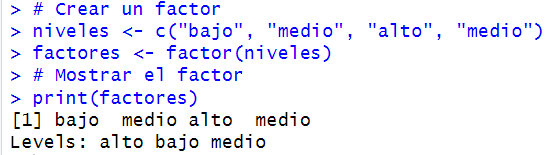
Un factor es un tipo de **dato categórico** que se usa para representar datos **cualitativos**. Los factores son útiles para analizar datos con categorías como "bajo", "medio", "alto" o "sí/no".

**Cómo crear un factor**

Se usa función factor() para convertir un vector en un factor.

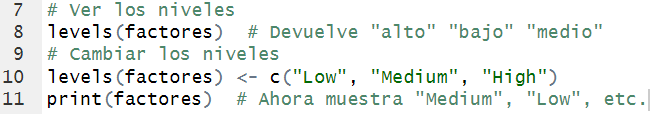


Salida:

****

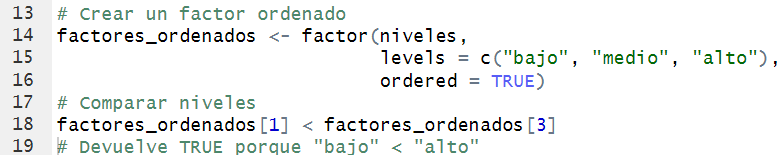
**Niveles en factores**

Los factores tienen niveles, que representan las categorías únicas.



**Ordenar factores**

Si las categorías tienen un orden lógico, se puede especificar con el argumento ordered = TRUE.

****

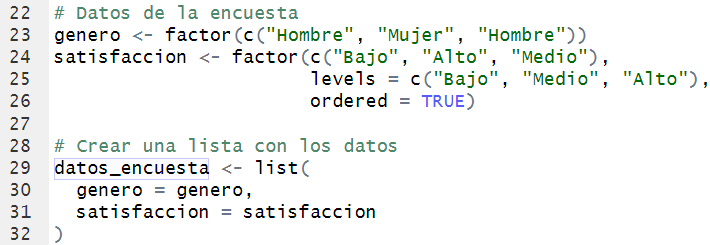
**Diferencias clave entre listas y factores**

|  |  |  |
| --- | --- | --- |
| Característica | Listas | Factores |
| Tipo de datos | Almacenan datos heterogéneos | Representan datos categóricos |
| Acceso a elementos | Mediante $ o [[]] |  |
| Usos comunes | Estructuras complejas y heterogéneas | Análisis estadístico |

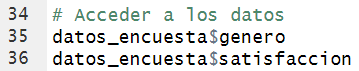
**Ejemplo combinado de una encuesta:**

Género de los participantes: c("Hombre", "Mujer", "Hombre").

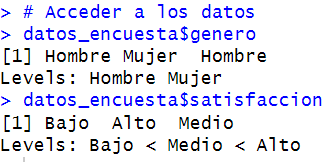
Nivel de satisfacción: c("Bajo", "Alto", "Medio").



Para acceder a los datos de satisfacción y participante:



Datos devueltos:



Búsquedas y Acceso en Listas

**Acceso a Elementos por Índice**

Puedes acceder a elementos de una lista por su índice numérico con [[ ]].

# Acceder al segundo elemento

datos\_encuesta[[2]] # Devuelve las edades

**Acceso por Nombre**

Si los elementos de una lista tienen nombres, puedes acceder a ellos con $ o [[ ]].

# Acceder por nombre

datos\_encuesta$nombres # Devuelve los nombres

datos\_encuesta[["edades"]] # También devuelve las edades

**Búsqueda Condicional**

Puedes combinar operadores lógicos para filtrar elementos dentro de una lista.

Ejemplo: Supongamos que quieres encontrar los nombres de los participantes mayores de 25 años:

# Obtener los nombres de participantes mayores de 25

datos\_encuesta$nombres[datos\_encuesta$edades > 25]

Búsquedas y Acceso en Factores

**Comparaciones con Factores**

Los factores ordenados pueden compararse con operadores relacionales.

# Buscar participantes que leen más que "Mensual"

datos\_encuesta$nombres[datos\_encuesta$frecuencia\_lectura > "Mensual"]

**Funciones Útiles para Factores**

levels(factor): Devuelve los niveles de un factor.

table(factor): Muestra un resumen de las frecuencias de cada nivel.

# Ver los niveles de frecuencia de lectura

levels(datos\_encuesta$frecuencia\_lectura)

# Tabla de frecuencia de lectura

table(datos\_encuesta$frecuencia\_lectura)

Funciones Comunes para Resúmenes de datos

**Resumen General**

La función summary() proporciona un resumen estadístico rápido para vectores, factores o listas.

# Resumen de edades

summary(datos\_encuesta$edades)

# Resumen de frecuencia de lectura

summary(datos\_encuesta$frecuencia\_lectura)

**Conteo de Elementos**

* **length():** Cuenta cuántos elementos hay en un vector o lista.
* **sum():** Suma los elementos de un vector numérico o el número de valores que cumplen una condición lógica.
* **which():** Devuelve los índices donde una condición es verdadera.

# Total de participantes

length(datos\_encuesta$nombres)

# Participantes con puntaje >= 8

sum(datos\_encuesta$puntaje\_satisfaccion >= 8)

# Índices de participantes que leen "Diario"

which(datos\_encuesta$frecuencia\_lectura == "Diario")

**Funciones de Agregación**

* **mean():** Calcula el promedio.
* **median():** Encuentra la mediana.
* **min() y max():** Encuentra el mínimo o máximo.

# Promedio de puntaje de satisfacción

mean(datos\_encuesta$puntaje\_satisfaccion)

# Edad máxima

max(datos\_encuesta$edades)

**Tablas de Frecuencias**

* table(): Crea una tabla con los conteos de cada categoría.

# Tabla de géneros literarios favoritos

table(datos\_encuesta$genero\_favorito)

**Ejemplo de Resumen con Funciones**

Suponer que se quiere hacer un resumen del caso práctico, incluyendo:

* Total de participantes.
* Edad promedio y máxima.
* Frecuencia más común de lectura.
* Género literario más popular.

# Total de participantes

total\_participantes <- length(datos\_encuesta$nombres)

# Edad promedio y máxima

edad\_promedio <- mean(datos\_encuesta$edades)

edad\_maxima <- max(datos\_encuesta$edades)

# Frecuencia más común de lectura

frecuencia\_comun <- names(which.max(table(datos\_encuesta$frecuencia\_lectura)))

# Género literario más popular

genero\_popular <- names(which.max(table(datos\_encuesta$genero\_favorito)))

# Mostrar resumen

cat("Resumen de la Encuesta:\n")

cat("Total de participantes:", total\_participantes, "\n")

cat("Edad promedio:", edad\_promedio, "\n")

cat("Edad máxima:", edad\_maxima, "\n")

cat("Frecuencia más común de lectura:", frecuencia\_comun, "\n")

cat("Género literario más popular:", genero\_popular, "\n")

* Salida:

Resumen de la Encuesta:

Total de participantes: 10

Edad promedio: 27.1

Edad máxima: 40

Frecuencia más común de lectura: Semanal

Género literario más popular: Ficción