

GRUPO #2

RUBY

Priscilla Azofeifa Durán
Luis Ortega Campos
Melanny Salazar Calvo
Esteban Baez



Agenda

- Historia
- Características
- Sintaxis
- Variables
- Tipos de datos
- Estructuras de control
- Gemas, librerías y más
- Manual de instalación



Historia

- Creado por Yukihiro Matsumoto en 1995.
- Inspirado en Perl y Python.
- Su popularidad se da por Ruby On Rails.



Características



01

Ruby es un lenguaje de programación interpretado.



02

Ruby es compatible con múltiples plataformas.



03

Ruby es un lenguaje de programación orientado a objetos.



04

Ruby cuenta con una comunidad activa de desarrolladores y un ecosistema de gemas.

Sintaxis



Cuenta con variables, constantes y métodos.



Los identificadores son "case sensitive", por lo que se distinguen en ellos las mayúsculas y minúsculas.



Los espacios en blanco son ignorados por Ruby, a excepción de cuando son parte de un String, ya que se detectan como texto.



Es posible agregar comentarios, pero se debe utilizar el símbolo numeral (#) al frente de una misma línea de código.

Palabras reservadas

BEGIN	do	next	then
END	else	nil	true
alias	elsif	not	undef
and	end	or	unless
begin	ensure	redo	until
break	false	rescue	when
case	if	return	while
def	in	self	_FILE_
defined	module	super	_LINE_

Variables

Las variables pueden ser de varios tipos, normalmente son definidas como `snake_case`.

Variables locales

Son variables definidas dentro de un método o bloque y solo son accesibles dentro de ese contexto.

Variables de instancia

Pertenecen a una instancia específica de una clase. Comienzan con el símbolo `@` y son accesibles en todos los métodos de la instancia.

Variables globales

Son compartidas por todas las instancias de una clase y se indican con `@@`.

TIPO DE DATOS



Estructuras de control

Condicionales

IF

ELSE

ELSIF

Condiciones anidadas

Condiciones compuestas

Ciclos

WHILE

FOR

ITERADORES

Arreglos

ARRAY

Otros

SWITCH

DEF

REDO

NEXT

BREAK

Ejemplos

```
case variable
when valor1
  # código si la variable es igual a valor1
when valor2
  # código si la variable es igual a valor2
else
  # código si la variable no coincide con ningún valor
end
```

Switch(case)

```
if condicion1
  # código si condicion1 es verdadera
elsif condicion2
  # código si condicion2 es verdadera
else
  # código si ninguna condición es verdadera
end
```

If, Else, Elif

```
if condicion_externa
  if condicion_interna
    # Código si ambas condiciones son verdaderas
  else
    # Código si solo la condicion_externa es verdadera
  end
else
  # Código si la condicion_externa es falsa
end
```

Condiciones anidadas

```
coleccion.each do |elemento|
  # código a ejecutar para cada elemento de la colección
end
```

Iteradores

Gemas, librerías y más

Ruby on Rails

Es un framework de desarrollo web. Algunas características y principios incluyen: convención de configuración, don't repeat yourself, MVC y gemas.

RSpec

Una biblioteca popular para realizar pruebas unitarias y de integración en Ruby.

ActiveRecord

Una parte fundamental de Rails que simplifica el acceso y la manipulación de bases de datos.

Devise

Una gema (Son bibliotecas y extensiones) para la autenticación de usuarios en Rails.

Nokogiri

Una biblioteca para analizar y manipular documentos XML y HTML.

MANUAL DE INSTALACIÓN

¡GRACIAS!

