

[Wikidot.com](#)

.wikidot.com

Share on      

[Edit](#) [History](#) [Tags](#) [Source](#)

[Explore »](#)

# Ruby Tutorial

...o como pasar un buen rato programando

- [admin](#)
  - [site manager](#)

[Create account](#) or [Sign in](#)



## Lección 1

- [Introducción](#)
- [Instalación](#)
- [El Primer Programa](#)
- [Números en Ruby](#)
- [Strings y diversión](#)
- [Variables](#)
- [Alcance de las variables](#)

## Lección 2

- [Introduciendo Datos](#)
- [Normas en los nombres](#)

- [Los métodos](#)
- [Los métodos: argumentos](#)
- [Rangos](#)
- [Arrays](#)

## Lección 3

- [Bloques](#)
- [Más malabares con strings](#)
- [Expresiones Regulares](#)
- [Condicionales](#)
- [Bucles](#)
- [Números Aleatorios](#)

## Lección 4

- [Clases y Objetos](#)
- [Accesores](#)
- [Ficheros: lectura/escritura](#)
- [Cargando librerías](#)
- [Herencia de clases](#)
- [Modificando clases](#)
- [Congelando objetos](#)
- [Serializando objetos](#)

## Lección 5

- [Control de acceso](#)
- [Excepciones](#)
- [Módulos](#)
- [Constantes](#)
- [Hashes y Símbolos](#)
- [La clase Time](#)

## Lección 6

- [self](#)
- [Duck Typing](#)
- [Azúcar Sintáctico](#)
- [Test de unidades](#)

## contacto

[e-mail](#)



## Introduciendo datos (gets)

Lecciones atrás vimos el método puts que saca datos por la pantalla. ¿Cómo podemos introducir nuestros propios datos? Para esto gets (get=coger, s=string) y chomp son de ayuda. Veamos el siguiente ejemplo:

```
# gets y chomp
puts "¿En qué ciudad te gustaría vivir?"
STDOUT.flush
ciudad = gets.chomp
puts "La ciudad es " + ciudad
```

El ejemplo superior, al ser ejecutado en SciTe, clickea en la pantalla de output y pon el nombre de tu ciudad favorita.

**STDOUT** es una constante global que almacena las salidas del programa. flush vacía cualquier dato

almacenado, y por lo tanto, limpiará cualquier resultado anterior.

**chomp** es un método para strings y gets almacena strings que provienen del teclado. El problema es que **gets** almacena lo escrito y el carácter '\n' (retorno de carro); **chomp** lo que hace es remover esa '\n'.

**RAILS:** los datos vienen de muchos sitios. En la típica aplicación de Rails, vienen de una base de datos. Como un desarrollador de Rails, puedes usar con frecuencia algunos de estos métodos, porque Rails recoge los datos que los usuarios escriben en los formularios Web.

## Ejercicio

Escribe un programa que pregunte por la temperatura en grados Fahrenheit. El programa usará este dato, y hallará el equivalente en grados Celsius. El resultado final lo mostrará en pantalla con dos decimales.

Nota: para formatear un resultado a dos decimales, hay dos opciones:

- usar el método **format**. Por ejemplo:

```
x = 45.5678
puts format("%.2f", x)
```

- otra forma es la función **round**:

```
puts (x*100).round/100.0
```

page\_revision: 8, last\_edited: 23 Feb 2010, 16:06 GMT-06 (279 days ago)

[EditTags](#) [History](#) [Files](#) [Print](#) [Site tools+ Options](#)

[Help](#) | [Terms of Service](#) | [Privacy](#) | [Report a bug](#) | [Flag as objectionable](#)

Powered by [Wikidot.com](#)

Unless otherwise stated, the content of this page is licensed under [Creative Commons Attribution-ShareAlike 3.0 License](#)

## Other interesting sites



### Robotech: Shadow Chronicles

The Future Is What We Make It...



### String Theory MUX

Connect to stringtheory.isunlimited.net, port 1407



### MarketingProject DETA



## Save Energy, Save Future

"The collabrative efforts for Energy saving"