

Taller de Ruby 1

Introducción

[Repositorio del curso](#)

[Verifica el entorno de desarrollo](#)

[Crea tu Workspace](#)

Bloque 1: RSpec

[Instalar la gema Rspec:](#)

[Crear archivos](#)

[Crear specs](#)

Bloque 2: Todito

[Archivos](#)

[Specs](#)

Introducción

Repositorio del curso

- Todo el código relacionado al curso se encuentra en Github
- <https://github.com/devacademy/ruby>

Verifica el entorno de desarrollo

- [Sublime Text 3](#)
- [Package Control](#)
- [Floobits](#)
- [Ruby](#)
- [RSpec](#)

Crea tu Workspace

- Crea un workspace con el nombre “ruby” en floobits.com
- Abre Sublime Text: Tools > Floobits > Join Workspace
- Ingresa la URL del workspace

Bloque 1: RSpec

Instalar la gema Rspec:

- En la consola escribir:

```
$ gem install rspec
```

Crear archivos

- Crear una carpeta con el nombre 'taller_1'
- Dentro de la carpeta taller_1, crear una carpeta 'rspec'
- Dentro de la carpeta rspec, crear dos archivos:
 - hola_mundo.rb
 - hola_mundo_spec.rb

Crear specs

- Definir el spec:
 - Dentro del archivo hola_mundo_spec.rb, llamar al archivo al que se hará un test:

```
require_relative './hola_mundo'  
RSpec.describe 'hola_mundo' do  
  end
```

- Dentro del describe, definir lo que se espera que haga el método


```
it 'saluda' do
  expect(hola_mundo).to eq('Hola Mundo!')
end
```
- Ingresar a la carpeta rspec desde la consola:


```
$ cd floobits/ruby/1_metodos/rspec
```
- Para correr los specs, escribir en la consola:


```
$ rspec hola_mundo_spec.rb
```
- Nos muestra los errores en el test
- Agregar el método hola_mundo en hola_mundo.rb:


```
def hola_mundo
  'Hola Mundo!'
end
```
- Correr de nuevo los specs, escribir en la consola:


```
$ rspec hola_mundo_spec.rb
```
- Los test pasan sin ningún error
- El ejemplo está en el [repositorio en GitHub](#).

[Aquí un Cheatsheet de RSpec](#)

Bloque 2: Todito

En este juego se da un castigo de acuerdo al número en el dado que te haya tocado:

- Si cae 1 (T): Toman todos
- Si cae 2 (O): El jugador que lanzó el dado ordena que tome otro jugador.
- Si cae 3 (D): Toma el jugador que está a la derecha del jugador que tiró el dado.
- Si cae 4 (I): Toma el jugador que está a la izquierda del jugador que tiró el dado.
- Si cae 5 (T): Toman todos
- Si cae 6 (O): El jugador que lanzó el dado ordena que tome otro jugador.

Archivos

- Copiar los archivos de la carpeta [todito](#)
- Dentro de la carpeta todito, tenemos dos archivos:
 - todito.rb: Donde estará el programa del juego
 - todito_spec.rb: Donde estarán todos los test que probarán el funcionamiento del juego.

Specs

- Ejecutar los specs, escribir en la consola:


```
$ rspec todito_spec.rb
```
- Crear un método lanzar_dado:
 - Devuelve el valor que se obtiene al lanzar un dado
 - El valor debe ser al azar (random)

- El valor debe estar en un rango del 1 al 6.
 - En ruby, el método `rand(x)` te devuelve un número desde el 0 hasta el número anterior a `x` (es decir, hasta `x - 1`)
- Ejecutar los specs de nuevo
- Crear un método `definir_castigo`:
 - Tiene que tomar como argumento el número obtenido al lanzar el dado.
 - Tiene que cumplir con las reglas del juego.
- Completar specs evaluando todos los casos (reglas del juego) definidas en el método.
- Ejecutar los specs para comprobar que el programa funcione correctamente