Unidad 4

4.1) Arreglos en Ruby

Temas: Crear Arreglos. Acceder a un elemento en un arreglo: índice; índices negativos;. Contar los elementos de un arreglo, rangos. Acceder secuencialmente a los elementos en un arreglo: .each; while; for. Acceder secuencialmente a los elementos en un arreglo Insertar y borrar elementos dentro de un arreglo: .push y .pop. Membresía: Operaciones sobre los elementos de un arreglo.

Creación de arreglos

Ya hablamos en las primeras unidades escrito acerca del los arrays o arreglos, habíamos dicho que eran al menos, bidemensionales, es decir, tienen una posición (índice) y un valor.

Valor	Aguila	Bandurria	Chincol	Cóndor
Índice/posición	0	1	2	3

En este caso cada pájaro tiene su posición en el índice en el array. Son 4 elementos, el 0 (cero) también cuenta.

Aguila tiene posición indice 0

Bandurria posición índice 1

Chincol posición índice 2

Cóndor posición índice 3

Si lo llevamos a código quedaría:

```
pajaros = [“Aguila”,”Bandurria”, “Chincol”, “Cóndor”]
```

Acceder a un elemento del array, indice, índice negativo

En el siguiente ejemplo vamos a imprimir nuestro array pájaros, y accederemos a diferentes elementos del array, también veremos cómo conocer cuántos objetos hay en un array. Finalmente establecemos rangos entre los elementos del array.

arrayBasico.rb — C:\Users\hacking\Downloads\ruby — Atom

File Edit View Selection Find Packages Help

Project

- 06Hashes.rb
- 07Hashes.rb
- 08Math2.rb
- 09Math2.rb
- holamundo.rb
- 2
  - bonus
  - .DS_Store
  - 08Get.rb
  - 09Estructura.rb
  - 010Estructura.rb
  - 011Metodo.rb
  - 012Rescate.rb
  - 013For.rb
  - 014While.rb
  - 015Each.rb
  - 016While.rb
  - 017Case.rb
  - 018For.rb
  - 020Metodo.rb
  - 021Var.rb
  - 022Arrays.rb
  - 023Arraypush.rb
  - arrayBasico.rb
  - iteracionesSobreArrays.rb
  - rangos.rb

holamundo.rb | 09Math2.rb | iteracionesSobreArrays.rb | arrayBasico.rb | 017Case.rb | 018For.rb | 010Estructura.rb

```ruby

pajaros = ["Aguila","Bandurria", "Chincol", "Cóndor"] #creamos nuestro array

puts  "Imprime el array completo"
puts pajaros

puts  "Imprime Bandurria,elemento 1 del array"

puts pajaros[1] #accedemos a un elemento del array

puts  "Imprime Cóndor,elemento -1 del array"

puts pajaros[-1] #Podemos usar índice negativo, imprimirá Cóndor

puts  "Imprime la cantidad de elementos del array"

puts pajaros.length

puts  "Imprime el primer elemento y el último de un array"

puts pajaros.first
puts pajaros.last

puts  "Rango:  imprime los valores entre el 2 y 3. Chincol y Cóndor "

puts pajaros[2..3] #establecemos un rango de objetos que nos interesa imprimir o seleccionar

```

2\arrayBasico.rb 24:68 LF UTF-8 Ruby GitHub Git (0)

Escribe aquí para buscar

ESP 14:14 07/02/2021

arrayBasico.rb — C:\Users\hacking\Downloads\ruby — Atom

File Edit View Selection Find Packages Help

Project

- 03Var.rb
- 04Time.rb
- 05Error.rb
- 06Hashes.rb
- 07Hashes.rb
- 08Math2.rb
- 09Math2.rb
- holamundo.rb
- 2
  - bonus
  - .DS_Store
  - 08Get.rb
  - 09Estructura.rb
  - 010Estructura.rb
  - 011Metodo.rb
  - 012Rescate.rb
  - 013For.rb
  - 014While.rb
  - 015Each.rb
  - 016While.rb
  - 017Case.rb
  - 018For.rb
  - 020Metodo.rb
  - 021Var.rb
  - 022Arrays.rb
  - 023Arraypush.rb

holamundo.rb | 09Math2.rb | arrayBasico.rb | 03Var.rb | 017Case.rb | 018For.rb | 010Estructura.rb

```ruby
titulo1 = "Módulo introducción a Ruby"
titulo2 = "2021"
titulo3 = "CHILE"



puts titulo1
puts titulo2
puts titulo3
pajaros = ["Aguila","Bandurria", "Chincol", "Cóndor"] #creamos nuestro array

puts pajaros #imprime todos los elementos del array
puts pajaros[1] #accedemos a un elemento del array, bandurria
puts pajaros[-1] #Podemos usar índice negativo, imprimirá Cóndor


puts pajaros.first #permite conocer el primer valor del array
puts pajaros.last #permite conocer el último valor del array

```

2\arrayBasico.rb* 20:1 LF UTF-8 Ruby GitHub Git (0)

Escribe aquí para buscar

ESP 13:26 07/02/2021

Creación de arrays con rango

Podemos crear también arrays con rangos, por ejemplo rangos numéricos de x número a y número, de una letra del abecedario a cualquier otra. Luego podemos conocer el contenido del array o imprimir posición y valores como lo vimos en el ejemplo anterior.

rangos.rb — C:\Users\hacking\Downloads\ruby — Atom

File Edit View Selection Find Packages Help

Project

- 03Var.rb
- 04Time.rb
- 05Error.rb
- 06Hashes.rb
- 07Hashes.rb
- 08Math2.rb
- 09Math2.rb
- holamundo.rb
- 2
  - bonus
  - .DS_Store
  - 08Get.rb
  - 09Estructura.rb
  - 010Estructura.rb
  - 011Metodo.rb
  - 012Rescate.rb
  - 013For.rb
  - 014While.rb
  - 015Each.rb
  - 016While.rb
  - 017Case.rb
  - 018For.rb
  - 020Metodo.rb
  - 021Var.rb
  - 022Arrays.rb
  - 023Arraypush.rb

holamundo.rb | 09Math2.rb | rangos.rb | 03Var.rb | 017Case.rb | 018For.rb | 010Estructura.rb

```ruby
titulo1 = "Módulo introducción a Ruby"
titulo2 = "2021"
titulo3 = "CHILE"




puts titulo1
puts titulo2
puts titulo3

puts "Creación de rangos numericos y alfabéticos"

numeros = (10..32).to_a #creamos un array llamado numeros que tiene como valores el rango entre 10 y 32
puts numeros
puts "Elementos en el array numérico:"
puts numeros.length #podemos conocer cuántos elementos hay en el array


letras = ('c'..'j').to_a #creamos un array llamado letras que tiene como valores el rango entre c y j
puts letras
puts "Elementos en el array de letras:"
puts letras.length #podemos conocer cuántos elementos hay en el array

```

2\rangos.rb 20:102 LF UTF-8 Ruby GitHub Git (0)

Escribe aquí para buscar

ESP 13:53 07/02/2021

Acceder secuencialmente a un array

Podría ser que necesitemos acceder a todos los objetos de un array, ya sea para realizar una operación matemática o añadirle un dato, podemos hacerlo mediante la utilización de ciclos iterativos como each, while, for, en el manual están todos los ejemplos, para cada uno.

Project
2
bonus
.DS_Store
08Get.rb
09Estructura.rb
010Estructura.rb
011Metodo.rb
012Rescate.rb
013For.rb
014While.rb
015Each.rb
016While.rb
017Case.rb
018For.rb
020Metodo.rb
021Var.rb
022Arrays.rb
023Arraypush.rb
arrayBasico.rb
ciclosArrays.rb
ciclosWhileArrays.rb
iteracionesSobreArrays.rb
rangos.rb
tallerSesion2.pdf
programming-ruby.pdf
rubyinstaller-devkit-2.7.2-1-x64.exe

holamundo.rb | 09Math2.rb | iteracionesSobre... | ciclosWhileArray... | ciclosArrays.rb | arrayBasico.rb | 017Case.rb | 018For.rb | 010Estructura.rb

```ruby
titulo1 = "Módulo introducción a Ruby"
titulo2 = "2021"
titulo3 = "CHILE"



puts titulo1
puts titulo2
puts titulo3

puts "Iteraciones sobre arrays"

miArray = [1, "arbol", "casa", 4, 5]

puts "Itera sobre  todos los elementos del array\n"

miArray.each { |a| print a, "\n " }#itera sobre todos los elementos del array

puts "Itera en sentido inverso\n"
miArray.reverse_each { |a| print "#{a} " } #hace una lectura reversa de los elementos del array

```

Unidad 4

4.2) Operaciones con múltiples arrays

Temas: Álgebra con arrays (Operaciones de unión, intersección, concatenación y diferencia). Matrices. Manejo básico de archivos. Abrir y leer un archivo con Ruby. Formato CSV. Guardar datos en un archivo con Ruby.

Operaciones con múltples arrays

Álgebra con arrays (Operaciones diferencia, intersección y unión)

Ya habíamos realizado algunas operaciones matemáticas, ahora veremos operaciones con conjuntos, diferencia, intersección y unión, es especialmente útil si vamos a manejar muchos conjuntos de datos o arrays.

En el siguiente ejemplo hemos creado 3 arrays que tienen elementos de tipo númerico, nos interesa conocer su diferencia, intersección y unión.

Diferencia

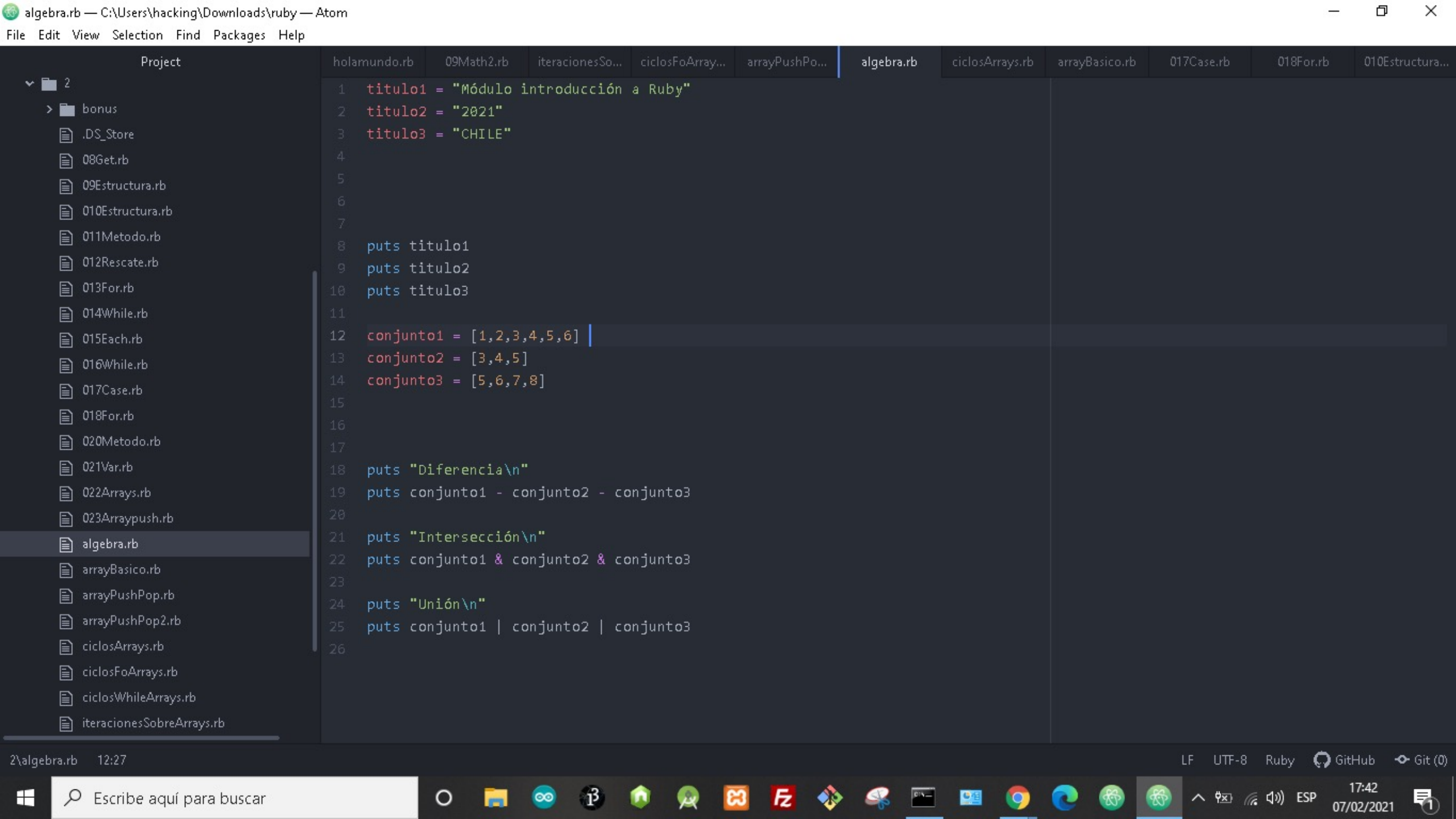
```
conjunto1 - conjunto2 - conjunto3
```

Intersección

```
conjunto1 & conjunto2 & conjunto3
```

Unión

```
conjunto1 | conjunto2 | conjunto3
```


algebra.rb — C:\Users\hacking\Downloads\ruby — Atom
File Edit View Selection Find Packages Help
Project
2
bonus
.DS_Store
08Get.rb
09Estructura.rb
010Estructura.rb
011Metodo.rb
012Rescate.rb
013For.rb
014While.rb
015Each.rb
016While.rb
017Case.rb
018For.rb
020Metodo.rb
021Var.rb
022Arrays.rb
023Arraypush.rb
algebra.rb
arrayBasico.rb
arrayPushPop.rb
arrayPushPop2.rb
ciclosArrays.rb
ciclosFoArrays.rb
ciclosWhileArrays.rb
iteracionesSobreArrays.rb
holamundo.rb
09Math2.rb
iteracionesSo...
ciclosFoArray...
arrayPushPo...
algebra.rb
ciclosArrays.rb
arrayBasico.rb
017Case.rb
018For.rb
010Estructura...
1 titulo1 = "Módulo introducción a Ruby"
2 titulo2 = "2021"
3 titulo3 = "CHILE"
4
5
6
7
8 puts titulo1
9 puts titulo2
10 puts titulo3
11
12 conjunto1 = [1,2,3,4,5,6]
13 conjunto2 = [3,4,5]
14 conjunto3 = [5,6,7,8]
15
16
17
18 puts "Diferencia\n"
19 puts conjunto1 - conjunto2 - conjunto3
20
21 puts "Intersección\n"
22 puts conjunto1 & conjunto2 & conjunto3
23
24 puts "Unión\n"
25 puts conjunto1 | conjunto2 | conjunto3
26
2\algebra.rb 12:27
LF UTF-8 Ruby GitHub Git (0)
Escribe aquí para buscar
ESP
17:42
07/02/2021

Matrices

	Columna 0	Columna 1	Columna 2
Fila 0	10	20	30
Fila 1	40	50	60
Fila 2	70	80	90

Una matriz, al igual que un array es bidimensional, y puede contener una gran cantidad de datos. En el código de ejemplo tendremos la siguiente matriz

Instalación gema matrix

`gem install matrix` y presionamos ENTER

matriz2.rb — C:\Users\hacking\Downloads\ruby — Atom

File Edit View Selection Find Packages Help

Project

- 010Estructura.rb
- 011Metodo.rb
- 012Rescate.rb
- 013For.rb
- 014While.rb
- 015Each.rb
- 016While.rb
- 017Case.rb
- 018For.rb
- 020Metodo.rb
- 021Var.rb
- 022Arrays.rb
- 023Arraypush.rb
- algebra.rb
- arrayBasico.rb
- arrayPushPop.rb
- arrayPushPop2.rb
- ciclosArrays.rb
- ciclosFoArrays.rb
- ciclosWhileArrays.rb
- iteracionesSobreArrays.rb
- matriz.rb
- matriz2.rb
- rangos.rb
- tallerSesion2.pdf

holamundo.rb | 09Math2.rb | iteracionesSo... | ciclosFoArray... | arrayPushPo... | matriz2.rb | ciclosArrays.rb | arrayBasico.rb | 017Case.rb | 018For.rb | 010Estructura...

```ruby
require 'matrix'

titulo1 = "Módulo introducción a Ruby"
titulo2 = "2021"
titulo3 = "CHILE"



puts titulo1
puts titulo2
puts titulo3

conjunto1 = Matrix[[10,20,30],[40,50,60],[70,80,90]]
conjunto2 = Matrix[[10,20,30],[40,50,60],[70,80,90]]

puts "Operación\nSumamos las matrices"

puts conjunto1+conjunto2 #suamrá 2 matrices

puts conjunto1[0,1] #imprimirá 20 porque es el valor del primer array en posición 1
puts conjunto1[1,1] #imprimirá 50 porque es el valor del segundo array en posición 1

```

2\matriz2.rb 19:44 LF UTF-8 Ruby GitHub Git (0)

Escribe aquí para buscar

ESP 18:28 07/02/2021

Manejo básico de archivos

Podemos interactuar con diferentes contenidos con Ruby, mediante, por ejemplo, la gema docx, podemos abrir y leer documentos, guardar contenido. Para eso instalaremos la gema docx y abriremos un archivo .docx que tendremos en la misma carpeta del script.

```
gem install docx
```

046_docx_html.rb — C:\Users\hacking\Downloads\ruby\6 — Atom

File Edit View Selection Find Packages Help

Project

6

- .DS_Store
- ~$llerSesion5.docx
- 042_gruff.rb
- 043_area.rb
- 044_dots.rb
- 045_docx.rb
- 046_docx_html.rb
- 047_docx_tabla_simple.rb
- 048_docx_tabla_compleja.rb
- 049_docx_tabla_html.rb
- 049_w_docx.rb
- 049_xls.rb
- 050_xls.rb
- 051_xls.rb
- 052_xls_size.rb
- 053rest_get_mi.rb
- prueba.docx
- prueba.xls
- tallerSesion6.pdf
- test.docx

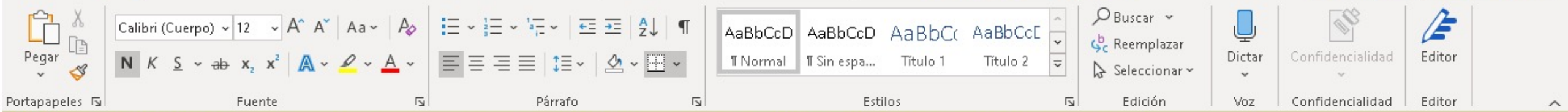
046_docx_html.rb

```ruby

titulo1 = "Curso Ruby "
titulo2 = "2021"
titulo3 = "CHILE"

puts titulo1
puts titulo2
puts titulo3

require 'docx'

doc = Docx::Document.open('prueba.docx')
doc.paragraphs.each do |lectura|
  puts lectura
end

```

046_docx_html.rb 14:15 LF UTF-8 Ruby GitHub Git (0)

Escribe aquí para buscar

ESP 3:15 14/02/2021



Navegación

Buscar en documento

Títulos Páginas Resultados

DOCX ES LA GEMA PARA TRABAJAR...

DOCX ES LA GEMA PARA TRABAJAR CON .DOCX – CURSO RUBY

Esta es una prueba para el curso de Ruby, estamos leyendo el documento.

Podemos establecer que está escrito en cada párrafo.

Estamos leyendo los párrafos del texto.

Y un poco más...

Símbolo del sistema

```
C:\Users\hacking\Downloads\ruby\6>ruby 046_docx_html.rb
Curso Ruby
2021
CHILE
DOCX ES LA GEMA PARA TRABAJAR CON .DOCX - CURSO RUBY
Esta es una prueba para el curso de Ruby, estamos leyendo el documento.

Podemos  establecer  que está escrito en cada párrafo.

Estamos leyendo los párrafos del texto.

Y un poco más…


C:\Users\hacking\Downloads\ruby\6>
```

Archivos.csv

Los archivos CSV son archivos que tienen sus datos separados por coma (comma separated values).

Para poder trabajar con ellos necesitamos la gema csv

```
gem install csv
```

Para este ejemplo utilizaremos el sitio https://www.mockaroo.com/ que permite generar datos de ficción para realizar ejercicios.

Mockaroo - Random Data Gener

mockaroo.com

mockaroo realistic data generator

PRICING SIGN IN

Need some mock data to test your app? Mockaroo lets you generate up to 1,000 rows of realistic test data in CSV, JSON, SQL, and Excel formats.

Download data using your browser or sign in and create your own Mock APIs.

Need more data? Plans start at just $50/year. Mockaroo is also available as a docker image that you can deploy in your own private cloud.

Field Name	Type	Options
id	Row Number	blank: 0 %
first_name	First Name	blank: 0 %
last_name	Last Name	blank: 0 %
email	Email Address	blank: 0 %

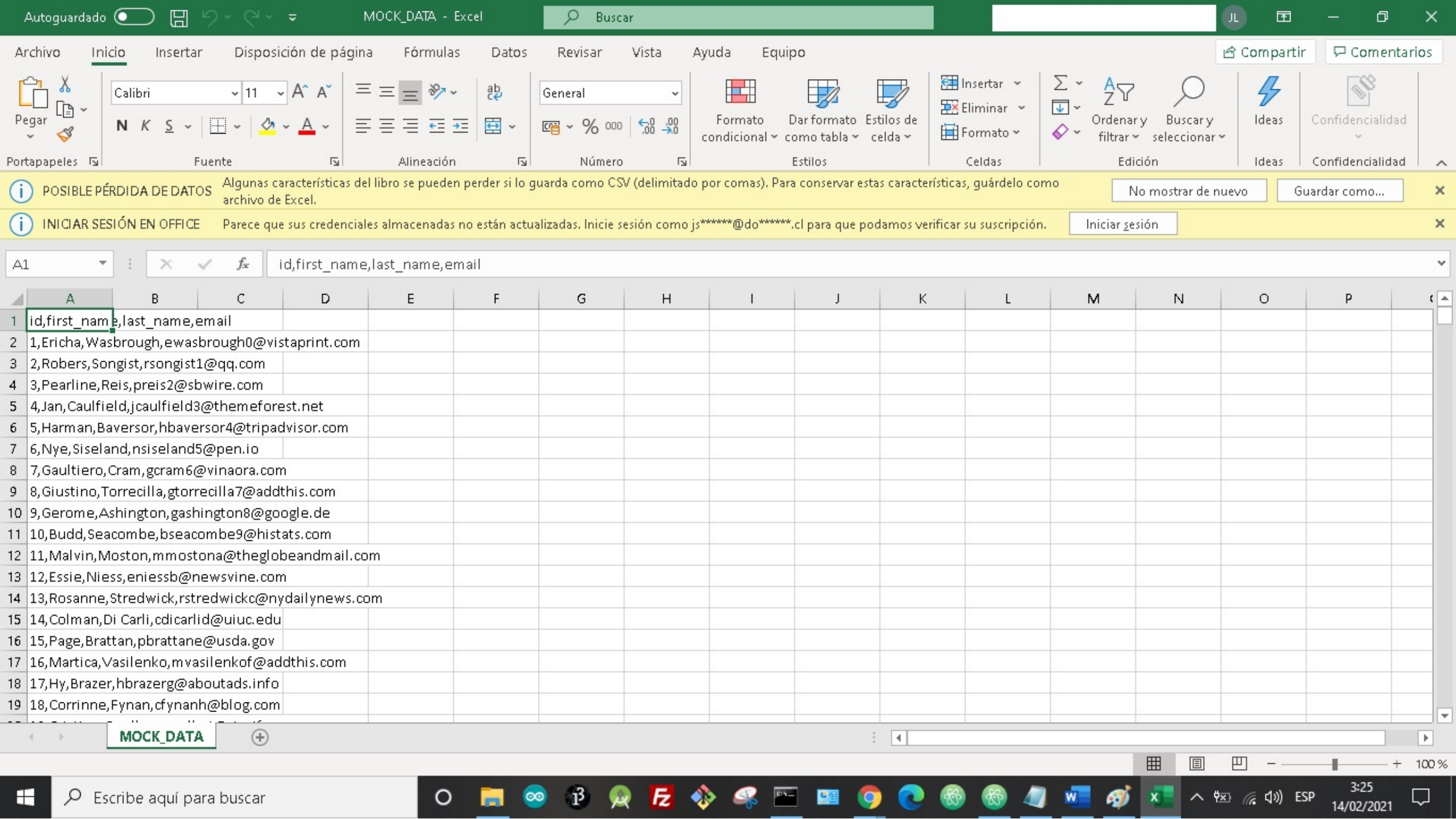
Add another field

Rows: 1000 Format: CSV Line Ending: Unix (LF) Include: header BOM

Download Data Preview More Want to save this for later? Sign up for free.

Escribe aquí para buscar

ESP 3:22 14/02/2021



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	id,first_name,last_name,email															
2	1,Ericha,Wasbrough,ewasbrough0@vistaprint.com															
3	2,Robers,Songist,rsongist1@qq.com															
4	3,Pearline,Reis,preis2@sbwire.com															
5	4,Jan,Caulfield,jcaulfield3@themeforest.net															
6	5,Harman,Baversor,hbaversor4@tripadvisor.com															
7	6,Nye,Siseland,nsiseland5@pen.io															
8	7,Gaultiero,Cram,gcram6@vinaora.com															
9	8,Giustino,Torrecilla,gtorrecilla7@addthis.com															
10	9,Gerome,Ashington,gashington8@google.de															
11	10,Budd,Seacombe,bseacombe9@histats.com															
12	11,Malvin,Moston,mmostona@theglobeandmail.com															
13	12,Essie,Niess,eniessb@newsvine.com															
14	13,Rosanne,Stredwick,rstredwickc@nydailynews.com															
15	14,Colman,Di Carli,cdicarlid@uiuc.edu															
16	15,Page,Brattan,pbrattane@usda.gov															
17	16,Martica,Vasilenko,mvasilenkof@addthis.com															
18	17,Hy,Brazer,hbrazerg@aboutads.info															
19	18,Corrinne,Fynan,cfynanh@blog.com															

054csv.rb — C:\Users\hacking\Downloads\ruby\6 — Atom
File Edit View Selection Find Packages Help
Project
6
.DS_Store
~$llerSesion5.docx
042_gruff.rb
043_area.rb
044_dots.rb
045_docx.rb
046_docx_html.rb
047_docx_tabla_simple.rb
048_docx_tabla_compleja.rb
049_docx_tabla_html.rb
049_w_docx.rb
049_xls.rb
050_xls.rb
051_xls.rb
052_xls_size.rb
053rest_get_mi.rb
054csv.rb
MOCK_DATA.csv
prueba.docx
prueba.xls
tallerSesion6.pdf
test.docx
046_docx_html.rb
054csv.rb
MOCK_DATA.csv
1
2 titulo1 = "Curso Ruby"
3 titulo2 = "2021"
4 titulo3 = "CHILE"
5
6 puts titulo1
7 puts titulo2
8 puts titulo3
9
10 require 'csv'
11 puts CSV.read("MOCK_DATA.csv")
12
054csv.rb 12:1
LF UTF-8 Ruby GitHub Git (0)
Escribe aquí para buscar
ESP
3:33
14/02/2021

Guardar un archivo en Ruby

Para el siguiente debemos tener un archivo docx “prueba_editado.docx” en nuestra carpeta del script, con el siguiente código podremos escribir y guardar datos en ese archivo.

Project — C:\Users\hacking\Downloads\ruby — Atom

File Edit View Selection Find Packages Help

Project

ruby
- 1
- 2
- 6
  - .DS_Store
  - ~$llerSesion5.docx
  - 042_gruff.rb
  - 043_area.rb
  - 044_dots.rb
  - 045_docx.rb
  - 046_docx_html.rb
  - 047_docx_tabla_simple.rb
  - 048_docx_tabla_compleja.rb
  - 049_docx_tabla_html.rb
  - 049_w_docx.rb
  - 049_xls.rb
  - 050_xls.rb
  - 051_xls.rb
  - 052_xls_size.rb
  - 053rest_get_mi.rb
  - 054csv.rb
  - MOCK_DATA.csv
  - prueba.docx
  - prueba.xls
  - tallerSesion6.pdf

pais.json | 049_w_docx.rb

```ruby
require 'docx'

doc = Docx::Document.open('test.docx')

doc.bookmarks['test1'].insert_text_after(", ha salido todo bien")
#Hay que tener los marcadores funcionando tengo dos: test y prueba_dos

doc.bookmarks['test2'].insert_multiple_lines(['Acá agregaré', 'más ', 'palabras'])
#El ejemplo de la gema esta malo, fijate que para varias lineas no lleva el after



doc.save('prueba_editada.docx')

```

6\049_w_docx.rb 1:1 CRLF UTF-8 Ruby GitHub Git

Escribe aquí para buscar

ESP 18:22 14/02/2021

Material complementario de la unidad

Link a video relacionado

1. https://www.youtube.com/watch?v=10KGLWPf9mc

Link a lectura complementaria

1. https://ruby-doc.org/core-2.7.0/Array.html

Link a investigación relacionada

1. https://github.com/ruby/matrix