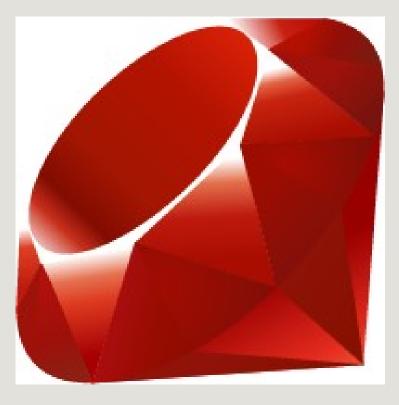
# Academia Ruby Curso Piloto dia 2.

# !Repaso!

#### ¿Qué se vio la clase pasada?



-Variables

-Condicionales

-Iteradores

- Imprimir y pedir

-Arreglos

# Imprimir, Pedir y asignar variables.

Puts "Esto saldra en una linea separada"

Print "En cambio esto saldra continuo a lo anterior"

P"Esto saldra con literal a como en codigo"

Esto saldra en una linea separada En cambio esto saldra continuo a lo siguiente, "Esto saldra con

literal a como en codigo\n"

```
Puts"introduzca una letra"
a = gets
Puts"introduzca una letra, otra vez"
b = gets.chomp
b ||= 10
```

#### Introduzca una letra

A

=> A\n

Introduzca una letra, otra vez

B

=>B

### Condicionales

```
a=9
b=10
c=a*b if a>10
B=11 unless c=90
```

```
Introduzca una letra

A
=> A\n
Introduzca una letra, otra vez
B
=> D
```

```
a=nil
b=0
if a
  print "successfully this is printed because a is
nil therefore nil is equal to False"
elsif b
  print "the same case as before is not ,because b is true!"
end
```

Successfully this is printed because a is nil therefore nil es equal to false

Edad=21
persona = (13...19).include?(edad) ? "Adolescente" : "Adulto"
puts persona

=> Adulto

# Iteradores

```
i = 0
num = 5
while i < num do
  puts("Itero i = #{i}")
  i +=1
end</pre>
```

```
Itero i = 0
Itero i = 1
Itero i = 2
Itero i = 3
Itero i = 4
```

```
z = 10
begin
a=z+1
z+=1
puts a
end while a <15
```

11

12

13

14

15

```
array = [1,2,3,4,5]
array.each do |i|
puts i
end
```

```
1234
```

### Clase #2

### Arrays

```
newarr = [45, 23, 1, 90]
puts newarr.sort
puts newarr.length
puts newarr.first
puts newarr.last
puts newarr[1, 3]
puts newarr[1..2]
Puts newarr.include?(3)
newarr.unshift(10)
Newarr.delete_at(0)
 1,23,45,90
```

```
1,23,45,90
4
45
90
23,1,90
23,1
False
=>[10,45,23,1,90]
=>[45,23,1,90]
```

#### Hashes

```
Animales = {'perro' => 'canidae', 'Gato' => 'felidae', 'Burro' => 'equida e', 'Gecko' => 'gekkota'}

puts Animales.length

puts Animales['perro'] # 'canine'

Animales['mono']= 'primate'

Puts Animales
```

Canidae

{'perro' => 'canidae', 'Gato' => 'felidae', 'Burro' => 'equidae',
'Gecko' => 'gekkota', 'Mono' => 'primate'}

## Bloques

```
a = lambda{puts"Hola soy un bloque de codigo"}
a.Call
Puts "ingrese la cantidad de parcsecs a convertir a kilometros"
x=gets.to_i
ecuacion_de_parsecs= lambda{return 30856776*x}
Puts ecuacion_de_parsecs.call
```

#### Hola soy un bloque de codigo

Ingrese la cantidad de parcsecs a convertir a kilometros

=>1

30856776