

# UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA



(UNAN-LEÓN)  
FACULTAD DE CIENCIAS Y TECNOLOGÍA

**Carrera:**

Ingeniería en Telemática

**Componente:**

Software Como Un Servidor

**Elaborado por:**

Chelsea María Moraga Noguera.

**Profesor:**

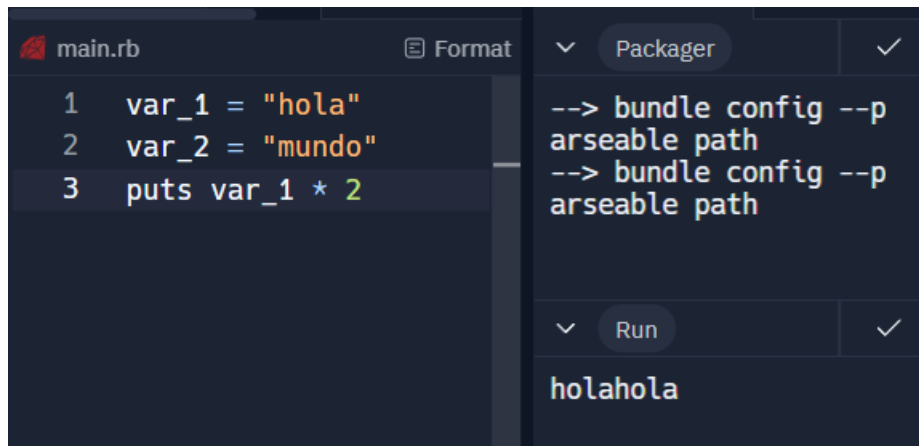
Ervin Ismael Montes Téllez

**Fecha:**

09/08/2024

*“A la Libertad por la Universidad”*

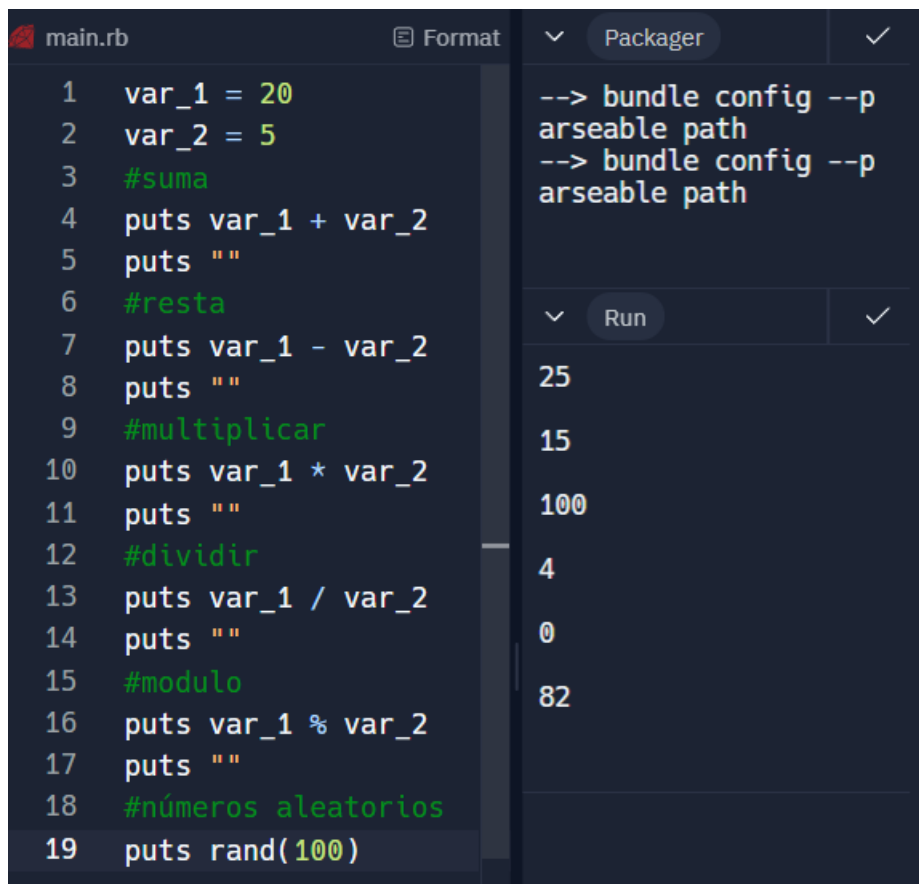
### Ejercicio 1



The screenshot shows a Ruby IDE with a file named `main.rb`. The code in the editor consists of three lines: `var_1 = "hola"`, `var_2 = "mundo"`, and `puts var_1 * 2`. The right sidebar has a 'Packager' tab with two identical entries: `--> bundle config --p arseable path`. Below this is a 'Run' button and the output `holahola`.

```
main.rb  Format  Packager  ✓  
1  var_1 = "hola"  
2  var_2 = "mundo"  
3  puts var_1 * 2  
  
--> bundle config --p arseable path  
--> bundle config --p arseable path  
  
Run  ✓  
holahola
```

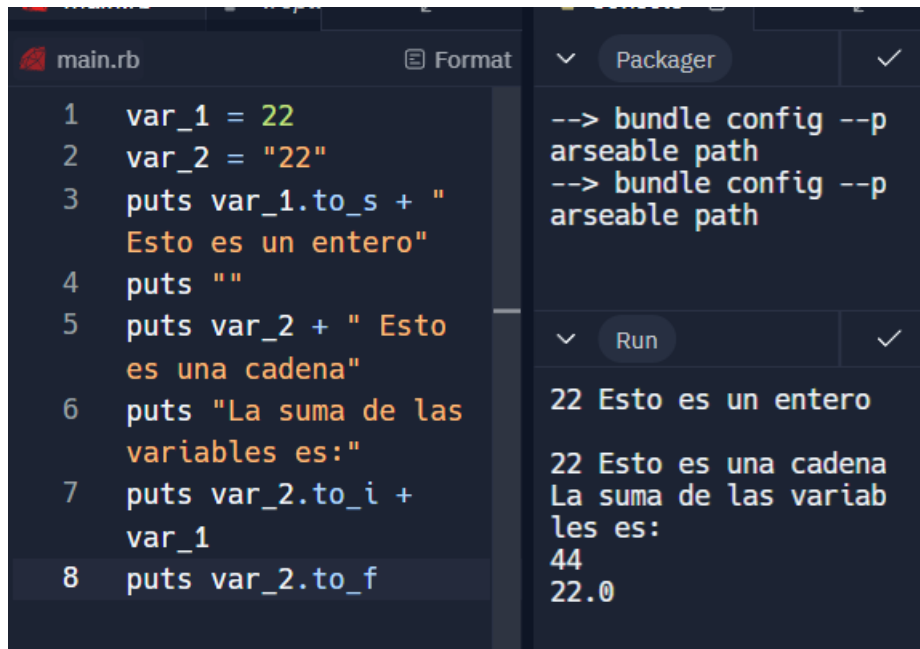
### Ejercicio 2



The screenshot shows a Ruby IDE with a file named `main.rb`. The code in the editor consists of 19 lines: `var_1 = 20`, `var_2 = 5`, `#suma`, `puts var_1 + var_2`, `puts ""`, `#resta`, `puts var_1 - var_2`, `puts ""`, `#multiplicar`, `puts var_1 * var_2`, `puts ""`, `#dividir`, `puts var_1 / var_2`, `puts ""`, `#modulo`, `puts var_1 % var_2`, `puts ""`, `#números aleatorios`, and `puts rand(100)`. The right sidebar has a 'Packager' tab with two identical entries: `--> bundle config --p arseable path`. Below this is a 'Run' button and the output: `25`, `15`, `100`, `4`, `0`, and `82`.

```
main.rb  Format  Packager  ✓  
1  var_1 = 20  
2  var_2 = 5  
3  #suma  
4  puts var_1 + var_2  
5  puts ""  
6  #resta  
7  puts var_1 - var_2  
8  puts ""  
9  #multiplicar  
10 puts var_1 * var_2  
11 puts ""  
12 #dividir  
13 puts var_1 / var_2  
14 puts ""  
15 #modulo  
16 puts var_1 % var_2  
17 puts ""  
18 #números aleatorios  
19 puts rand(100)  
  
--> bundle config --p arseable path  
--> bundle config --p arseable path  
  
Run  ✓  
25  
15  
100  
4  
0  
82
```

### Ejercicio 3



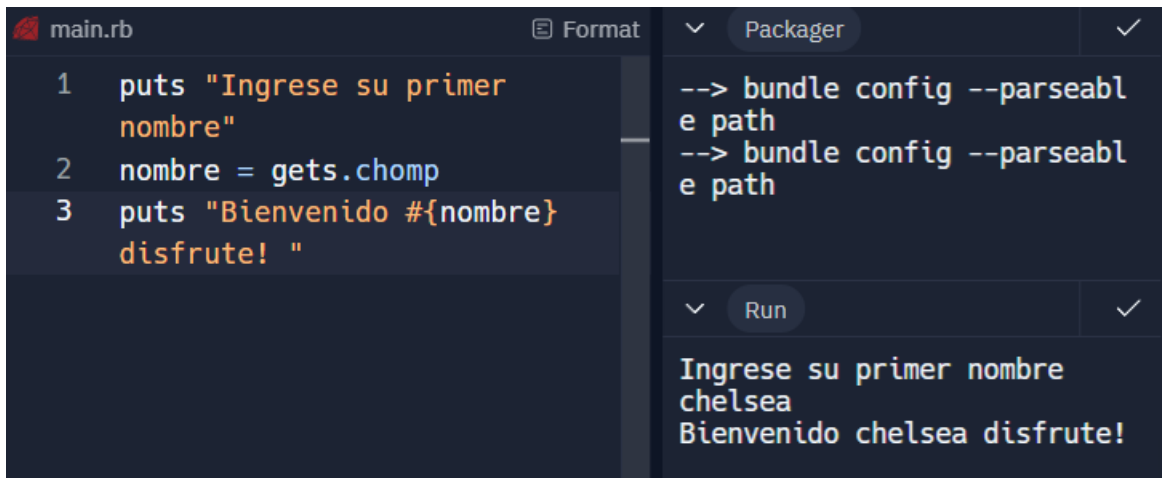
The screenshot shows a Ruby IDE with a file named `main.rb`. The code in the editor is as follows:

```
1 var_1 = 22
2 var_2 = "22"
3 puts var_1.to_s + "
  Esto es un entero"
4 puts ""
5 puts var_2 + " Esto
  es una cadena"
6 puts "La suma de las
  variables es:"
7 puts var_2.to_i +
  var_1
8 puts var_2.to_f
```

On the right side, there are two panels. The top panel, labeled 'Packager', shows the command `--> bundle config --p arseable path` twice. The bottom panel, labeled 'Run', shows the output of the program:

```
22 Esto es un entero
22 Esto es una cadena
La suma de las variab
les es:
44
22.0
```

### Ejercicio 4



The screenshot shows a Ruby IDE with a file named `main.rb`. The code in the editor is as follows:

```
1 puts "Ingrese su primer
  nombre"
2 nombre = gets.chomp
3 puts "Bienvenido #{nombre}
  disfrute! "
```

On the right side, there are two panels. The top panel, labeled 'Packager', shows the command `--> bundle config --parseabl e path` twice. The bottom panel, labeled 'Run', shows the output of the program:

```
Ingrese su primer nombre
chelsea
Bienvenido chelsea disfrute!
```

## Ejercicio 5

```
main.rb  Format  Packager  336ms on 16:56:15, 08/09 ✓

1 puts "Ingrese su nombre"
2 nombre = gets.chomp
3 #Imprime el nombre ingresado
4 puts "Nombre => " + nombre
5 # convierte al revés el nombre
6 puts "Método reverse => " +
  nombre.reverse
7 #Mayúscula
8 puts "Método upcase => " +
  nombre.upcase
9 #Minúscula
10 puts "Método downcase => " +
  nombre.downcase
11 #Intercambia las minúsculas por
  mayúscula (viceversa)
12 puts "Método swapcase => " +
  nombre.swapcase
13 #cambia el primer carácter a
  mayúscula
14 puts "Método capitalize => " +
  nombre.capitalize
15 #devuelve el tamaño del string
  ingresado
16 puts "Método length => " +
  nombre.length.to_s
17
```

```
--> bundle config --parseable pat
h
--> bundle config --parseable pat
h

Run  4s on 16:56:16, 08/09 ✓

Ingrese su nombre
chelsea
Nombre => chelsea
Método reverse => aeslehc
Método upcase => CHELSEA
Método downcase => chelsea
Método swapcase => CHELSEA
Método capitalize => Chelsea
Método length => 7
```

## Ejercicio 6

```
main.rb  Format  Packager  288ms on 16:57:12, 08/09 ✓

1 iterador = " "
2 while iterador.downcase != "s"
3 puts "Ingrese un nombre"
4 nombre = gets.chomp
5 tamaño = nombre.length
6 if (tamaño >= 5 )
7 puts "Su nombre tiene más de 5
  caracteres"
8 else
9 puts "Su nombre tiene menos de 5
  caracteres"
10 end
11 puts "\nPara salir presione la
  letra S"
12 iterador = gets.chomp
13 end
14 puts "Ha salido del programa"
```

```
--> bundle config --parseable pat
h
--> bundle config --parseable pat
h

Run  21s on 16:57:12, 08/09 ✓

Ingrese un nombre
chelsea
Su nombre tiene más de 5 caracter
es

Para salir presione la letra S
s
Ha salido del programa
```