

---

# Ruby Monstas



---

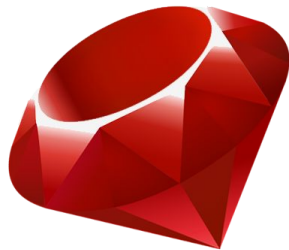
## Session 4

---

# Agenda

---

- Control structures
- Exercises



---

# Control structures

---

## Introduction

---

# If

---

Sometimes we want to execute code depending on the situation.

**If** a certain situation is given, we want to execute a body of code.

---

# The condition

---

```
if condition
```

```
  # the condition is true
```

```
end
```

**condition** is a **boolean**.

---

# What could be a condition?

---

`1 < 2`

`"Hello" == "World"`

`"Hello".include?("llo")`

`[4, 5, 6].include?(7)`

`{"Marion" => 7}.has_key?("Marion")`

**Anything that returns `true` or `false`!**

---

# If: example

---

```
number = 42
```

```
if number > 10
```

```
  puts "#{number} is greater than 10"
```

```
end
```

Output

42 is greater than 10

---

# If: example in IRB

---

```
irb> number = 42
```

```
=> 42
```

```
irb> if number > 10
```

```
irb>   puts "#{number} is greater than 10"
```

```
irb> end
```

```
42 is greater than 10
```

```
=> nil
```

---



# If / else: the pattern

---

```
if condition
  # the condition is true
else
  # the condition is false
end
```

# If, else: example

---

```
number = 4
```

```
if number > 10
```

```
  puts "#{number} is greater than 10"
```

```
else
```

```
  puts "#{number} is lower or equal to 10"
```

```
end
```

Output

4 is lower or equal to 10

---

# If, else: example in IRB

---

```
irb> number = 4
```

```
=> 4
```

```
irb> if number > 10
```

```
irb>   puts "#{number} is greater than 10"
```

```
irb> else
```

```
irb>   puts "#{number} is lower or equal to 10"
```

```
irb> end
```

```
4 is lower or equal to 10
```

```
=> nil
```

---

# If / else: the pattern

---

```
if condition1
    # condition1 is true
elseif condition2
    # condition2 is true
else
    # none of the conditions are true
end
```

---

# If, elsif, else: example

---

```
season = "spring"

if season == "summer"
  puts "It's too hot!"
elsif season == "winter"
  puts "It's too cold!"
else
  puts "It's just right!"
end
```

## Output

It's just right!

---

# If, elsif, else: example in IRB

---

```
irb> season = "spring"
=> "spring"
irb> if season == "summer"
irb>   puts "It's too hot!"
irb> elsif season == "winter"
irb>   puts "It's too cold!"
irb> else
irb>   puts "It's just right!"
irb> end
It's just right!
=> nil
```

---

# Loop

---

A **loop** cycles through code until we tell it to stop with **break**.

---

# Loop: the pattern

---

```
loop do
  if condition
    break
  end
end
```

---



# Loop: Example

---

```
counter = 1
```

```
loop do
  puts counter
  counter = counter + 1
  if counter > 3
    break
  end
end
```

## Output

```
1
2
3
```

---

# Loop: Example

---

```
irb> counter = 1
```

```
=> 1
```

```
irb> loop do
```

```
  puts counter
```

```
  counter = counter + 1
```

```
  if counter > 3
```

```
    break
```

```
  end
```

```
end
```

```
1
```

```
2
```

```
3
```

```
=> nil
```

---

# Each: the pattern

---

```
array = []  
array.each do |element|  
  # do stuff with element  
end
```

# Each: example

---

```
names = ["Alice", "Bob", "Eve"]
```

```
names.each do |name|  
  puts name  
end
```

## Output

Alice

Bob

Eve

```
["Alice", "Bob", "Eve"]
```

---

# Each: example

---

```
irb> names = ["Alice", "Bob", "Eve"]
```

```
=> ["Alice", "Bob", "Eve"]
```

```
irb> names.each do |name|
```

```
  puts name
```

```
end
```

Alice

Bob

Eve

```
=> ["Alice", "Bob", "Eve"]
```

---

# Coming up

---

- Boolean operators
  - Methods
-

# Demo

---

Todo Script

---

**Tip: Work in pairs**

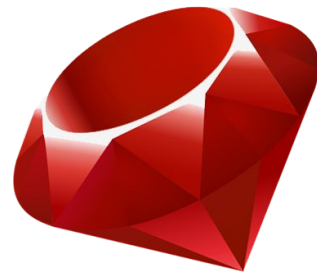
---

---



---

# Time to practice



---

Let's get to it!

---