

# Ruby: The Big Picture

---

## DISCOVERING RUBY



**Paolo Perrotta**

FREELANCE DEVELOPER

@nusco

# Version Check



**This course applies to:**

- Ruby 2.x
- Ruby 3.x

# Overview

## **Discovering Ruby**

- Let's write some Ruby!

## **Investigating Ruby Objects**

- Ruby and object-oriented programming

## **Unlocking Ruby's Hidden Magic**

- Ruby's cool metaprogramming features

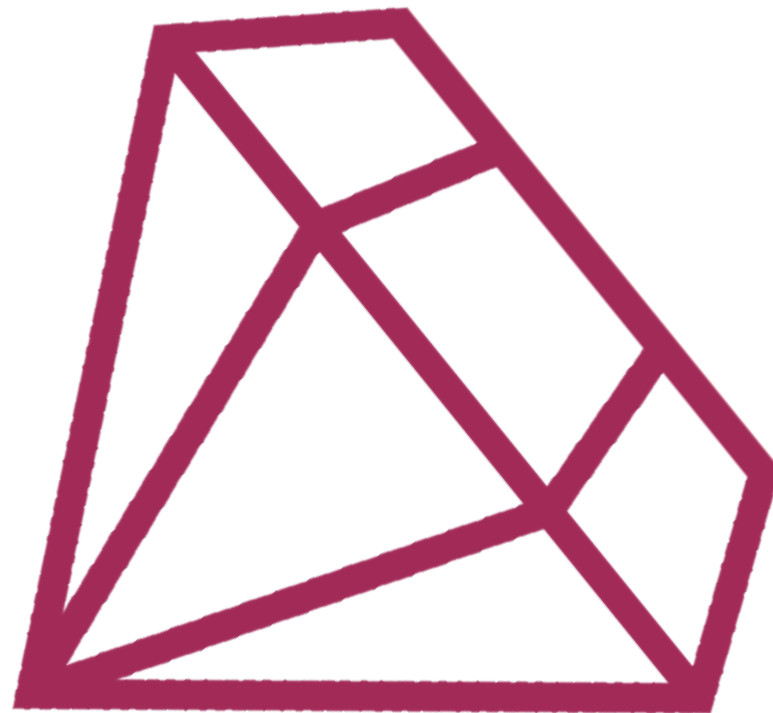
## **Exploring Ruby's Ecosystem**

- Interpreters, libraries and community

## **Understanding Ruby's Pros and Cons**

- Strong and weak points

Let's Dive In!



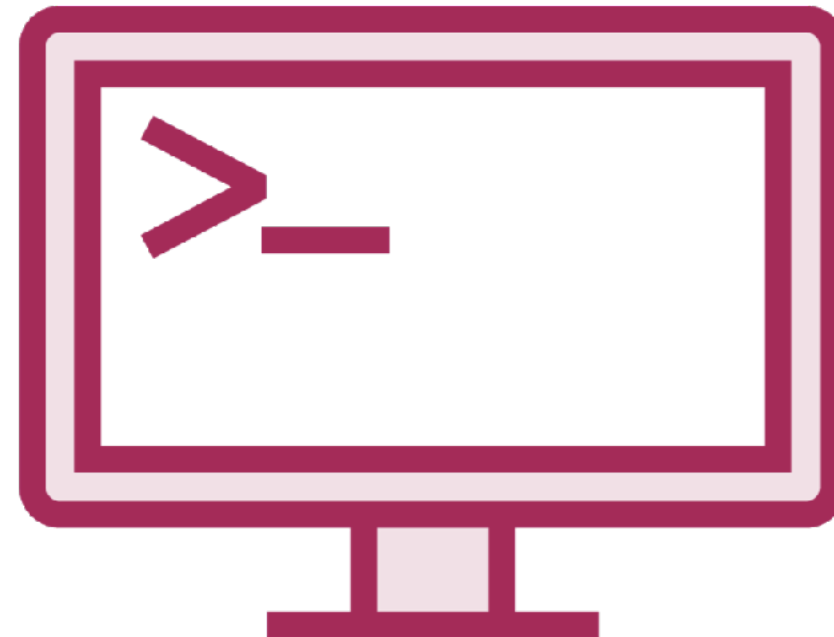
# Running Ruby Code

---

# Running Ruby Code



As a program



In a shell

# Public Service Announcements

---

# Public Service Announcements

## Ignore the details

No stress—I just want to show you what Ruby looks like

## You can follow along

Type Ruby statements in your own interpreter, if you wish



# A “Scripting Language”

---

# Comparing Languages

## Java (static, compiled)

```
int doSomething(String a, int b, int c) {  
    int result = 0;  
    if (a.equals("mod")) {  
        result = b % c;  
    }  
    return result;  
}
```

## JavaScript (dynamic, interpreted)

```
function doSomething(a, b, c) {  
    result = 0;  
    if (a === "mod") {  
        result = b % c;  
    }  
    return result;  
}
```

## Ruby (dynamic, interpreted)

```
def do_something(a, b, c)  
    result = 0  
    if a == "mod"  
        result = b % c  
    end  
    return result  
end
```

Ruby is dynamic and interpreted  
(a “scripting language”).

# Ruby Is a Scripting Language

**Like...**

**...JavaScript, Python**

**Unlike...**

**...Java, C#, C**

# A Problem with Jazz

---

albums.csv X



```
1  Title,Author,Genre,Year
2  Revolver,The Beatles,Pop,1966
3  Time Out,Dave Brubeck,Jazz,1959
4  Exile on Main St.,The Rolling Stones,Rock,1972
5  Kind of Blue,Miles Davis,Jazz,1959
6  The Wall,Pink Floyd,Rock,1979
7  Thriller,Michael Jackson,Pop,1982
8  Blue Train,John Coltrane,Jazz,1958
9  Led Zeppelin II,Led Zeppelin,Rock,1969
10 A Love Supreme,John Coltrane,Jazz,1964
11 The Epic,Kamasi Washington,Jazz,2015
12 |
```

# Loading CSV Data

---

```
1 Title,Author,Genre,Year
2 Revolver,The Beatles,Pop,1966
3 Time Out,Dave Brubeck,Jazz,1959
4 Exile on Main St.,The Rolling Stones,Rock,1972
5 Kind of Blue,Miles Davis,Jazz,1959
6 The Wall,Pink Floyd,Rock,1979
7 Thriller,Michael Jackson,Pop,1982
8 Blue Train,John Coltrane,Jazz,1958
9 Led Zeppelin II,Led Zeppelin,Rock,1969
10 A Love Supreme,John Coltrane,Jazz,1964
11 The Epic,Kamasi Washington,Jazz,2015
12 |
```

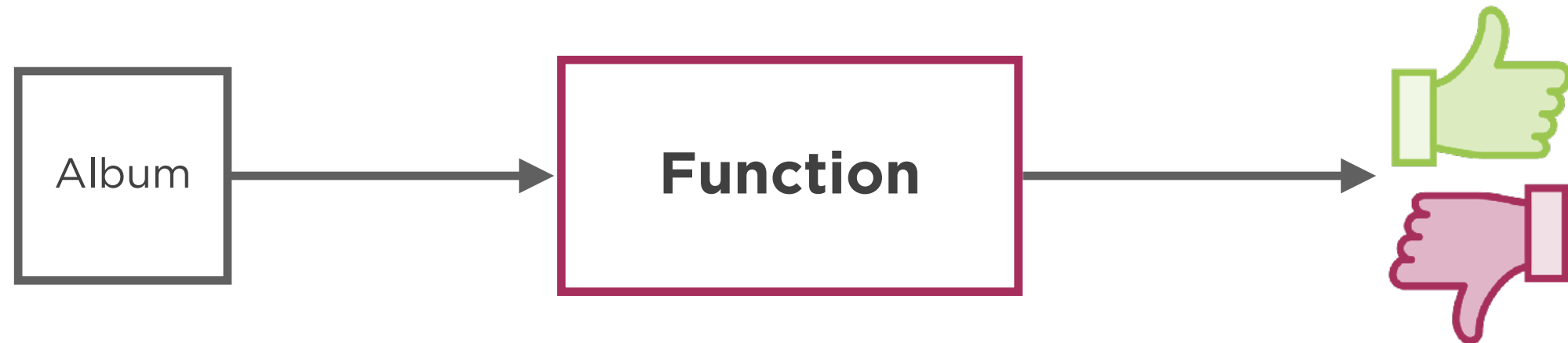
```
>pry
[1] pry(main)> require "csv"
=> true
[2] pry(main)> albums = CSV.read("albums.csv")
=> [["Title", "Author", "Genre", "Year"],
    ["Revolver", "The Beatles", "Pop", "1966"],
    ["Time Out", "Dave Brubeck", "Jazz", "1959"],
    ["Exile on Main St.", "The Rolling Stones", "Rock", "1972"],
    ["Kind of Blue", "Miles Davis", "Jazz", "1959"],
    ["The Wall", "Pink Floyd", "Rock", "1979"],
    ["Thriller", "Michael Jackson", "Pop", "1982"],
    ["Blue Train", "John Coltrane", "Jazz", "1958"],
    ["Led Zeppelin II", "Led Zeppelin", "Rock", "1969"],
    ["A Love Supreme", "John Coltrane", "Jazz", "1964"],
    ["The Epic", "Kamasi Washington", "Jazz", "2015"]]
[3] pry(main)> albums[1]
=> ["Revolver", "The Beatles", "Pop", "1966"]
[4] pry(main)> albums[0]
=> ["Title", "Author", "Genre", "Year"]
[5] pry(main)> albums[2..4]
=> [["Time Out", "Dave Brubeck", "Jazz", "1959"],
    ["Exile on Main St.", "The Rolling Stones", "Rock", "1972"],
    ["Kind of Blue", "Miles Davis", "Jazz", "1959"]]
[6] pry(main)> █
```



# Writing a Function

---

# A Function to Recognize Classic Jazz



# A Ruby Function

```
def function_name(argument1, argument2, argument3)  
  return ... # some code to compute the result  
end
```

# A Ruby Function

```
def classic_jazz?(album)
  return (album[2] == "Jazz" and album[3].to_i.between?(1950, 1959))
end
```

# A Ruby Function

```
def classic_jazz?(album)
  album[2] == "Jazz" and album[3].to_i.between?(1950, 1959)
end
```

# A Different Way of Looping

---

# *A for* Loop

```
selected_albums = []
```

```
for i in 0...albums.size do  
  if classic_jazz?(albums[i])  
    selected_albums << albums[i]  
  end  
end
```



Ruby gives you more than one  
way to do the same thing.



# Introducing Blocks

```
albums.select { |album| album[2] == "Jazz" and album[3].to_i.between?(1950, 1959) }
```

# Completing the Program

---

```
1  GENRE = 2
2  YEAR = 3
3
4  require "csv"
5  CSV.read("albums.csv")[1..]
6    .select {|album|
7      album[GENRE] == "Jazz" and album[YEAR].to_i.between?(1950, 1959)
8    }
9    .sort_by {|album| album[YEAR].to_i }
10   .each {|album| puts album.join(',') }
11
```

Ruby is expressive.

# A Few Things We Learned

---

# Summary

**Ruby is a scripting language**

**Ruby generally gives you more than one way  
to do the same thing**

**Ruby is expressive**