Ruby Monstas

Session 18

Agenda

- Classes / Objects, Object-oriented programming Recap
- Exercises

Recap

Classes & Objects

Object-oriented programming

We know them already!

We saw them already in the past.

```
my_array = Array.new

my_hash = Hash.new
```

We know them already!

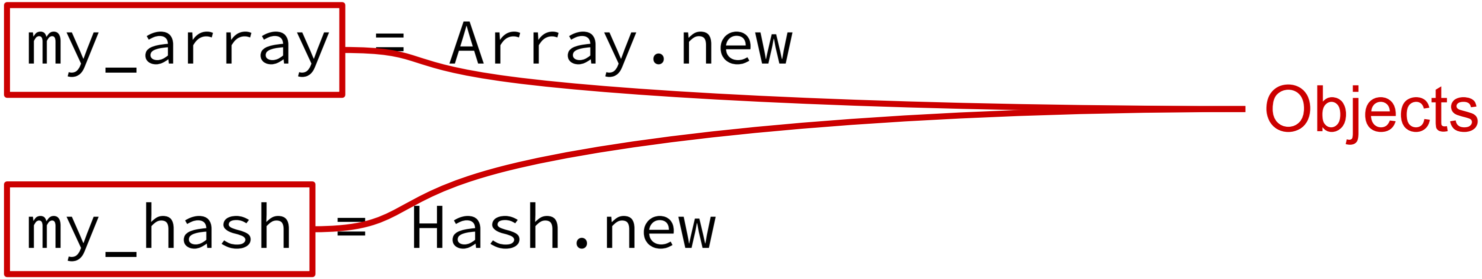
We saw them already in the past.

```
my_array = Array.new

my_hash = Hash.new
```

Classes

We know them already!

We saw them already in the past.



```
my_array = Array.new

my_hash = Hash.new
```

Objects

The class method

We can ask all our objects what class they are.

```
irb> "Hello World!".class
=> String

irb> { :name => "Ferdinand", :location =>
"Zurich" }.class
=> Hash
```

Classes vs. Objects

So what are classes and what are objects?

Classes are **ideas**, and **objects** are manifestations of these ideas, or actual **things**.

Example: Class Person

Let’s dive into an example. We have a Person class (idea), which can have many person objects (manifestations)

```
irb> lucy = Person.new("Lucy")
=> #<Person:0x007f8b8289a868>
irb> lucy.name
=> "Lucy"
```

Example: Class Person

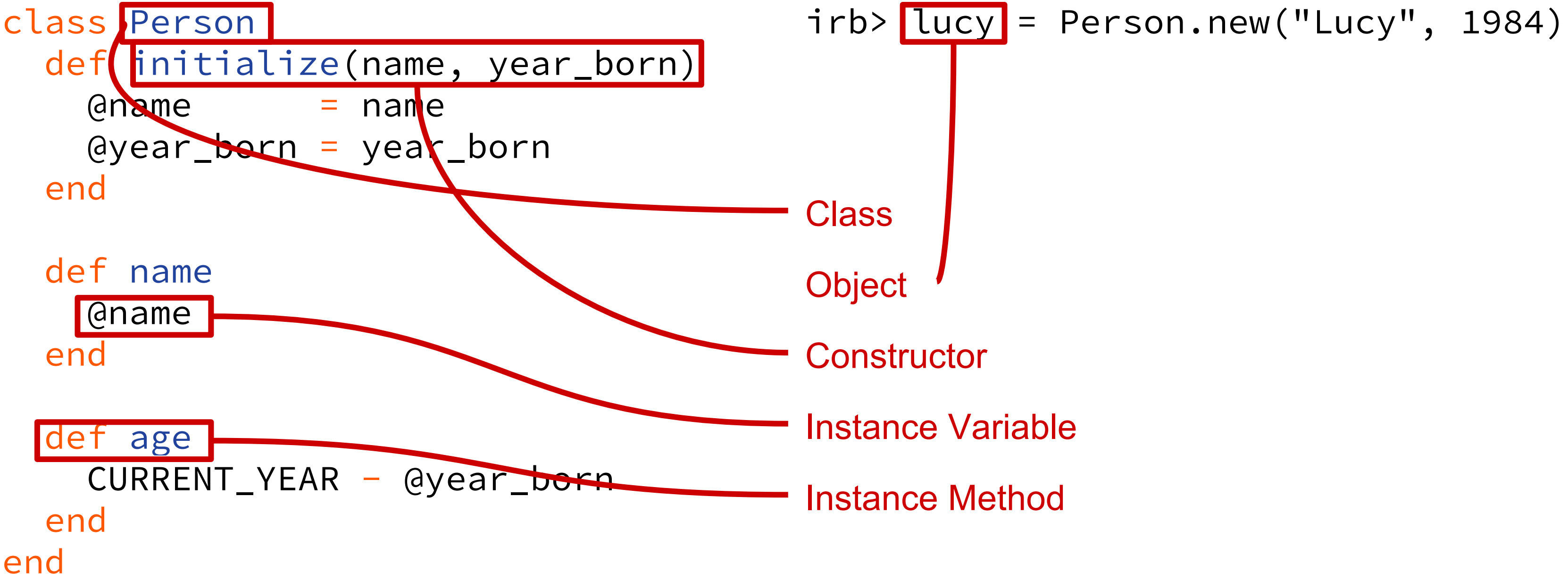
```ruby
class Person
  def initialize(name)
    @name = name
  end

  def name
    @name
  end
end
```

Example: Class Person

```ruby
class Person
  def initialize(name)
    @name = name
  end

  def name
    @name
  end
end
```

```
irb> lucy = Person.new("Lucy")
=> #<Person:0x007f8b8289a868>

irb> lucy.name
=> "Lucy"
```

Example: Class Person



```
class Person
  def initialize(name, year_born)
    @name      = name
    @year_born = year_born
  end

  def name
    @name
  end

  def age
    CURRENT_YEAR - @year_born
  end
end
```

```
irb> lucy = Person.new("Lucy", 1984)
```

Example: Class Person

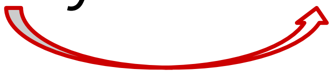
```
class Person
  def initialize(name, year_born)
    @name      = name
    @year_born = year_born
  end

  def name
    @name
  end

  def age
    CURRENT_YEAR - @year_born
  end
end
```

```
irb> lucy = Person.new("Lucy", 1984)
=> #<Person:0x007f8b8289a868>

irb> lucy.name
=> "Lucy"

irb> lucy.age
=> 31
```

Instance

Objects that are created from classes are also called **an instance**. Therefore objects are **instantiated** from a class.

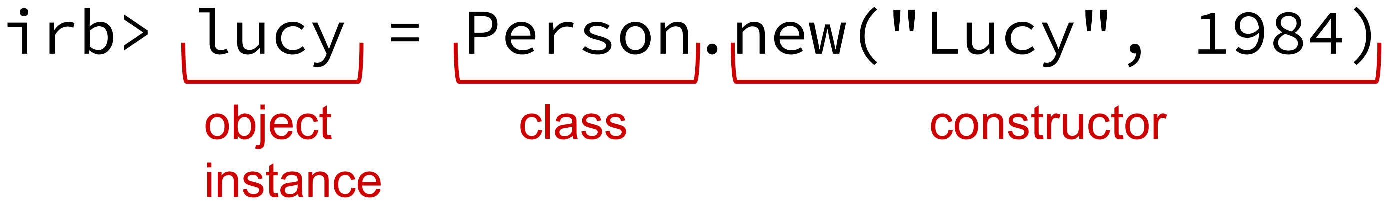
```
irb> lucy = Person.new("Lucy", 1984)
```



is an instance of

Instance

Objects that are created from classes are also called **an instance**. Therefore objects are **instantiated** from a class.



```
irb> lucy = Person.new("Lucy", 1984)
```

object instance

class

constructor

Summary

Classes are introduced with `class` keyword

Classes are ideas, objects are things

Every object knows its class

Objects offer methods which are defined in classes

Each instance (object) has “unique” data in it that it operates on

One more example: Cat

From Exercise 14:

In this exercise, we are going to implement a little program that lets you organise your feline friends. For the sake of simplicity, each of the cats that we’ll deal with has exactly three properties, no more and no less:

- A name
- The number of lives it has left (out of its 9 lives)
- The type of its coat (e.g. tabby or tuxedo)

One more example: Cat

So what would our cat class look like?

Let’s program this together!

Time to practice

Let’s get to it!