

# OBJECTS & CLASSES

with tic tac toe, arrays and hashes

# Syntax



```
class Dog
  def speak
    puts 'woof'
  end
end
```

```
@fido = Dog.new ← A new object of the Dog class
@fido.speak
```

# The Cookie and it's Cutter



- Cookie is the object, Cutter is the class that forms it
- Other analogies:
  - ▣ House is the Object, Blue Print is the Class
  - ▣ Rubber ducky is the Object, it's Mold is the Class

# New & Initialize



`Dog.new` creates a new object of the class `Dog`. If there is an `initialize` method defined in the `Dog` class, it will be called when `Dog.new` is called.

```
class Dog
  def initialize(name = 'fido')
    @name = name
  end
end
```

# More About Parameters & Arguments

```
def initialize  
  @name  
end
```

← You cannot send arguments to Dog.new because initialize is not defined with parameters.

```
def initialize(name)  
  @name = name  
end
```

← You must send arguments to Dog.new because initialize is defined with parameters.

```
def initialize(name = 'fido')  
  @name = name  
end
```

← You may send arguments to Dog.new but it isn't required. They are optional.

# Inheritance

```
class Human
  def eye_color=(color)
    @eye_color = color
  end
end
```

```
class Rubyist < Human
end
```

```
@rubyist = Rubyist.new
@rubyist.eye_color = 'brown'
```

## Modules: Ruby's Answer to Multiple Inheritance

```
module Movement
  def walk
    puts 'walking'
  end
end
```

```
class Dog
  include Movement
  def speak
    puts 'woof'
  end
end
```

```
@fido = Dog.new
@fido.walk
```

# TDD Tic Tac Toe



- clone the learn-to-program github repo
- spec tic\_tac\_toe\_spec.rb



# Running Sinatra Locally



- ❑ `ruby app.rb`
- ❑ go to `http://localhost:4567/`

# Sinatra Deployed to Heroku



In your applications root folder you need to create three files, 'app.rb', 'config.ru' and '.gems'.

To create a subdomain of heroku.com that you can use, cd into the root of your app's directory and type:

```
'heroku create <optional custom subdomain>'
```

To push your app to heroku, type:

```
'git push heroku master'
```

# app.rb



- app.rb should contain the urls you want your application to use. Example:

```
require 'rubygems'  
require 'sinatra'  
get '/' do  
  'Hello world!'  
end
```

# config.ru



- config.ru should require your app.rb file and it tells heroku to run the app using Sinatra. Example:

```
require 'app'  
run Sinatra::Application
```

# .gems



- .gems tells heroku which gems to install, in this case the only gem you'll need is Sinatra. Example:

```
sinatra -v 1.0
```

# Exercise



- Make an app in Sinatra that says “Hello World!”

# Tic Tac Toe & Sinatra



- If you finish getting your implementation of tic tac toe into sinatra and deployed to heroku, here are some features to add to your implementation:
  - ▣ doesn't allow a square to be modified after initial modification
  - ▣ declares winner (or cat's game) once all squares are taken

# Homework



- Chapters 12 & 13
- `calculator_spec.rb` from learn-to-program github repo, do `advanced_calculator_spec.rb` if you want more challenge!