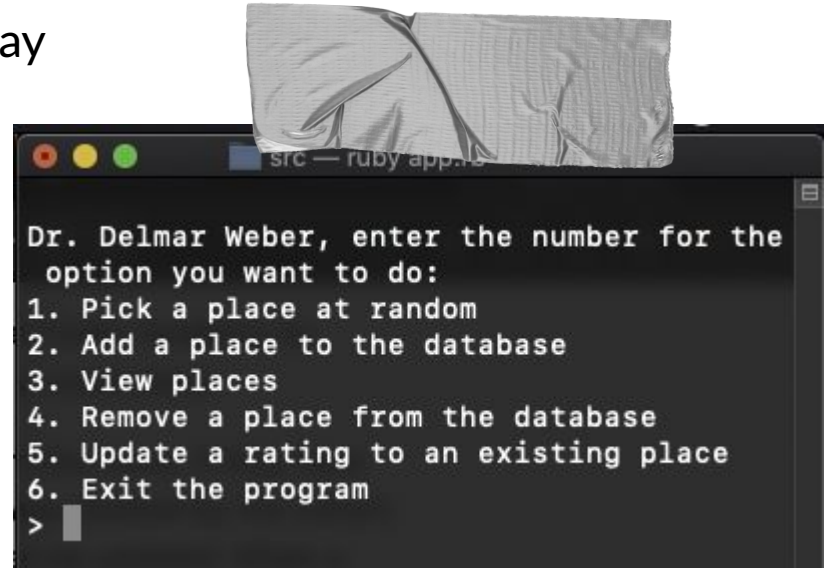

CoderAcademy Friday Selector App



Want to do something new?

Remove decision fatigue

Designed to facilitate the selection of a venue for those looking to socialise after class on Friday nights.



```
src — ruby app.r  
Dr. Delmar Weber, enter the number for the  
option you want to do:  
1. Pick a place at random  
2. Add a place to the database  
3. View places  
4. Remove a place from the database  
5. Update a rating to an existing place  
6. Exit the program  
> █
```



Code

Choose one approach to grab the audience's attention right from the start: unexpected, emotional, or simple.

→ **CSV**

Data persistence

→ **class Place**

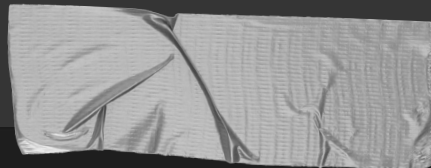
CSV converted to instance objects

→ **places = [{place1}, {place2}]**

Array of objects


CSV

Handling



```
3 PLACES_FILE = 'list_places.csv'
4
5 # ##Read information from CSV file:
6 def read_csv
7   values = []
8   File.open(PLACES_FILE).each_with_index do |line, _index|
9     values << line if line.length > 1
10  end
11  # turns .csv into array, splitting at each comma
12  places_mapped = values.map do |place|
13    place.split(',')
14  end
15  # adds .csv information into class to read the class database
16  places_mapped.each do |place|
17    Place.new(created_by: place[0], name: place[1], location: place[2], rating: place[3], visits: place[4])
18  end
19 end
20
21 # ##Write information to CSV file:
22 def update_places_csv(places)
23   File.open(PLACES_FILE, 'w') do |line|
24     places.each do |i|
25       the_line = "#{i.created_by},#{i.name},#{i.location},#{i.rating},#{i.visits}\n"
26       line.write the_line
27     end
28   end
29 end
```

class Place



```
3 # create new place instance
4 class Place
5   attr_reader :name, :location, :created_by
6   attr_accessor :visits, :rating
7   def initialize(hash)
8     @created_by = hash[:created_by]
9     @name = hash[:name]
10    @location = hash[:location]
11    @rating = hash[:rating]
12    @visits = hash[:visits]
13  end
14  def increase_visits
15    @visits = @visits.to_i + 1
16  end
17  def self.all
18    ObjectSpace.each_object(self).to_a
19  end
20 end
21
```

The places array contains instance objects of the class place. **SIX OPTIONS** from the main menu give the ability to work with this data.

```
17 loop do
18   puts "#{name}, enter the number for the option you want to do:"
19   puts '1. Pick a place at random'
20   puts '2. Add a place to the database'
21   puts '3. View places'
22   puts '4. Remove a place from the database'
23   puts '5. Update a rating to an existing place'
24   puts '6. Exit the program'
25   print '> '
26   option = STDIN.gets.strip.to_i
27
28   if option == 1
29     | run_option_1(places)
30   elsif option == 2
31     | run_option_2(places, name)
32   elsif option == 3
33     | run_option_3(places)
34   elsif option == 4
35     | run_option_4(places)
36   elsif option == 5
37     | run_option_5(places)
38   elsif option == 6
39     | run_option_6
40     | break
41   else
42     | is_valid
43   end
44 end
```