

Evidence for Implementation and Testing Unit

Rachel Johnson

E19

I.T 3 - Example of searching

Table to be searched

id	title	author
1	Brave New World	Aldous Huxley
2	Oryx and Crake	Margaret Atwood
3	The Timetraveller's Wife	Audrey Niffenegger
4	Charlie and the Chocolate Factory	Roald Dahl
5	The Girl with All the Gifts	M R Carey
6	Chinese Cinderella	Adeline Yen Mah
7	Cloud Atlas	David Mitchell
8	A Clockwork Orange	Anthony Burgess
9	The Lion, the Witch and the Wardrobe	C S Lewis
10	1984	George Orwell

(10 rows)

Search function

Function searches within the table for a specified id number and returns the associated book

```
1  require('pg')
2
3  class Book
4
5    attr_reader :id, :title, :author
6
7    def initialize(inputs)
8      @id = inputs['id'].to_i if inputs['id']
9      @title = inputs['title']
10     @author = inputs['author']
11   end
12
13   def Book.find_by_id(id)
14     db = PG.connect({dbname: "books", host: "localhost"})
15     sql = "SELECT * FROM books WHERE id = $1;"
16     values = [id]
17     db.prepare("find", sql)
18     result = db.exec_prepared("find", values)
19     db.close()
20     return result.map {|book| Book.new(book)}
21   end
22
23 end
24
25 p Book.find_by_id(5)
26
```

Result

```
[➔ PDA git:(master) ✕ ruby IT3_evidence.rb ]
[#<Book:0x007fcc45148778 @id=5, @title="The Girl with All the Gifts", @author="M R Carey">]
➔ PDA git:(master) ✕ █
```

I.T 4 - Example of sorting

Table to be sorted is same as that used for I.T 3.

Function

Function sorts the table by title in ascending order.

```
1  require('pg')
2
3  class Book
4
5      attr_reader :id, :title, :author
6
7      def initialize(inputs)
8          @id = inputs['id'].to_i if inputs['id']
9          @title = inputs['title']
10         @author = inputs['author']
11     end
12
13     def Book.sort()
14         db = PG.connect({dbname: "books", host: "localhost"})
15         sql = "SELECT * FROM books ORDER BY title ASC;"
16         db.prepare("sort", sql)
17         result = db.exec_prepared("sort", [])
18         db.close()
19         return result.map {|book| Book.new(book)}
20     end
21
22 end
23
24 Book.sort().each{|book| p book}
25 # Added .each{...} functionality above so output of sort function is easier to
26 # read in the terminal
27
```

Result

```
[→ PDA git:(master) * ruby IT4_evidence.rb ]
#<Book:0x007ffc910891d0 @id=10, @title="1984", @author="George Orwell">
#<Book:0x007ffc91089108 @id=8, @title="A Clockwork Orange", @author="Anthony Burgess">
#<Book:0x007ffc91088f28 @id=1, @title="Brave New World", @author="Aldous Huxley">
#<Book:0x007ffc91088ac8 @id=4, @title="Charlie and the Chocolate Factory", @author="Roald Dahl">
#<Book:0x007ffc91088938 @id=6, @title="Chinese Cinderella", @author="Adeline Yen Mah">
#<Book:0x007ffc910887f8 @id=7, @title="Cloud Atlas", @author="David Mitchell">
#<Book:0x007ffc91088708 @id=2, @title="Oryx and Crake", @author="Margaret Atwood">
#<Book:0x007ffc910882a8 @id=5, @title="The Girl with All the Gifts", @author="M R Carey">
#<Book:0x007ffc91088050 @id=9, @title="The Lion, the Witch and the Wardrobe", @author="C S Lewis">
#<Book:0x007ffc9108a918 @id=3, @title="The Timetraveller's Wife", @author="Audrey Niffenegger">
→ PDA git:(master) *
```

I.T 5 - Example of an array, a function that uses an array and the result

Array and function

```
1  cheesecake_ingredients = ["biscuits", "butter", "cream cheese", "sugar", "double cream", "raspberries"]
2
3  def display_ingredients(ingredients)
4    p "The ingredients in this dish are:"
5    ingredients.each {|ingredient| p ingredient}
6  end
7
8  display_ingredients(cheesecake_ingredients)
```

Result

```
[➔ PDA git:(master) x ruby IT5_evidence.rb ]
"The ingredients in this dish are:"
"biscuits"
"butter"
"cream cheese"
"sugar"
"double cream"
"raspberries"
➔ PDA git:(master) x █
```

I.T 6 Example of a hash, a function that uses the hash and the result

Hash and function

```
1  menu_prices = {
2    starter: 6.50,
3    main: 17.50,
4    dessert: 5.50
5  }
6
7  def total_cost(menu)
8    running_total = 0
9    menu.each {|key, value| running_total += value}
10   p "The total cost of the meal is £#{ '%.2f' % running_total}"
11 end
12
13 total_cost(menu_prices)
14
```

Result

```
[→ PDA git:(master) ✖ ruby IT6_evidence.rb
"The total cost of the meal is £29.50"
→ PDA git:(master) ✖ █]
```