I. T 5

An array in a program:

```ruby
@customers = [
  {
    name: "Craig",
    pets: [],
    cash: 1000
  },
  {
    name: "Zsolt",
    pets: [],
    cash: 50
  }
]
```

A function that uses the array:

```ruby
def customer_pet_count(customer)
  return customer[:pets].count
end
```

The result of the function running:

```ruby
def test_customer_pet_count
  count = customer_pet_count(@customers[0])
  assert_equal(0, count)
  p count
end
```

```
→ week_01_pet_shop git:(master) × ruby specs/pet_shop_spec.rb
Run options: --seed 54931

# Running:

...............0
....

Finished in 0.001510s, 13245.0344 runs/s, 17218.5447 assertions/s.]

20 runs, 26 assertions, 0 failures, 0 errors, 0 skips
```

I. T 6

A hash in a program:

```ruby
@pet_shop = {
    pets: [
      {
        name: "Sir Percy",
        pet_type: :cat,
        breed: "British Shorthair",
        price: 500
      },
      {
        name: "King Bagdemagus",
        pet_type: :cat,
        breed: "British Shorthair",
        price: 500
      },
      {
        name: "Sir Lancelot",
        pet_type: :dog,
        breed: "Pomsky",
        price: 1000,
      },
      {
        name: "Arthur",
        pet_type: :dog,
        breed: "Husky",
        price: 900,
      },
      {
        name: "Tristan",
        pet_type: :dog,
        breed: "Basset Hound",
        price: 800,
      },
      {
        name: "Merlin",
        pet_type: :cat,
        breed: "Egyptian Mau",
        price: 1500,
      }
    ],
    admin: {
      total_cash: 1000,
      pets_sold: 0,
    },
    name: "Camelot of Pets"
  }
```

A function that uses the hash:

```ruby
def pet_shop_name(pet_shop)
  return pet_shop[:name]
end
```

The result of the function running:

```
def test_pet_shop_name
  name = pet_shop_name(@pet_shop)
  assert_equal("Camelot of Pets", name)
  p name
end
```

```
[→ week_01_pet_shop git:(master) × ruby specs/pet_shop_spec.rb  ]
Run options: --seed 38346

# Running:

......"Camelot of Pets"
.............

Finished in 0.001464s, 13661.2024 runs/s, 17759.5632 assertions/s.

20 runs, 26 assertions, 0 failures, 0 errors, 0 skips
```

I. T 3

Function that searches data:

```
def Transaction.find(id)
  sql = "SELECT * FROM transactions
        WHERE id = $1;"
  values = [id]
  transaction_hash = SqlRunner.run(sql, values).first
  return Transaction.new(transaction_hash)
end
```

The result of the function running:

```
[1] pry(main)> Transaction.find(1)
=> #<Transaction:0x007ff83c25b200
 @amount=10.0,
 @category_id=1,
 @comment="Lost so went to MacDonalds!",
 @id=1,
 @transaction_date="2017-12-19",
 @vendor_id=1>
```

I. T 4

Function that sorts data:

```
SELECT name, price FROM shows ORDER BY price ASC;
```

The result of the function running:

```
                name                 | price
-------------------------------------+-------
 Two girls, one cup of comedy        |  6.00
 Best of Burlesque                   |  7.99
 Two become One                      |  8.50
 Urinetown                           |  8.50
 Paul Dabek Mischief                 | 12.99
 Le Haggis                           | 12.99
 Joe Stilgoe: Songs on Film - The Sequel | 16.50
 Game of Thrones - The Musical       | 16.50
 Shitfaced Shakespeare               | 16.50
 Aaabeduation - A Magic Show         | 17.99
 Camille O'Sullivan                  | 17.99
 Balletronics                        | 32.00
 Edinburgh Royal Tattoo              | 32.99
(13 rows)
```