

Unit I & T - I.T 6 - Demonstrate the use of a hash in a program.

Take screenshots of:

- A hash in a program;

```
28 ▾ @pet_shop = {
29 ▾   pets: [
30 ▾     {
31       name: "Sir Percy",
32       pet_type: :cat,
33       breed: "British Shorthair",
34       price: 500
35     },
36 ▾     {
37       name: "King Bagdemagus",
38       pet_type: :cat,
39       breed: "British Shorthair",
40       price: 500
41     },
42 ▾     {
43       name: "Sir Lancelot",
44       pet_type: :dog,
45       breed: "Pomsky",
46       price: 1000,
47     },
48 ▾     {
49       name: "Arthur",
50       pet_type: :dog,
51       breed: "Husky",
52       price: 900,
53     },
54 ▾     {
55       name: "Tristan",
56       pet_type: :dog,
57       breed: "Basset Hound",
58       price: 800,
59     },
60 ▾     {
61       name: "Merlin",
62       pet_type: :cat,
63       breed: "Egyptian Mau",
64       price: 1500,
65     }
66   ],
```

- A function that uses the hash;

```
31
32 def find_pet_by_name(data_category, name)
33   pet = {}
34   @pet_shop[:pets].each {|pet_hash|
35     if pet_hash[:name] == name
36       return pet_hash
37     end
38     pet.merge(pet_hash)
39   }
40   return pet[:name]
41 end
42
```

- The result of the function running;

```
122  
123     def test_find_pet_by_name__returns_pet  
124         pet = find_pet_by_name(@pet_shop, "Arthur")  
125         assert_equal("Arthur", pet[:name])  
126     end  
127
```

```
➔ specs git:(master) ✖ ruby pet_shop_spec.rb
```

```
Run options: --seed 53877
```

```
# Running:
```

```
.
```

```
Finished in 0.000679s, 1472.7542 runs/s, 1472.7542 assertions/s.
```

```
1 runs, 1 assertions, 0 failures, 0 errors, 0 skips
```

```
➔ specs git:(master) ✖
```