```
//: # Lesson 4 Exercises - Collections
import UIKit

//: ## Array initialization

//: ### Exercise 1
//: 1a) Initialize the array, cuteAnimals. It should be of type String.
 Type your answer below.
let cuteAnimals: Array<String> = []

//: 1b) Initialize an array of 5 bools using array literal syntax.
let arrayLiteral: [Bool] = [true, false, false, true, true]

//: ## Array operations: count, insert, append, remove,
 retrieveWithSubscript

//: ### Exercise 2
//: Print out the number of spaniels in the array below.
var spaniels = ["American Cocker", "Cavalier King Charles", "English
 Springer", "French", "Irish Water","Papillon", "Picardy","Russian",
 "French", "Welsh Springer"]
print(spaniels.count)

//: ### Exercise 3
//: Insert "indigo" into the array below so that its index is after
 "blue" and before "violet".
var colors = ["red", "orange", "yellow", "green", "blue", "violet"]
colors.insert("indigo", at: 5)

//: ### Exercise 4
//: Insert "English Cocker" into the spaniels array so that its index
 is before "English Springer".
spaniels.insert("English Cocker", at: 2)

//: ### Exercise 5
//: Append "Barcelona" to the end of the olympicHosts array.
var olympicHosts = ["London", "Beijing","Athens", "Sydney", "Atlanta"]
olympicHosts.append("Barcelona")

//: ### Exercise 6
//: Remove "Lyla" and "Daniel" from the waitingList array and add them
 to the end  of the admitted array.
var admitted = ["Jennifer", "Vijay", "James"]
var waitingList = ["Lyla", "Daniel", "Isabel", "Eric"]
waitingList[0...1].removeAll()
print(waitingList)
admitted.append(contentsOf: waitingList)
```

```swift
//: ### Exercise 7
//: Using subscript syntax, print out the 2nd and 3rd names from the admitted array.
print(admitted[2...3])

//: ## Dictionary initialization

//: ### Exercise 8
//: a) Initialize an empty dictionary which holds Strings as keys and Bools as values.
let dictionary: [String: Bool] = [:]

//: b) Initialize a dictionary using array literal syntax. The keys should be the Strings: "Anchovies", "Coconut", "Cilantro", "Liver" and each value should be a Bool representing whether you like the food or not.
let foodDictionary = [
    "Anchovies" : true,
    "Coconut" : true,
    "Cilantro" : true,
    "Liver" : false
]

//: ## Dictionary operations: count, insert, remove, update, retrieve with subscript

//: ### Exercise 9
//: Insert an entry for George H.W. Bush to the dictionary below.
var presidentialPetsDict = ["Barack Obama":"Bo", "Bill Clinton": "Socks", "George Bush": "Miss Beazley", "Ronald Reagan": "Lucky"]
presidentialPetsDict["George H. W. Bush"] = "Millie"
print(presidentialPetsDict)
//desired output
// ["Barack Obama":"Bo", "George Bush": "Miss Beazley","Bill Clinton": "Socks", "George H. W. Bush": "Millie", "Ronald Reagan": "Lucky"]

//: ### Exercise 10
//: Remove the entry for "George Bush" and replace it with an entry for "George W. Bush".
presidentialPetsDict["George Bush"] = nil
presidentialPetsDict["George W. Bush"] = "Miss Beazley"

//: ### Exercise 11
//: We've initialized a new dictionary of presidentialDogs with the entries from presidentialPets. Update the entry for Bill Clinton by replacing "Socks" the cat with "Buddy" the dog.
var presidentialDogs = presidentialPetsDict
presidentialDogs["Bill Clinton"] = "Buddy"
```

```
//: ### Exercise 12
//: Use subscript syntax to fill in the println statement below and
 produce the following string: "Michele Obama walks Bo every morning."
 You'll need to retrieve a value from the presidentialDogs dictionary
 and unwrap it using if let.
print("Michele Obama walks \(presidentialDogs["Barack Obama"] ?? "")
 every morning.")


//: ### Exercise 13
var studioAlbums = ["Led Zeppelin":1969, "Led Zeppelin II": 1969, "Led
 Zeppelin III": 1970, "Led Zeppelin IV": 1971, "Houses of the
 Holy":1973, "Physical Graffiti": 1975, "Presence":1976, "In Through
 the Out Door":1979, "Coda":1982]
//: How many studio albums did Led Zeppelin release?
//: how many elements are in the dictionary?
studioAlbums.count
```