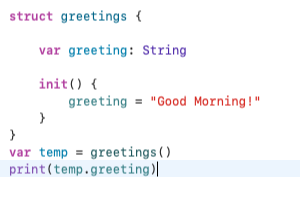
Exercise 2

//initializers

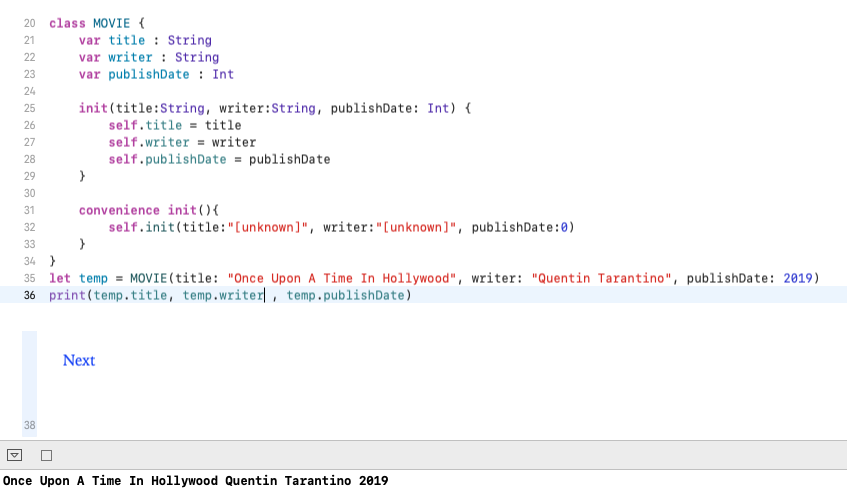
1.



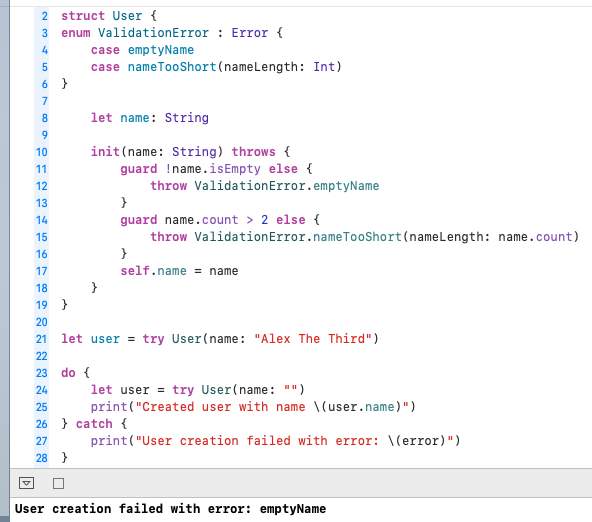
2.

* **Rule 1:** A designated initializer must call a designated initializer from its immediate superclass.
* **Rule 2:** A convenience initializer must call another initializer from the *same class*.
* **Rule 3:** A convenience initializer must ultimately call a designated initializer.

3.

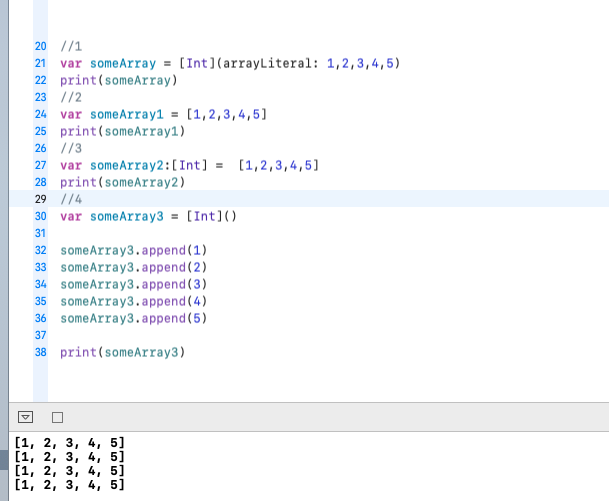


4.

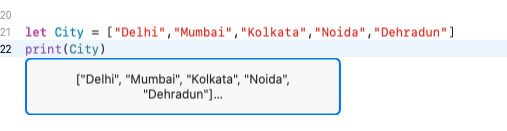


// Array

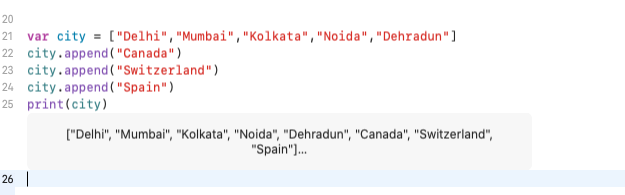
1.



2.

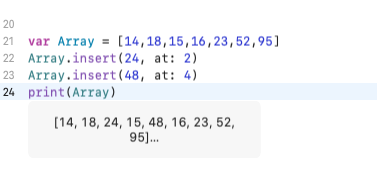


3.





4.



//Set

1.



2.



3.



4.



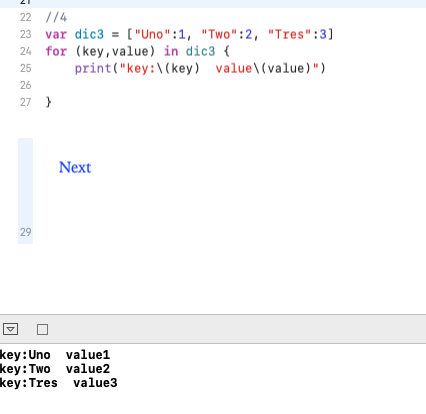
5.



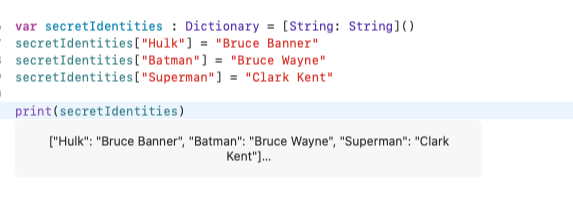
//Dictionary

1.





2.



3.



4.



//Subscripts

1.

Classes, structures, and enumerations can define *subscripts*, which are shortcuts for accessing the member elements of a collection, list, or sequence. You use subscripts to set and retrieve values by index without needing separate methods for setting and retrieval. For example, you access elements in an Array instance as someArray[index] and elements in a Dictionary instance as someDictionary[key].

subscript(index: Int) -> Int {

get {

// Return an appropriate subscript value here. }

set(newValue) {

// Perform a suitable setting action here.

}

}

2.

