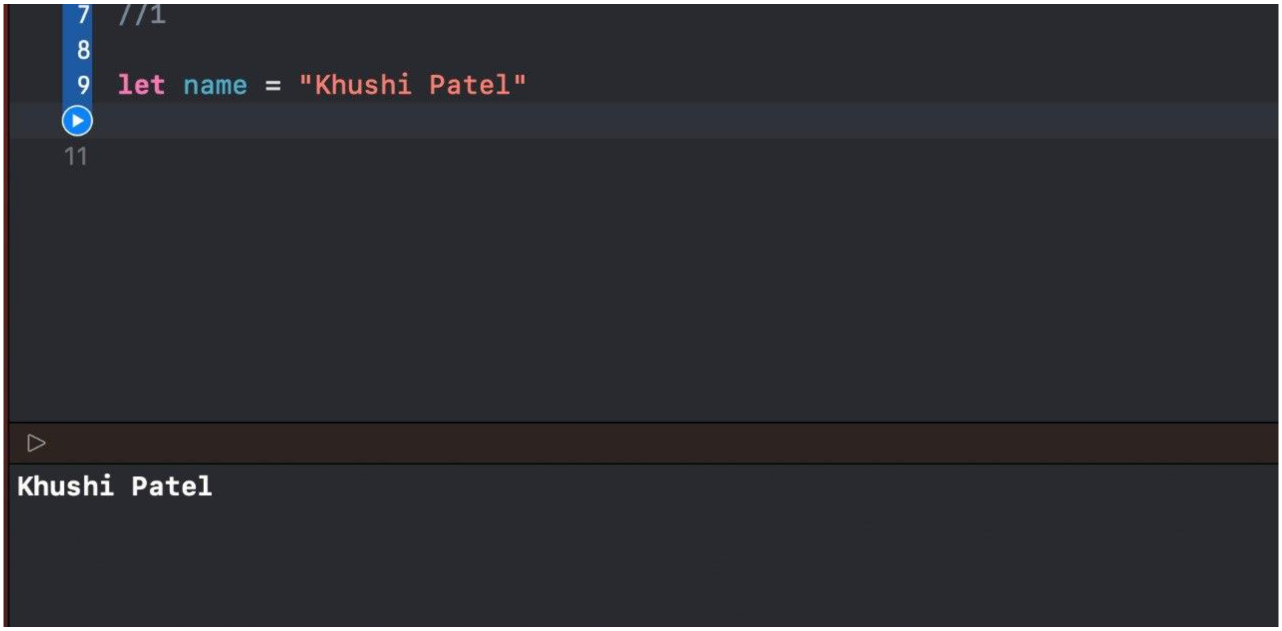
Name- Khushi Nitinkumar Patel
PRN- 2020BTECS00037
Batch- T2

Assignment no- 5

Exercise - String Basics

1. Create a `name` constant and assign it a string literal representing your name.

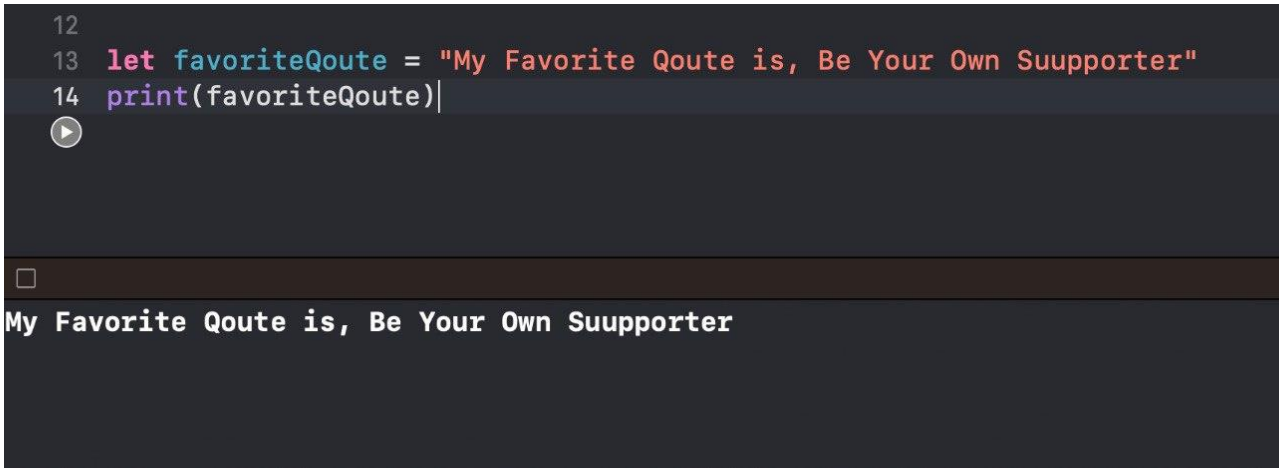


```
//1

let name = "Khushi Patel"
```

```
Khushi Patel
```

2. Create a `favoriteQuote` constant and assign it the following string literal:

- "My favorite quote is <INSERT QUOTE HERE>."



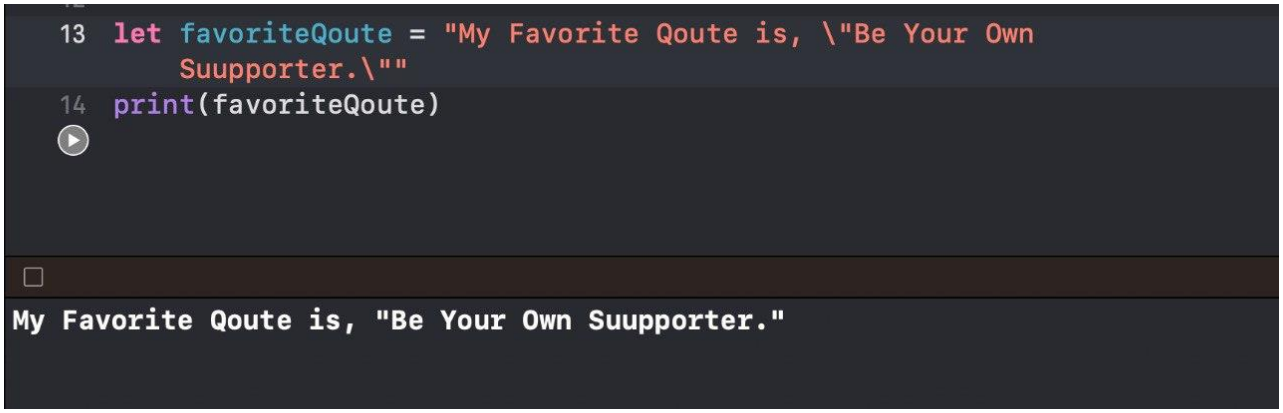
```
let favoriteQoute = "My Favorite Qoute is, Be Your Own Suupporter"
print(favoriteQoute)
```

```
My Favorite Qoute is, Be Your Own Suupporter
```

3. Write in your own favorite quote where indicated, and be sure to include escaped quotation marks. When finished, print the value of `favoriteQuote`.

Example: If your favorite quote is "The grass is always greener on the other side" the value of `favoriteQuote` should be such that printing `favoriteQuote` results in the following:

`My favorite quote is "The grass is always greener on the other side."`

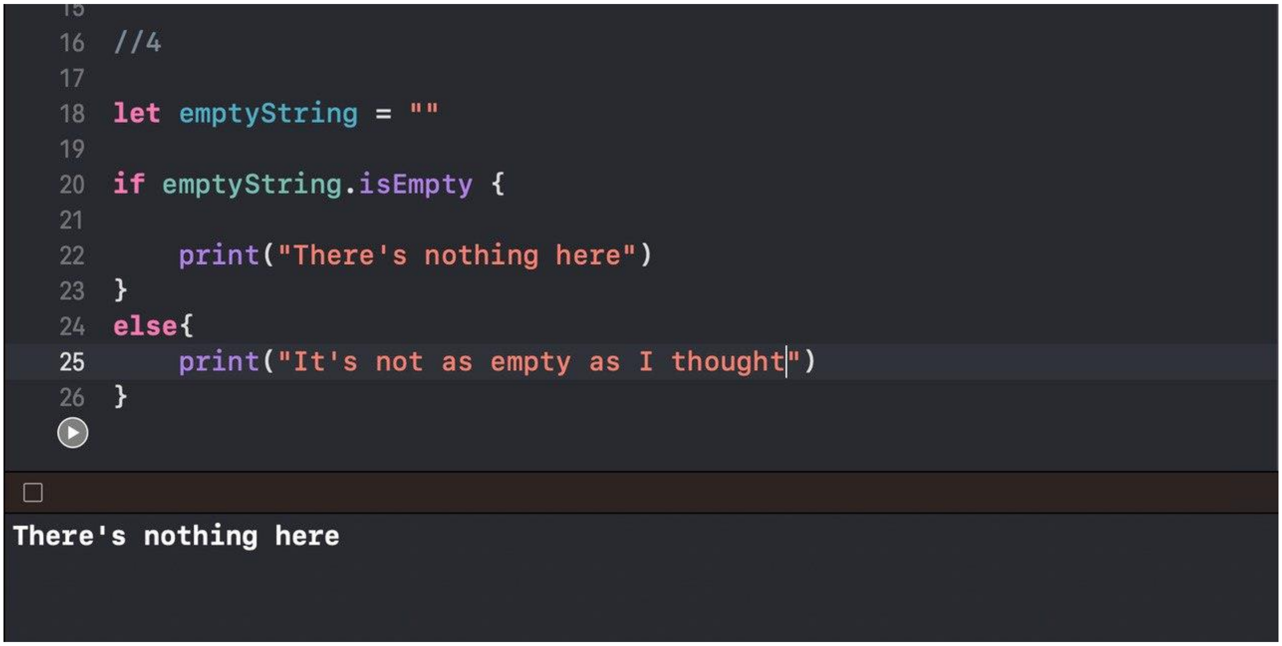


```swift
let favoriteQoute = "My Favorite Qoute is, \"Be Your Own Suupporter.\""
print(favoriteQoute)
```

```
My Favorite Qoute is, "Be Your Own Suupporter."
```

4. Write an if-else statement that prints "There's nothing here" if `emptyString` is empty, and "It's not as empty as I thought" otherwise.

let emptyString = ""

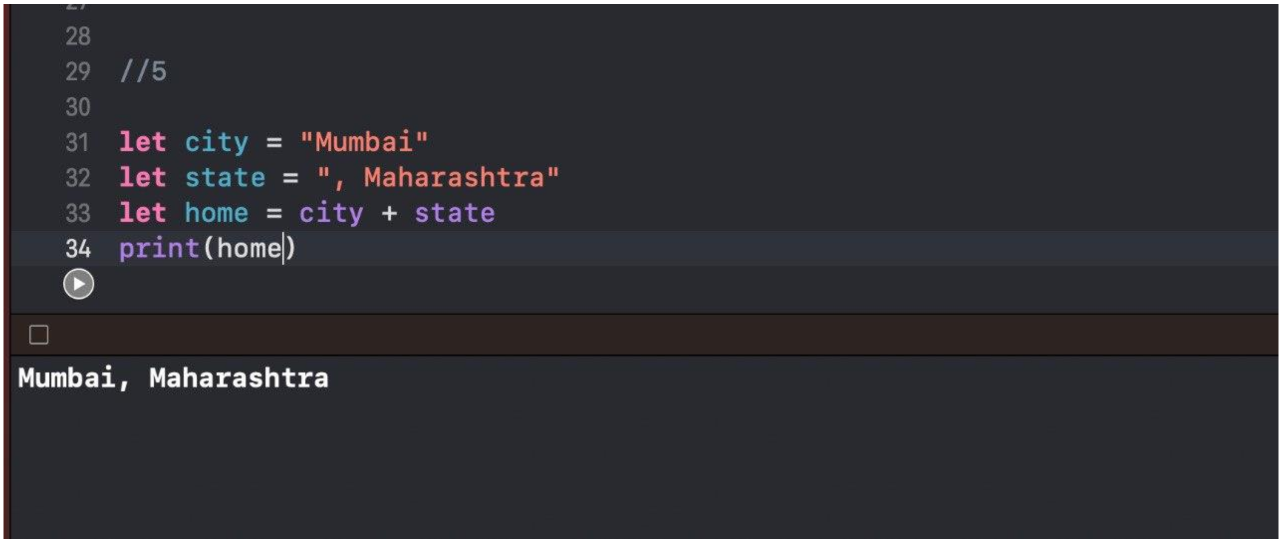


```swift
//4

let emptyString = ""

if emptyString.isEmpty {

    print("There's nothing here")
}
else{
    print("It's not as empty as I thought")
}
```

```
There's nothing here
```

Exercise - Concatenation and Interpolation

5. Create a `city` constant and assign it a string literal representing your home city. Then create a `state` constant and assign it a string literal representing your home state. Finally, create a `home` constant and use string concatenation to assign it a string representing your home city and state (i.e. Portland, Oregon). Print the value of `home`.

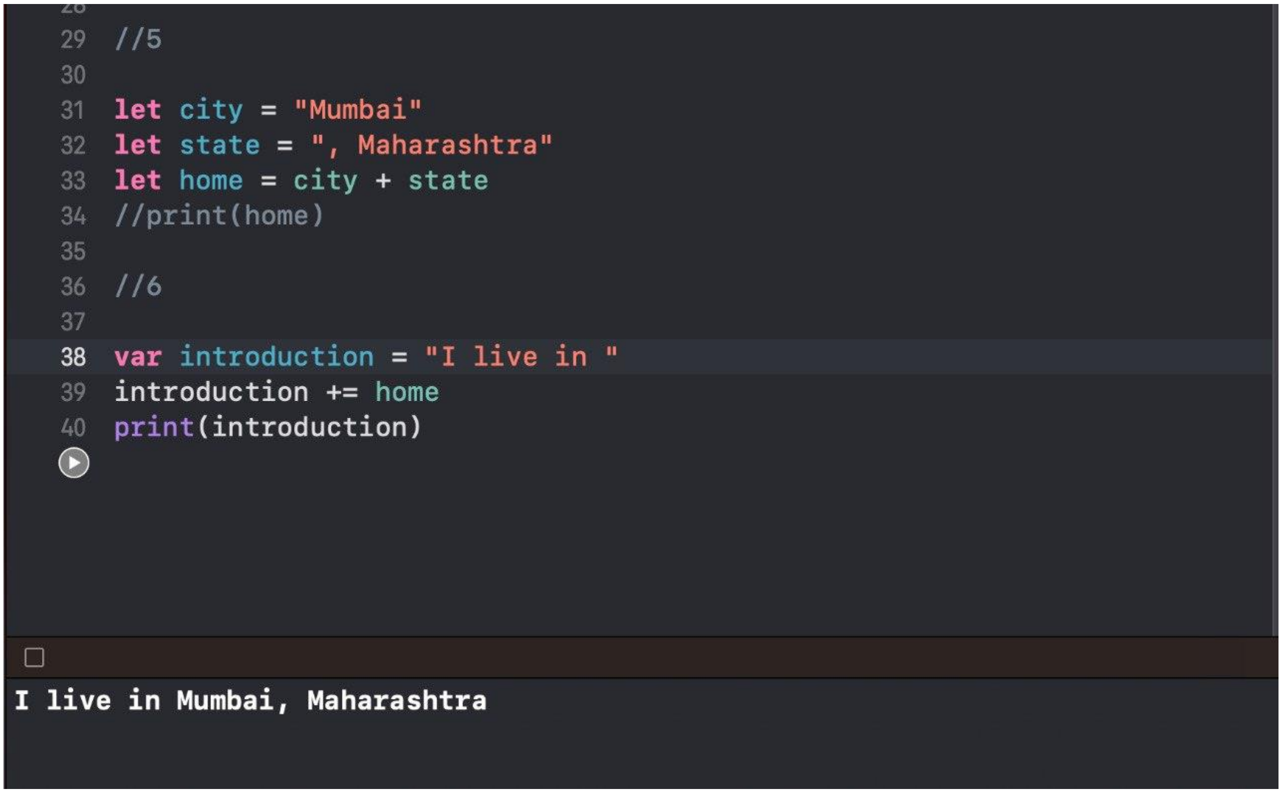


```swift
//5

let city = "Mumbai"
let state = ", Maharashtra"
let home = city + state
print(home)
```

```
Mumbai, Maharashtra
```

6. Use the compound assignment operator (`+=`) to add `home` to `introduction` below. Print the value of `introduction`.

var introduction = "I live in "



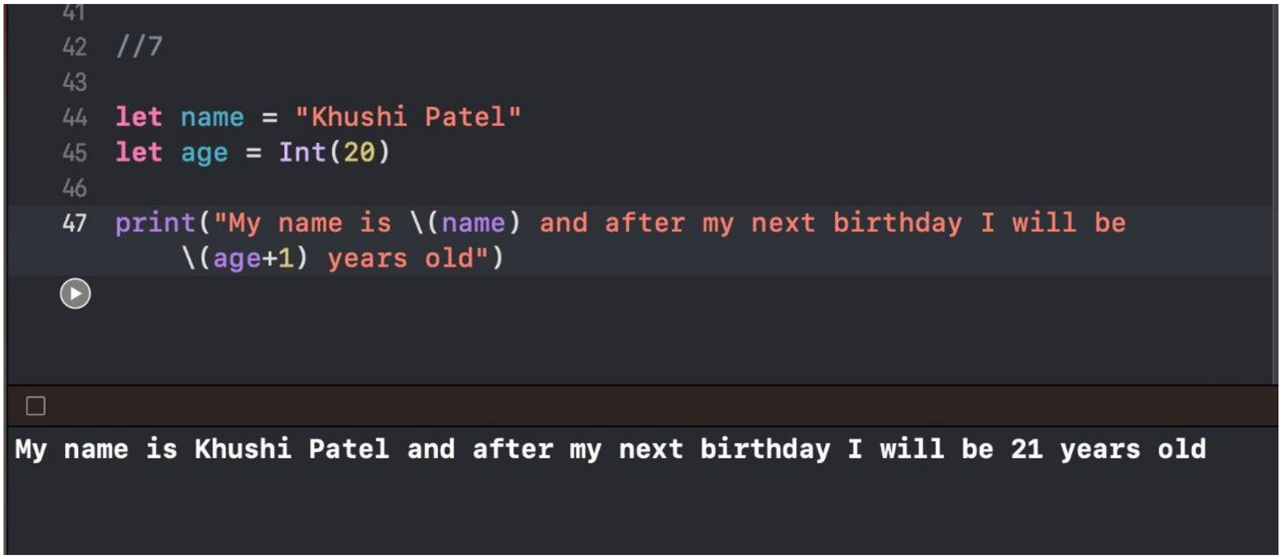
```swift
//5

let city = "Mumbai"
let state = ", Maharashtra"
let home = city + state
//print(home)

//6

var introduction = "I live in "
introduction += home
print(introduction)
```

```
I live in Mumbai, Maharashtra
```

7. Declare a `name` constant and assign it your name as a string literal. Then declare an `age` constant and give it your current age as an `Int`. Then print the following phrase using string interpolation:

- "My name is <INSERT NAME HERE> and after my next birthday I will be <INSERT AGE HERE> years old."

Insert `name` where indicated, and insert a mathematical expression that evaluates to your current age plus one where indicated.



```
//7

let name = "Khushi Patel"
let age = Int(20)

print("My name is \(name) and after my next birthday I will be
    \(age+1) years old")
```

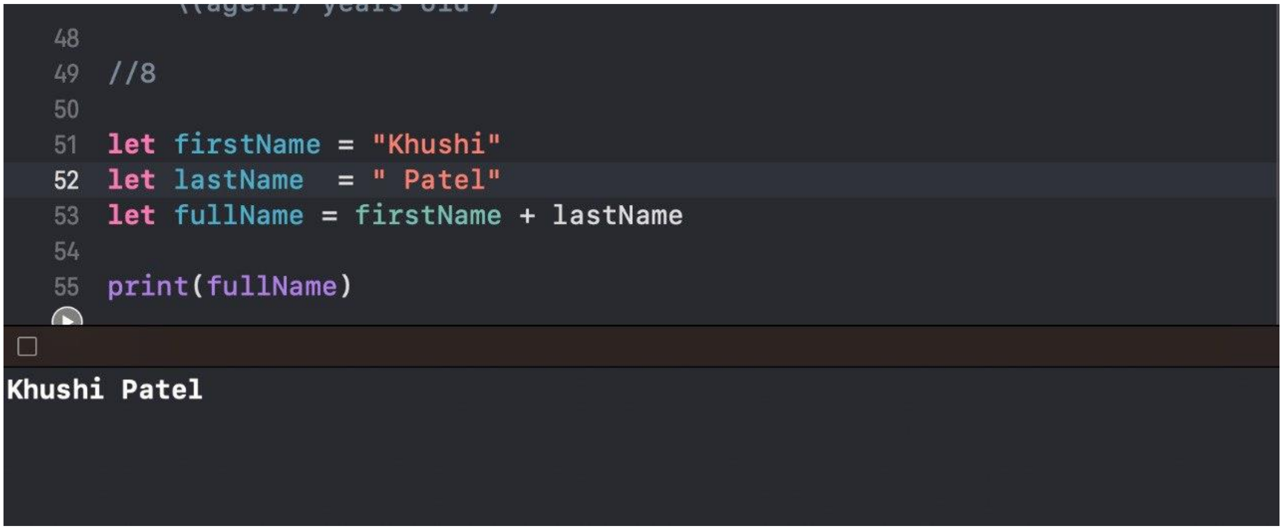
□

```
My name is Khushi Patel and after my next birthday I will be 21 years old
```

App Exercise - Notifications

8. In your app, you may want to search for other users. This would be easier with first and last names stored separately. This is not an uncommon practice.

Create `firstName` and `lastName` constants and assign them string literals representing a user's first name and last name, respectively. Create a `fullName` constant that uses string concatenation to combine `firstName` and `lastName`. Print the value of `fullName`.



```
    \(age+1) years old")

//8

let firstName = "Khushi"
let lastName  = " Patel"
let fullName = firstName + lastName

print(fullName)
```

□

```
Khushi Patel
```

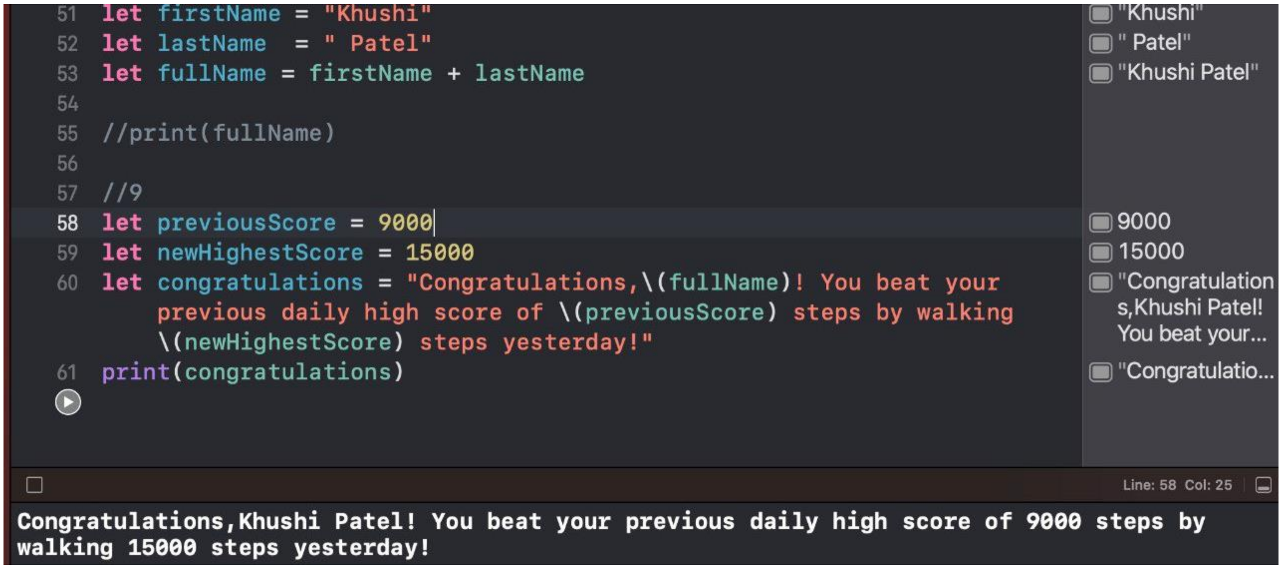
9. Occasionally users of your fitness tracking app will beat previous goals or records. You may want to notify them when this happens for encouragement purposes. Create a new constant `congratulations` and assign it a string literal that uses string interpolation to create the following string:

- "Congratulations, <INSERT USER'S FULL NAME HERE>! You beat your previous daily high score of <INSERT PREVIOUS HIGHEST STEPS HERE> steps by walking <INSERT NEW HIGHEST STEPS HERE> steps yesterday!"

Insert `fullName`, `previousBest` and `newBest` where indicated. Print the value of `congratulations`.

let previousBest = 14392

let newBest = 15125

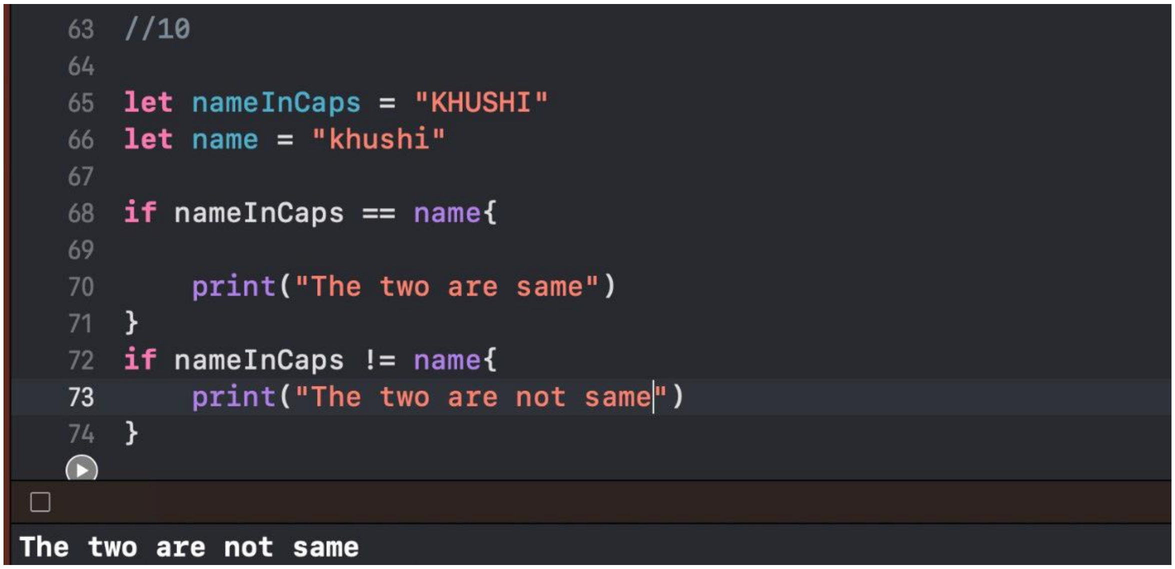


```swift
let firstName = "Khushi"
let lastName  = " Patel"
let fullName = firstName + lastName

//print(fullName)

//9
let previousScore = 9000
let newHighestScore = 15000
let congratulations = "Congratulations,\(fullName)! You beat your
    previous daily high score of \(previousScore) steps by walking
    \(newHighestScore) steps yesterday!"
print(congratulations)
```

```
Congratulations,Khushi Patel! You beat your previous daily high score of 9000 steps by
walking 15000 steps yesterday!
```

Exercise - String Equality and Comparison

10. Create two constants, `nameInCaps` and `name`. Assign `nameInCaps` your name as a string literal with proper capitalization. Assign `name` your name as a string literal in all lowercase. Write an if-else statement that checks to see if `nameInCaps` and `name` are the same. If they are, print "The two

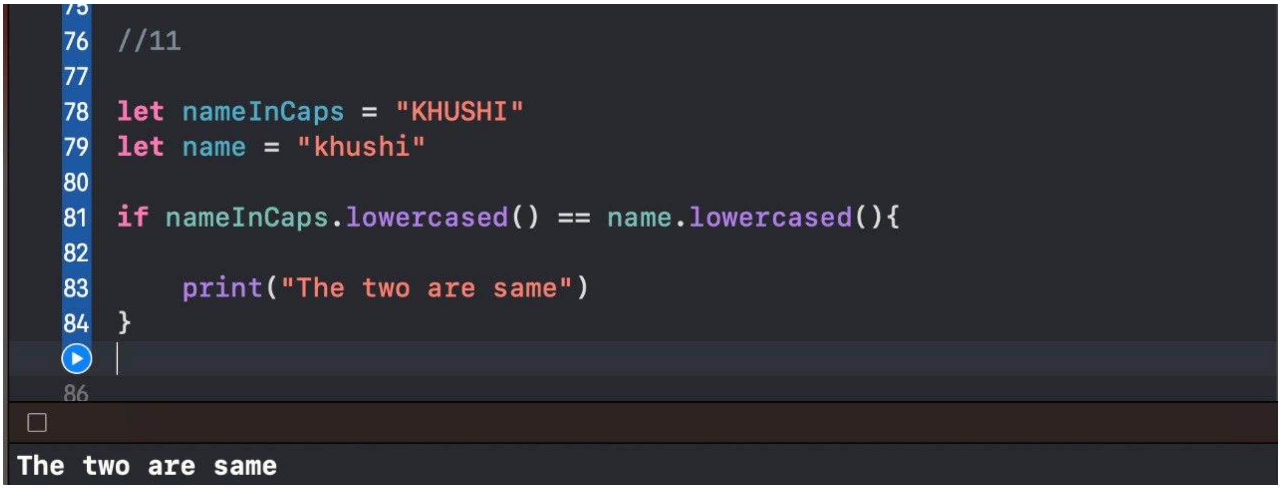


```swift
//10

let nameInCaps = "KHUSHI"
let name = "khushi"

if nameInCaps == name{

    print("The two are same")
}
if nameInCaps != name{
    print("The two are not same")
}
```

```
The two are not same
```

11. Write a new if-else statement that also checks to see if `nameInCaps` and `name` are the same. However, this time use the `lowercased()` method on each constant to compare the lowercase version of the strings. If they are equal, print the following statement using string interpolations:

- "<INSERT LOWERCASED VERSION OF `nameInCaps` HERE> and <INSERT LOWERCASED VERSION OF `name` HERE> are the same."

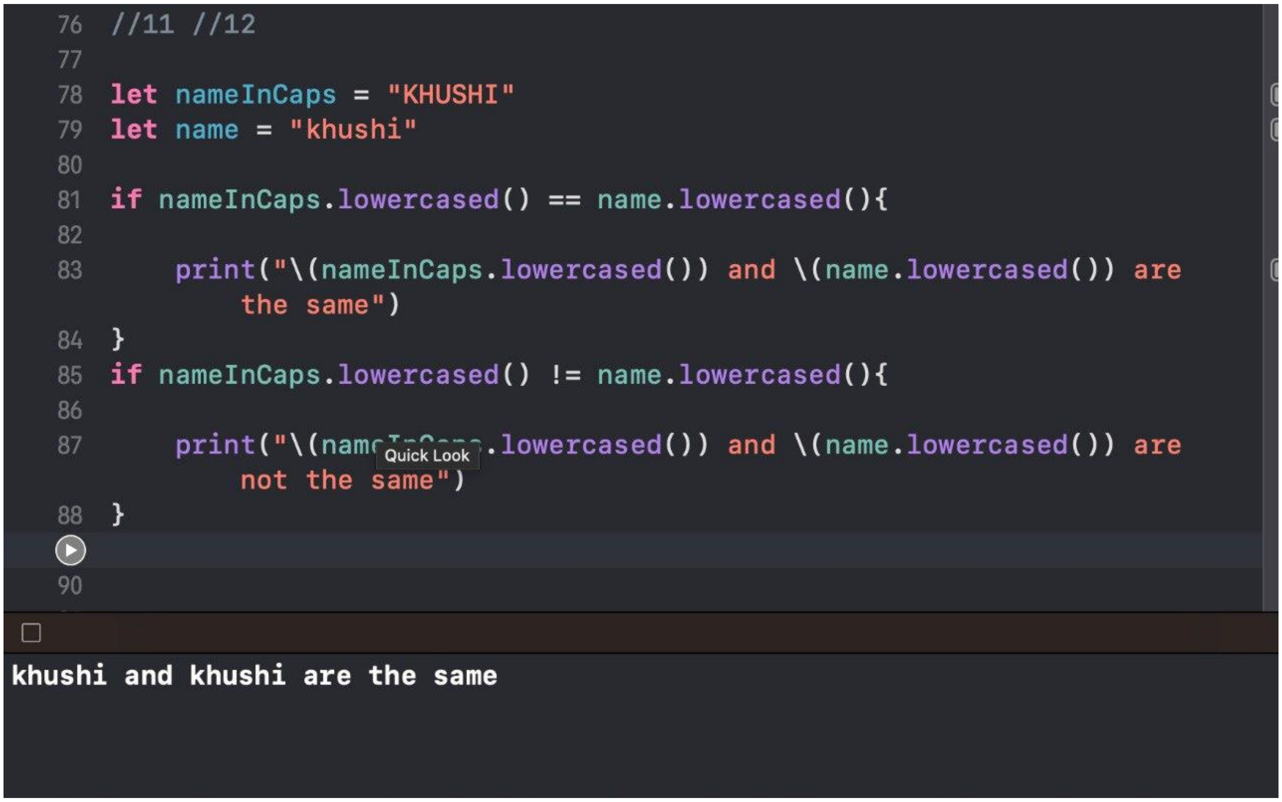


```
//11

let nameInCaps = "KHUSHI"
let name = "khushi"

if nameInCaps.lowercased() == name.lowercased(){

    print("The two are same")
}
```

```
The two are same
```

12. If they are not equal, print the following statement using string interpolation:

- "<INSERT LOWERCASED VERSION OF `nameInCaps` HERE> and <INSERT LOWERCASED VERSION OF `name` HERE> are not the same."

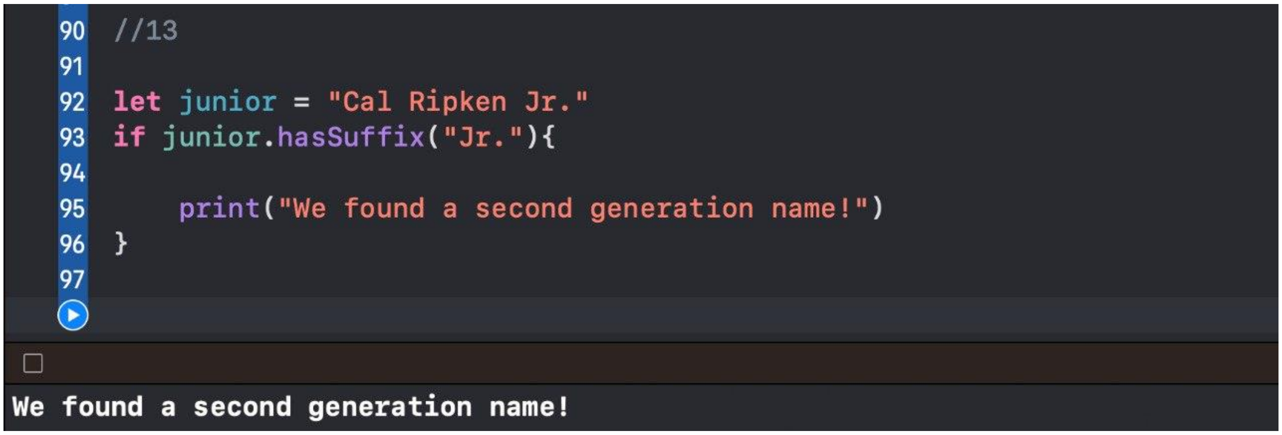


```
//11 //12

let nameInCaps = "KHUSHI"
let name = "khushi"

if nameInCaps.lowercased() == name.lowercased(){

    print("\(nameInCaps.lowercased()) and \(name.lowercased()) are the same")
}
if nameInCaps.lowercased() != name.lowercased(){

    print("\(nameInCaps.lowercased()) and \(name.lowercased()) are not the same")
}
```

```
khushi and khushi are the same
```

13. Imagine you are looking through a list of names to find any that end in "Jr." Write an if statement below that will check if `junior` has the suffix "Jr." If it does, print "We found a second generation name!"

let junior = "Cal Ripken Jr."



```swift
//13

let junior = "Cal Ripken Jr."
if junior.hasSuffix("Jr."){

    print("We found a second generation name!")
}

```

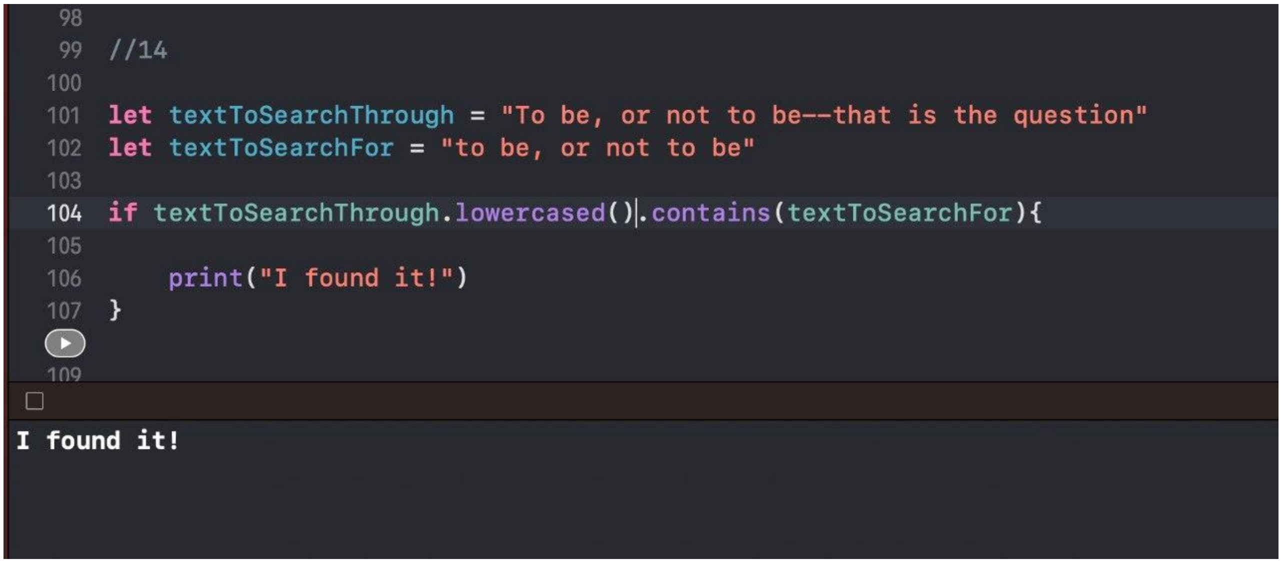
```
We found a second generation name!
```

14. Suppose you are trying to find a document on your computer that contains Hamlet's famous soliloquy written by Shakespeare. You write a simple app that will check every document to see if it contains the phrase "to be, or not to be." You decide to do part of this with the `contains(_:)` method. Write an if statement below that will check if `textToSearchThrough` contains `textToSearchFor`. If it does, print "I found it!" Be sure to make this functionality case insensitive.

import Foundation

let textToSearchThrough = "To be, or not to be--that is the question"

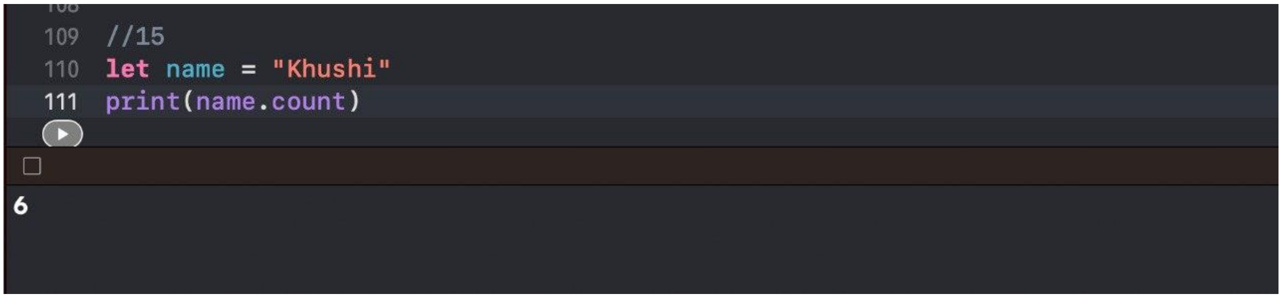
let textToSearchFor = "to be, or not to be"



```swift

//14

let textToSearchThrough = "To be, or not to be--that is the question"
let textToSearchFor = "to be, or not to be"

if textToSearchThrough.lowercased().contains(textToSearchFor){

    print("I found it!")
}

```

```
I found it!
```

15. Print to the console the number of characters in your name by using the `count` property on `name`.



```swift
//15
let name = "Khushi"
print(name.count)
```

□

```
6
```

App Exercise - Password Entry and User Search

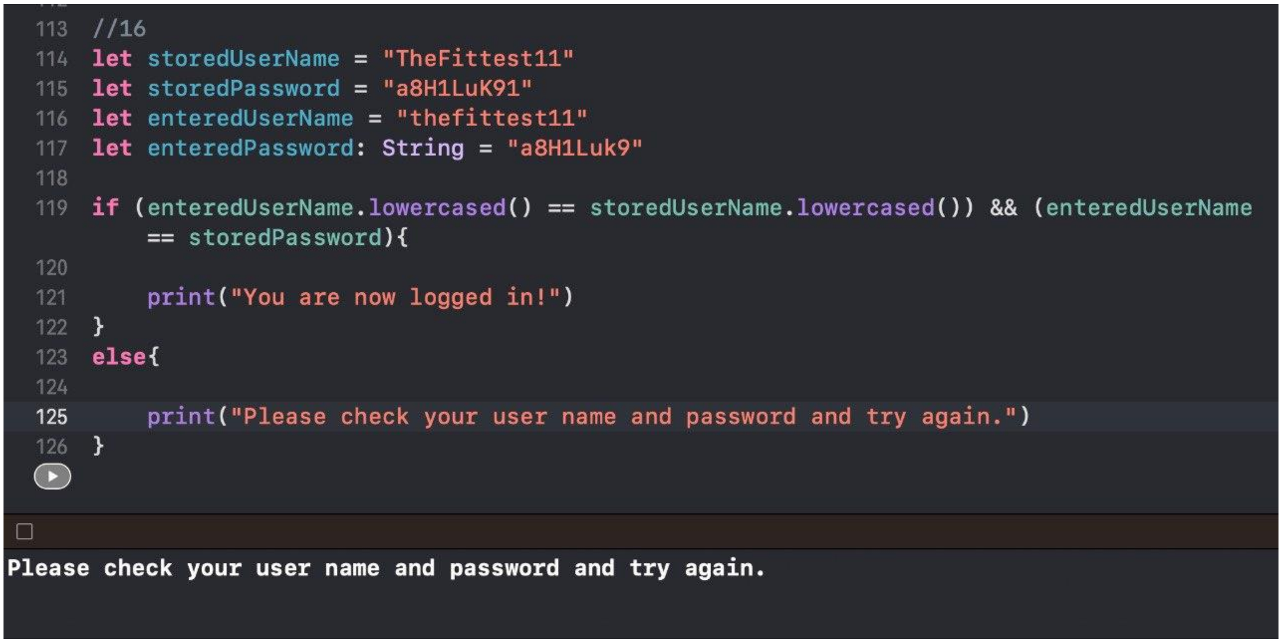
16.You think it might be fun to incorporate some friendly competition into your fitness tracking app. Users will be able to compete with friends in small fitness challenges. However, to do this users will need a password-protected account. Write an if-else statement below that will check that the entered user name and password match the stored user name and password. While the password should be case sensitive, users should be able to log in even if their entered user name has the wrong capitalization. If the user name and password match, print "You are now logged in!" Otherwise, print "Please check your user name and password and try again."

let storedUserName = "TheFittest11"

let storedPassword = "a8H1LuK91"

let enteredUserName = "thefittest11"

let enteredPassword: String = "a8H1Luk9"



```swift
//16
let storedUserName = "TheFittest11"
let storedPassword = "a8H1LuK91"
let enteredUserName = "thefittest11"
let enteredPassword: String = "a8H1Luk9"

if (enteredUserName.lowercased() == storedUserName.lowercased()) && (enteredUserName == storedPassword){

    print("You are now logged in!")
}
else{

    print("Please check your user name and password and try again.")
}
```

□

```
Please check your user name and password and try again.
```

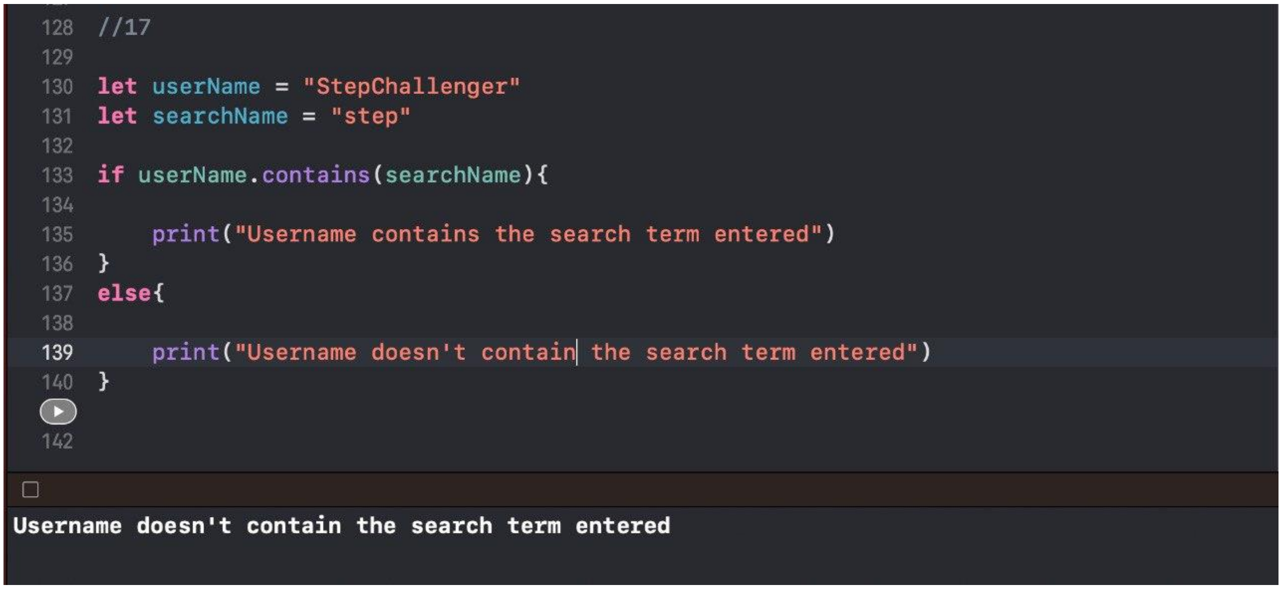
17. Now that users can log in, they need to be able to search through a list of users to find their friends. This might normally be done by having the user enter a name, and then looping through all user names to see if a user name contains the search term entered. You'll learn about loops later, so for now you'll just work through one cycle of that. Imagine you are searching for a friend whose user name is StepChallenger. You enter "step" into a search bar and the app begins to search. When the app comes to the user name "stepchallenger," it checks to see if "StepChallenger" contains "step."

Using `userName` and `searchName` below, write an if-else statement that checks to see if `userName` contains the search term. The search should *not* be case sensitive.

import Foundation

let userName = "StepChallenger"

let searchName = "step"



```
//17

let userName = "StepChallenger"
let searchName = "step"

if userName.contains(searchName){

    print("Username contains the search term entered")
}
else{

    print("Username doesn't contain the search term entered")
}

```

```
Username doesn't contain the search term entered
```

****