**>> Hello, playground**

let playground = "Hello, playground"

print(playground)

print(" Output: \(playground)")

var firstname = "Sujeet"

var lastname:String = "Lohan"

var address:String

address = "Davis Campus"

print("First Name: \(firstname)")

print("Last Name: \(lastname)")

print(" Address: \(address)")

**>> Creating a mutable string**

let playground = "Hello, playground"

var mutablePlayground = "Hello, mutable playground"

mutablePlayground += " is an Example "

print(" Output of mutable string: \(mutablePlayground)")

**>> mutablePlayground’s Characters:**

let playground = "Hello, playground"

var mutablePlayground = "Hello, mutable playground"

mutablePlayground += "!"

for ch: Character in mutablePlayground.characters

{

print("\(ch)")

}

print("Displaying characters of play ground")

for c: Character in playground.characters

{

print("\(c)")

} print("Total characters : \(playground.characters.count)")

\*/

**>> unicode examples**

let oneCoolDude = "\u{1F60E}"

print(oneCoolDude)

**>> Finding the fifth character and printing range**

let playground = "Hello, playground"

let start = playground.startIndex

let end = playground.index(start, offsetBy: 4)

let fifthCharacter = playground[end] // 'o' print(fifthCharacter)

let range = start...end

print(playground[range])

print(oneCoolDude)

**>> Code Examples on Optionals**

var programName: String?

programName = "iOS development"

print(programName)

**>> Adding a condition**

var errorCodeString: String?

errorCodeString = "404"

print(errorCodeString)

if errorCodeString != nil {

let theError = errorCodeString! // here with !, you are using forced unwrapping

print(theError)

}

**>> Optional binding - used to detect whether an optional contains a value**

import Cocoa

var errorCodeString: String?

errorCodeString = "404"

if let theError = errorCodeString {

}

print(theError)

**>> Nesting optional binding**

import Cocoa

var errorCodeString: String?

errorCodeString = "404"

if let theError = errorCodeString {

print(theError)

if let errorCodeInteger = Int(theError) {

print("\(theError): \(errorCodeInteger)") }

}

**>> implicitly unwrapping optionals binding**

import Cocoa

var errorCodeString: String!

errorCodeString="404"

print(errorCodeString)