

TEB



iOS Programlama Eğitimi

iOS



- iPhone, iPad, iPod Touch ve Apple Watch isimli cihazlar üzerinde çalışan işletim sistemi
- iPhone OS 1,2,3
- iOS 4,5,6,7,8
- En güncel versiyon iOS 8

Cocoa, Cocoa Touch & Xcode



- Cocoa: Uygulama geliştirme framework'ü
- Cocoa Touch : Cocoa'nın iOS için özelleşmiş formu
- Xcode : Geliştirme Ortamımız (IDE)
 - Mac App Store'dan ücretsiz indirilebilir
 - Güncel versiyon Xcode 6
 - Gerekli bütün araçları içerisinde barındırıyor
 - Oldukça gerçekçi bir simulatöre sahip

iPhone Ailesi



- iPhone & iPhone 3G & iPhone 3GS

iOS 7 ve sonrası

- iPhone 4

iOS 8 ve sonrası

- iPhone 4S
- iPhone 5 & iPhone 5C & iPhone 5S
- iPhone 6 & iPhone 6 Plus

iPhone & iPad & iPod Touch



- iPad
 - iPad 2
 - iPad 3rd Gen. & iPad 4th Gen.
 - iPad Mini & iPad Mini Retina
 - iPad Air
-
- iPod Touch 1-5. Nesil

Apple Watch



- 2015'te piyasaya çıkacak



Objective - C



- Cocoa ve Cocoa Touch platformlarının geliştirme dili
- 80'lerde Brad Cox tarafından oluşturuldu.
- Steve Jobs'ın şirketi Next tüm haklarını satın aldı
- Apple'ın 1996 yılında Next'i satın almasıyla Apple'ın resmi dili haline geldi
- 2014 Eylül ayına kadar ☺

Swift



- Apple'ın taptaze programlama dili
- Reasons of creation,
 - Obj-C 'nin garip syntax'ı
 - Manuel hafıza yönetimi
 - Giriş bariyerinin diğer dillere göre yüksek olması
 - Obj-C'de bir türlü çözilemeyen problemler
- Getirileri
 - Hız
 - Öğrenme ve programlama kolaylığı
 - Daha basit mimari
 - Hafıza yönetimi ile uğraşmama

Swift'e Giriş



```
println("Hello World")
```

Değişken & Sabitler



- Değişken tanımlarken ***var***
- Sabit tanımlarken ***let***

Değişken & Sabitler



- Değişken tanımlarken ***var***
- Sabit tanımlarken ***let***

1. `var myVariable = 42`

Değişken & Sabitler



- Değişken tanımlarken ***var***
 - Sabit tanımlarken ***let***
-
1. `var myVariable = 42`
 2. `let myConstant = 123`

Değişken & Sabitler



- Değişken tanımlarken ***var***
 - Sabit tanımlarken ***let***
-
1. `var myVariable = 42`
 2. `let myConstant = 123`
 3. `myVariable = 120`

Değişken & Sabitler



- Değişken tanımlarken ***var***
- Sabit tanımlarken ***let***

1. `var myVariable = 42`
2. `let myConstant = 123`
3. `myVariable = 120`
4. `myConstant = 10`

Değişken & Sabitler



- Değişken tanımlarken ***var***
- Sabit tanımlarken ***let***

1. `var myVariable = 42`
2. `let myConstant = 123`
3. `myVariable = 120`
4. ~~`myConstant = 10`~~

Type Safety and Type Inference



- Değişkenlerin tipini her zaman yazmak zorunda değilsiniz

Type Safety and Type Inference



- Değişkenlerin tipini her zaman yazmak zorunda değilsiniz
 1. `let myDoubleConst:Double = 70`

Type Safety and Type Inference



- Değişkenlerin tipini her zaman yazmak zorunda değilsiniz
 1. `let myDoubleConst:Double = 70`
 2. `let otherDoubleConts = 70.0`

Type Safety and Type Inference



- Değişkenlerin tipini her zaman yazmak zorunda değilsiniz
 1. `let myDoubleConst:Double = 70`
 2. `let otherDoubleConts = 70.0`
 3. `let myIntConst = 70`

Type Safety and Type Inference



- Fakat bir değişkenin tipi sonradan değiştirilemez.

Type Safety and Type Inference



- Fakat bir değişkenin tipi sonradan değiştirilemez.
 1. `var customerName = "Süleyman"`

Type Safety and Type Inference



- Fakat bir değişkenin tipi sonradan değiştirilemez.
 1. `var customerName = "Süleyman"`
 2. `customerName = 1000`

Type Safety and Type Inference



- Fakat bir değişkenin tipi sonradan değiştirilemez.
 1. `var customerName = "Süleyman"`
 2. ~~`customerName = 1000`~~

Strings



- Değerler String'lerin içine kolayca yerleştirilebilir.

Strings



- Değerler String'lerin içine kolayca yerleştirilebilir.
- \()

Strings



- Değerler String'lerin içine kolayca yerleştirilebilir.
- \()
- `let balance = 10000`

Strings



- Değerler String'lerin içine kolayca yerleştirilebilir.
- `\()`
 1. `let balance = 10000`
 2. `let balanceSummary = "Bakiyeniz: \(balance) TL"`

Arrays



1. `let emptyArray = String [] ()`

Arrays



1. `let emptyArray = String [] ()`
2. `let creditCards= ["Bonus Card", "Total Card", "She Card"]`

Arrays



1. `let emptyArray = String [] ()`
2. `var creditCards= ["Bonus Card", "Total Card", "She Card"]`
3. `creditCards= [] // arrayi bosaltir`

Arrays



1. `let emptyArray = String [] ()`
2. `var creditCards= ["Bonus Card", "Total Card", "She Card"]`
3. `creditCards= [] // arrayi bosaltir`
4. `creditCards[0] = "Super Card"`

Arrays



1. `let emptyArray = String [] ()`
2. `var creditCards= ["Bonus Card", "Total Card", "She Card"]`
3. `creditCards= [] // arrayi bosaltir`
4. `creditCards[0] = "Super Card"`
5. `let cardName = creditCards [0]`

Dictionaries



```
1. var userCards:Dictionary<String,Array> = [  
2.   “user1” : [“Bonus Card”, “Total Card”],  
3.   “user2” : [“Total Card”]  
4. ]
```

Dictionaries



1. `var userCards:Dictionary<String,Array> = [`
2. `“user1” : [“Bonus Card”, “Total Card”],`
3. `“user2” : [“Total Card”]`
4. `]`
5. `userCards[“user3” : “She Card”]`
6. `let myCards = userCards[“user1”]`

Xcode & Playground Demo

