

Init Methods

- אתחול עצם ומשתניו באמצעות מתודת האתחול המובנית
- אתחול עצם ומשתניו בעזרת מתודת אתחול משלנו
- אי-אתחול

Init Methods

- מטרת מתודת האתחול היא לקבוע ערכים עבור משתני המחלקה \ מבנה, וכן לבצע פעולות נוספות הנדרשות במידת הצורך.

```
struct Point3D{  
    var x : Double  
    var y : Double  
    var z : Double  
  
    init(){  
        x = 0  
        y = 0  
        z = 0  
    }  
}
```

```
class GameObject{  
    var point : Point3D  
    var objectName : String  
    var objectImageFileName : String?  
  
    init(){  
        point = Point3D()  
        objectName = "No Name"  
    }  
}
```

```
var gameObject = GameObject()
```

Init Methods

כמו כן, חוץ ממתודת האתחול המובנית ניתן גם להגדיר מתודות אתחול משלנו.

אגב, אתם וודאי זוכרים את האתחול
המובנה של struct

נראה דוגמא

```
var gameObject1 = GameObject()  
var gameObject2 = GameObject(atPoint: Point3D(x:1 ,y:1 ,z:1))  
var gameObject3 = GameObject(atPoint: gameObject2.point, withName: "Bill")  
var gameObjecr4 = GameObject(atPoint: Point3D(x:1 ,y:2 ,z:3), withName: "John of arc-cos", andFileName: "john@2x.png")
```

Init Methods

```
class GameObject{
    var point : Point3D
    var objectName : String
    var objectImageFileName : String?

    init(){
        point = Point3D()
        objectName = "No Name"
    }

    init(atPoint p : Point3D){
        point = p
        objectName = "No Name"
    }

    init(atPoint p : Point3D, withName name : String){
        point = p
        objectName = name
    }

    init(atPoint p : Point3D, withName name : String, andFileName fileName : String){
        point = p
        objectName = name
        objectImageFileName = fileName
    }
}
```

Init Methods

```
class RightTriangle{  
    let height : Double  
    let base : Double  
    let rein : Double
```

נראה דוגמא נוספת (ללא init מובנה)

```
    let scope : Double
```

```
    var t = RightTriangle(height: 3, base: 4, rein: 5)
```

```
    let sinAlpha : Double  
    let cosAlpah : Double
```

```
    init(height : Double, base : Double, rein : Double){  
        self.height = height  
        self.base = base  
        self.rein = rein
```

תרגילון:

הוסיפו בדיקה אם הערכים חוקיים

```
        scope = (base * height) / 2.0  
        sinAlpha = sin(height/rein)  
        cosAlpah = cos(base/rein)
```

```
    }
```

Init Methods

למחלקה יכולים להיות 2 סוגי מתודות אתחול

designated initializer

אתחול ייעודי, לכל מחלקה יהיה לפחות אחד כזה.
אתחול ייעודי, תפקידו לאתחל את כל המשתנים של העצם שנוצר מהמחלקה.

עד כה, ראינו מאתחלים ייחודיים

Init Methods

סוג שני

convenience initializer

אתחול נוחות נועד כשמו לצורכי נוחות, הזנת ארגומנטים מסויימים, או אתחול מסוג מסויים לעצם מהמחלקה.

אין חובה לקיומו
עליו לקרוא לאתחול ייעוד

Init Methods

מאתחל ייחודי יכול לקרוא למאתחל ייחודי אחר על מנת שיסיים את עבודתו.

בעת ירושה ודריסה, המאתחל הייחודי, **חייב** לקרוא למימוש האתחול הייחודי במחלקת האב.

בעת הוספת מתודת מאתחל ייחודי חדש במחלקה יורשת, יש לקרוא למאתחל ייחודי אחר.

Init Methods

תהליך אתחול מוגדר מ2 שלבים.

1. כל המשתנים מאותחלים ומוצבים.

2. לאחר סיום אתחול כל המשתנים, ניתן לקרוא ערכים ממשתנים, וכן ניתן לקרוא למתודות עצמים ומחלקה במידת הצורך

Init Methods

מאחורי הקלעים

1. בדיקה שכל המשתנים מאותחלים (אופציונאליים מאותחלים לnil)
2. בדיקה שבעת ירושה יש קריאה למתודת אתחול ייעודית במחלקת האב
3. בדיקה שאתחול נוחות קורא לאתחול ייעודי
4. בדיקה שאין קריאות לערכי משתנים או למתודות עצם לפני סיום אתחול כל המשתנים

Init Methods

נראה דוגמאות מימוש

```
class Automobile{  
    var wheelsCount : UInt  
    var engineCapacity : Double  
    let licensePlateNumber : String  
  
    init(licensePlateNumber : String){  
        wheelsCount = 4  
        engineCapacity = 1000  
        self.licensePlateNumber = licensePlateNumber  
    }  
}
```

Init Methods

נראה דוגמאות מימוש

```
class Car : Automobile{
  var numberOfSeats : UInt
  var color : String?

  init(licensePlateNumber : String, seatsCount : UInt){
    numberOfSeats = seatsCount
    super.init(licensePlateNumber: licensePlateNumber)
  }

  convenience init(licensePlateNumber: String, seatsCount : UInt, color:String){
    self.init(licensePlateNumber : licensePlateNumber, seatsCount : seatsCount)
    self.color = color
  }
}
```

De-Initializer

אי-אתחול.

נקודה רגע משתחרר עצם, דוגמאות שימוש:

1. "ניקוי" תהליכים (הורדת קובץ הופרעה, נפסיק את תהליך התקשורת מול השרת)
2. שמירת מידע מסויים (משחק הופסק, נשמור את הניקוד ו\או את המצב)
3. לחילופין ביצוע פעולות (שחקן פוקר השתחרר - כספו חוזר לשולחן)

```
class FileDownloader{  
    var fileURL : String  
    init(fileURL : String){  
        self.fileURL = fileURL  
    }  
  
    func startDownload(onFinish : (Data) -> Void){  
        //start fetching data from server  
        //onFinish arg is function or block  
        //that gets the result (Data instance)  
    }  
  
    func cacnelDownload(){  
        //cancel the download  
    }  
  
    deinit{  
        //when FileDownloader being released  
        //We should stop the donwload  
        cacnelDownload()  
    }  
}
```

De-Initializer

נראה דוגמת קוד

```
class Data{  
    //...  
}
```

De-Initializer

שימוש בקוד שראינו לעיל

```
var downloader : FileDownloader? = FileDownloader(fileURL: "http://www.myWebsite/files/image.png")
downloader?.startDownload({ data in
    println(data)
})

//and after a while

downloader = nil
```

סיכום

ראינו כיצד מאתחלים מחלקה ומבנה.
יצרנו אתחול משלנו.
ניהלנו אתחול בעת ירושת מחלקות.
וקישרנו בין מתודות אתחול שונות.

כמו כן, ראינו איך לבצע תהליכים בעת שעצם הנוצר ממחלקה משתחרר.

Init Methods

שאלות?