POO în SWIFT

Curs 3
08.12.2017

Clases

Buzoianu Stefan
IOS Developer

CLASE

BENEFICII

- Prin moștenire proprietățile sunt “trecute” de la o clasă la alta
- Reference counting ARC (ce-o fi aia?) permite unei clase sa aibe cel puțin o instanță
- Proprietățile sunt definite pentru a stoca valori
- Metodele dezvoltă partea de funcționalitate
- Access level (seteaza tipul de acces asupra membrilor)
- Stările inițiale sunt setate prin initializer (init)
- Utilizarea de protocoale (ce-o fi aia?) extinde abilitățile unei clase

Tema nr.3: Studiu Reference counting ARC si protocoale

Buzoianu Stefan
IOS Developer

CLASE

EXEMPLU

```
class Human {
    var name = ""
    var height = 0.0 // stored property
    var description: [String: String] {
        return ["name": self.name, "height":"\(self.height)"]
    } // computed property
}

var person = Human() // instanță a clasei Om
person.name = "Mihai"
person.height = 1.73

print(person.description["name"]!)  // se  afisează Mihai
print(person.description["height"]!) // se afisează 1.73
```

Buzoianu Stefan
IOS Developer

Instance Methods

- oferă acces la proprietățile definite în clasă
- oferă și definește funcționalitate pentru aceste proprietăți

```
class calculations {
   let a: Int // nu le-am atribuit valoare pentru că ele vor fi definite prin initializer
   let b: Int
   let res: Int
   init (a: Int, b: Int) {
       self.a = a
       self.b = b
       res = a + b }
    func tot (c: Int) -> Int {  // vom discuta mai târziu despre funcții
        return res - c
    }
    func printCurrentResult ()  {
        print("Result is: " + tot(c: 50) ) // afiseaza 850
        print("Result is: \( tot(c: 20) )") // afiseaza 820
} }

 let calculatedObject = calculations(a: 600, b: 300)
 calculatedObject.printCurrentResult() // accesarea unei metode se face prin “.”
```

Buzoianu Stefan
IOS Developer

Instance Variables

- stored property - reține o valoare constantă sau o variablă într-o instanță
- computed property - calculează valoarea și apoi o returneaza / reține într-o instanță

```
class Sample {
  var no1 = 0.0, no2 = 0.0
  var length = 300.0, breadth = 150.0 // stored property
  var middle: (Double, Double) { // computed property
      get {
            return (length / 2, breadth / 2)
      }
      set(axis) {
            no1 = axis.0 - (length / 2)
            no2 = axis.1 - (breadth / 2)
      }
  }
}
var result = Sample()
print(result.middle) // afisează (150.0, 75.0)
result.middle = (0.0, 10.0)
print(result.no1) // afisează -150.0
print(result.no2) // afisează -65.0
```

Buzoianu Stefan
IOS Developer

Moștenirea - Inheritance

- subclasă: când o clasă moștenește proprietăți și metode dintr-o altă clasă.
- superclasă: o clasă ce conține proprietăți și metode moștenite de o altă clasă.

```
class Rectangle { // superclasă
    var width = 42
    var height = 10
    var area: String { // computed property
        return "area \(self.width*self.height)px"
    }
}
class Square: Rectangle { // subclasa Square - moștenește clasa Rectangle
    var color = "Blue"
    override var area: String { // proprietate supra-scrisă
        return super.area + " override a rectangle and its color is \(self.color)"
    }
}

let squ = Square()
squ.width = 10
squ.color = "Pink"
print("Square with \(squ.area)") // Square with area 100px override a rectangle and its color is Pink
```

Buzoianu Stefan
IOS Developer

Access levels

- private: disponibil doar la nivel de obiect, nu este accesibil la exterior si nici mostenit
- public: este accesibil atat la nivel de obiect, de subclase cat si din exterior

```
class Rectangle {
    var width = 42; var height = 10
    private var angle = 90
    private var area: String {
        return "area \(self.width*self.height)px"
    }
    public func getArea() -> String { return self.area }
}
class Square: Rectangle {
    var color = "Blue"
    override var area: String { // Eroare. area este privat, nu pot face override
        return super.getArea() + " override a rectangle and its color is \(self.color)"
                // super.area nu ar fi functionat, area este privat in clasa Rectangle
    }
}

let squ = Square()
squ.width = 10
print("Square with \(squ.area)") // Fara override: Succes. area este public si definit in
clasa Square, nu este mostenit din Rectangle
squ.angle = 30 // Eroare. angle este privat in clasa Rectangle
```

Buzoianu Stefan
IOS Developer

PLAYGROUND

https://github.com/sbuzoianu/CS3/MyPlayground.playground.zip

Buzoianu Stefan
IOS Developer