

Apple Swift kurzus

2. gyakorlat

Változók, konstansok:

- **var**: **Type** - Type típusú változó
pl.: **var integer: Int = 10**
var double: Double = 10.0
- **let** - konstans, mely csak inicializálható
pl.: **let integer = 10**
let double = 10.0
- inferencia: az objektumok típusára való következtetés

Osztály:

- **class** Osztálynév { }

Objektum:

- minden objektum
- osztályból kerül példányosításra
pl.: **let person = Person()**
- létrehozáskor az **init()** függvény kerül meghívásra
- ha egy objektum **nil** értéket kap, a **deinit()** függvény kerül meghívásra

Optional:

- objektum, melynek vagy van értéke vagy **nil**
- jelölés: ? és !
pl.: **var Person: Person?**

Optional chaining:

- **if let** együttes használata
- ha a jobb oldalon levő Optional nem **nil**, igaz értékkel tér vissza
pl.:

```
if let personPhone = person.cellphone?.type { //Optional Chaining
    print(personPhone)
} else {
    print("no cellphone found...")
}
```

Automatic Reference Counting:

- memóriamenedzsment
- minden létrehozott objektum **strong** referenciával bír a heap-en
- egy objektum addig él, amíg szükség van rá
- a nem használt objektumok törlésre kerülnek - **deinit()**
- ha egy objektum **nil** értéket vesz fel (Optional), akkor törölhető a heap-ről

Strong Reference Cycle:

- az objektumok egymásra hivatkozása **strong** referenciával valósul meg
- amíg egy objektumra **strong** referenciával hivatkoznak, nem kerül törlésre
- **probléma**: két objektum egymásra hivatkozik
- **megoldás**: **strong** referencia helyett alkalmazzunk **weak/unowned** referenciát
- a **strong** referencia megszűnésekor az objektumok törlődnek
pl.: lásd: *02_Swift_ARC.swift* és *02_Swift_ARC2.swift*

Weak vs. unowned:

- **weak** változók tetszőleges típusúak lehetnek (Optional is!)
- **unowned** változóknak mindig rendelkezniük kell értékkel (nem lehet Optional!)
- egyaránt megszüntethető mindkét referencia, ha nincs **strong** a heap-en

Typealias:

- típusokhoz megadhatóak helyettesítő nevek (felhasználó által definiált típushoz is!)

```
pl.:  typealias D = Double  
var double: D = 10.0
```