

IOS - Praktikum

Wintersemester 2016 / 2017
Übungsblatt 1

Arantxa Ye

IOS-Praktikum

Aufgabe 2b)

Aufgabe 2: (H)

Objective-C in Swift und umgekehrt

b. Legen Sie ein Swift-Projekt an. Verwenden Sie als Elternklasse eine Objective-C-Klasse und lassen davon eine Swift-Kindklasse erben. Rufen Sie eine Methoden der Elternklasse aus Swift heraus auf.

Vorgehensweise

- 1) .h-file in Objective-C
- 2) .m-file in Objective-C
- 3) Swift-Klasse

Person.h

```
//  
// Person.h  
// iOS_Blatt1Aufg2b  
//  
// Created by Arantxa Ye on 01.11.16.  
// Copyright © 2016 Arantxa Ye. All rights reserved.  
//  
  
#import <Foundation/Foundation.h>  
  
// Alles in der h File ist von außen sichtbar und verwendbar  
@interface Person : NSObject  
  
//Klassenvariablen  
@property NSString * _Nullable name;  
@property int age;  
@property double height;  
  
//Konstruktoren und Methoden  
//Rückgabotyp: instancetype  
//Nonnull: gibt Instanz von Klasse Person zurück, die nicht null sein kann  
- (instancetype _Nonnull)initWithAge:(int)age name:(NSString * _Nullable)name andHeight:(double)height;  
  
- (instancetype _Nonnull)init;  
  
- (NSString* _Nonnull)greet;  
  
@end
```

Person.m

```
#import "Person.h"
@implementation Person

//Konstruktor
- (instancetype)init{

    self = [super init];           //ruft Konstruktor von der Super-Klasse auf (Elternklasse: NSObject)

    if(self){
        self.age = 23;             //weist den Klassenvariablen feste Werte zu
        self.name = @"Arantxa";
        self.height = 1.80;
    }
    return self;
}

//Konstruktor mit 3 Argumente: age, name, height
- (instancetype)initWithAge:(int)age name:(NSString*)name andHeight:(double)height{

    self = [super init];           //ruft Konstruktor von der Super-Klasse auf (Elternklasse: NSObject) (wie oben)

    if(self){
        self.age = age;            //if überprüft ob Konstruktoraufruf funktioniert hat
        self.name = name;          //weist den Klassenvariablen die Werte der Argumente zu
        self.height = height;
    }
    return self;                   //gibt das erzeugte Objekt zurück
}

//Methode
- (NSString*)greet{
    return @"Hello";
}

@end
```

Student.swift

```
import UIKit

// Klasse Student erbt von Person
class Student: Person {

    //Variable
    var matrikelNumber: Int = -1

    //eigene init-Funktion für Student, überschreibt die init-Funktion der Eltern-Klasse
    override init(){
        super.init()           //ruft Konstruktor aus der Eltern-Klasse Person auf
        self.matrikelNumber = 11205431 //weist Klassenvariable festen Wert zu
    }

    //erweitert den Konstruktor aus der Eltern-Klasse Person um Argument matrikelNumber
    init(age: Int32, name: String?, height: Double, andMatrikelNumber matrikelNumber: Int) {
        super.init(age: age, name: name, andHeight: height) //ruft Konstruktor aus Eltern-Klasse mit Argumenten auf
        self.matrikelNumber = matrikelNumber                //weist der Klassenvariable den Wert des Arguments zu
    }

    //Methode
    func greetFriend() -> String {

        var greeting = super.greet() //ruft Methode greet aus Eltern-Klasse auf und übergibt Wert Variable greeting
        greeting += " Friend!"        //fügt der Variable greeting String hinzu
        return greeting               //gibt greeting zurück
    }
}
```