Министерство образования

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Отчет по лабораторной работе №3

«Использование языка программирования Swift: структуры и классы, методы»

Выполнил: Наджмиддинзоде У.Х.

Студент группы 310902

Проверил: Усенко Ф. В.

Минск 2024

Цель работы – Выполнить разработку приложения с использованием языка программирования Swift: структуры и классы, методы.

* 1. Создайте класс ПЕРСОНА с методами, позволяющими вывести на экран информацию о персоне, а также определить ее возраст (в текущем году). Создайте дочерние классы: АБИТУРИЕНТ (фамилия, дата рождения, факультет), СТУДЕНТ (фамилия, дата рождения, факультет, курс), ПРЕПОДАВАТЕЛЬ (фамилия, дата рождения, факультет, должность, стаж), со своими методами вывода информации на экран и определения возраста. Создайте список из п персон, выведите полную информацию из базы на экран, а также организуйте поиск персон, чей возраст попадает в заданный диапазон.

Листинг кода:

import Foundation

public class Person {

public var surname: String

public var birthDate: Date

public init(surname: String, birthDate: Date) {

self.surname = surname

self.birthDate = birthDate

}

public func calculateAge() -> Int {

let calendar = Calendar.current

let currentDate = Date()

let ageComponents = calendar.dateComponents([.year], from: birthDate, to: currentDate)

return ageComponents.year ?? 0

}

public func printInfo() {

print("Фамилия: \(surname), Дата рождения: \(birthDate), Возраст: \(calculateAge())")

}

}

public class Abiturient: Person {

public var faculty: String

public init(surname: String, birthDate: Date, faculty: String) {

self.faculty = faculty

super.init(surname: surname, birthDate: birthDate)

}

override public func printInfo() {

super.printInfo()

print("Факультет: \(faculty)")

}

}

public class Student: Abiturient {

public var course: Int

public init(surname: String, birthDate: Date, faculty: String, course: Int) {

self.course = course

super.init(surname: surname, birthDate: birthDate, faculty: faculty)

}

override public func printInfo() {

super.printInfo()

print("Курс: \(course)")

}

}

public class Teacher: Person {

public var faculty: String

public var position: String

public var experience: Int

public init(surname: String, birthDate: Date, faculty: String, position: String, experience: Int) {

self.faculty = faculty

self.position = position

self.experience = experience

super.init(surname: surname, birthDate: birthDate)

}

override public func printInfo() {

super.printInfo()

print("Факультет: \(faculty), Должность: \(position), Стаж: \(experience) лет")

}

}

import Foundation

import PersonModule

public struct Search {

public static func searchByAgeRange(persons: [Person], minAge: Int, maxAge: Int) -> [Person] {

return persons.filter { person in

let age = person.calculateAge()

return age >= minAge && age <= maxAge

}

}

}

import Foundation

public struct Validation {

public static func validateNonEmptyString(\_ input: String) -> String? {

return input.isEmpty ? nil : input

}

public static func validateBirthDate(\_ date: Date) -> Bool {

return date <= Date()

}

}

import Foundation

import PersonModule

import ValidationModule

import SearchModule

func test() {

let dateFormatter = DateFormatter()

dateFormatter.dateFormat = "yyyy-MM-dd"

let person1 = Abiturient(surname: "Иванов", birthDate: dateFormatter.date(from: "2000-05-10")!, faculty: "ФКСиС")

let person2 = Student(surname: "Петров", birthDate: dateFormatter.date(from: "1999-04-12")!, faculty: "ФКП", course: 3)

let person3 = Teacher(surname: "Сидоров", birthDate: dateFormatter.date(from: "1980-07-30")!, faculty: "ФИТУ", position: "Доцент", experience: 10)

let persons: [Person] = [person1, person2, person3]

for person in persons {

person.printInfo()

}

print("\nПоиск людей в возрасте от 20 до 25 лет:")

let foundPersons = Search.searchByAgeRange(persons: persons, minAge: 20, maxAge: 25)

for person in foundPersons {

person.printInfo()

}

}

test()

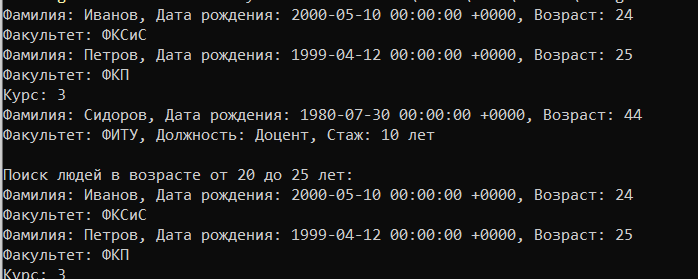


Рисунок 1 – Результат выполнения программы

Вывод: в ходе выполнения лабораторной работы я разработал приложение на языке программирования Swift, используя структуры, классы и методы