Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Дисциплина: Современные языки программирования

Отчёт

По лабораторной работе №3

**Использование языка программирования Swift: структуры и классы, методы**

Вариант 13

Выполнила:

студентка группы 310101

Сердюк У. В.

Проверил:

Усенко Ф. В.

Минск 2024

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

Экземпляр класса при инициализации принимает аргументы: имя, должность и стаж работы сотрудника, метод *printInfo()* выводит информацию о сотруднике в формате: «Имя: Василий Должность: Системный администратор Стаж: 3 года» При выводе стажа нужно учитывать, что «года» должно заменяться на «лет» или «год» в зависимости от числа.

В данном задании реализован класс *Employee*, представляющий информацию о сотруднике. Класс включает три свойства: имя, должность и стаж работы, которые задаются при создании экземпляра через инициализатор. Основной функционал обеспечивается методом *printInfo*, выводящим подробную информацию о сотруднике с учётом правильного склонения слова "год" для стажа. Логика выбора подходящего склонения реализована в приватном методе *formatYears*, который анализирует последние цифры числа. В примерах демонстрируется создание объектов класса *Employee* с разными данными и использование метода *printInfo* для их корректного отображения.

Результат выполнения работы программы представлен на рисунке 1.

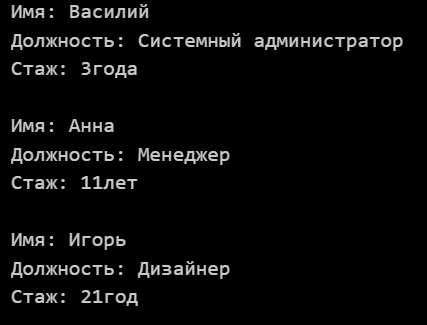


Рисунок *1* – Результат выполнения работы программы 1

**Вывод:** В ходе лабораторной работы разработано приложение на языке программирования Swift, выполнены задания на создание класса *Employee* для реализации методов в нём.

**Ответы на вопросы**

3. Что такое структура?

Классы и структуры являются универсальными и гибкими конструкциями, которые станут строительными блоками для кода вашей программы. Для добавления функциональности в классах и структурах можно объявить свойства и методы, применив тот же синтаксис, как и при объявлении констант, переменных и функций.

В отличие от других языков программирования, Swift не требует создавать отдельные файлы для интерфейсов и реализаций пользовательских классов и структур. В Swift, вы объявляете структуру или класс в одном файле, и внешний интерфейс автоматически становится доступным для использования в другом коде.

Экземпляр *класса* традиционно называют *объектом*. Тем не менее, классы и структуры в Swift гораздо ближе по функциональности, чем в других языках, и многое в этой главе описывает функциональность, которую можно применить к экземплярам *и* класса, *и* структуры. В связи с этим, употребляется более общий термин – *экземпляр*.

7. Ключевое слово *self.*

*Self* − это ссылка на текущий экземпляр класса, структуры или перечисления. Используется для доступа к свойствам или методам текущего экземпляра. Также применяется для разрешения конфликтов, если имя параметра метода совпадает с именем свойства экземпляра.

Пример кода:

struct Point {

var x = 0.0, y = 0.0

func isToTheRightOf(x: Double) -> Bool {

return self.x > x

}

}

В данном примере *self.x* используется, чтобы отличить свойство экземпляра *x* от параметра метода *x*.