

ДИСЦИПЛИНА Разработка приложений на языке Котлин
(полное наименование дисциплины без сокращений)

ИНСТИТУТ информационных технологий

КАФЕДРА информационных технологий в атомной энергетике
(полное наименование кафедры)

ВИД УЧЕБНОГО
МАТЕРИАЛА Практическая работа
(в соответствии с пп 1-11)

ПРЕПОДАВАТЕЛЬ Золотухин Святослав Александрович
(фамилия, имя, отчество)

СЕМЕСТР 5 семестр 2025 – 2026 учебный год
(указать семестр обучения, учебный год)

ЗАДАНИЕ ДЛЯ 5 ПРАКТИЧЕСКОЙ РАБОТЫ ПО ДИСЦИПЛИНЕ «РАЗРАБОТКА ПРИЛОЖЕНИЙ НА ЯЗЫКЕ КОТЛИН»

Задание 1:

Создайте класс с именем **Cat**.

В классе должны быть следующие приватные функции:

- **rest()** - выводит на экран «Sleep»
- **voice()** - выводит на экран «Meow»
- **feed()** - выводит на экран «Eat»

Также необходимо реализовать одну публичную функцию:

- **randomAction()** - случайным образом вызывает одну из закрытых функций.

Задание 2:

Создайте класс **Student**.

В этом классе должны быть следующие приватные свойства:

- **var firstName: String** - имя студента. Геттер для этого свойства должен возвращать значение с первой заглавной буквой, а сеттер - убирать лишние пробелы при установке значения.
- **var lastName: String** - фамилия студента. Геттер для этого свойства также должен возвращать значение с первой заглавной буквой, а сеттер - убирать лишние пробелы при установке значения.
- **var scores: IntArray** - массив из последних десяти оценок студента

А также следующие публичные методы:

- Методы для получения и изменения свойства **firstName** с реализацией описанной логики
- Методы для получения и изменения свойства **lastName** с аналогичной логикой

- Методы для получения и изменения массива **scores**
- Метод, который добавляет новую оценку в массив **scores**, удаляя первую оценку и добавляя новую в конец (например: 2, 5, 4 → добавляем 3 → результат: 5, 4, 3). Реализовать именно через массив
- Метод, возвращающий среднюю оценку студента (считается как среднее арифметическое всех элементов массива **scores**)

Задание 3:

Создайте класс **StudentService**.

В классе должны быть следующие публичные функции:

- **findBestStudent()** - принимает на вход массив объектов типа **Student** (из предыдущего задания), возвращает студента с наивысшей средней оценкой. Если таких студентов несколько, вернуть любого.
- **sortStudentsByLastName()** - принимает массив объектов **Student**, сортирует его по фамилиям в алфавитном порядке и возвращает отсортированный массив.

Требования к отчету:

Титульный лист, оглавление, текст задачи, выполненные задания (краткое описание кода реализации каждого задания с указанием листинга кода и скриншотов работы каждой программы), вывод (что было сделано в ходе выполнения работы), список использованных источников.

Оформление работ обязательно должно отвечать требованиям СМКО МИРЭА.

При защите работы необходимо ответить на несколько контрольных вопросов.

Литература:

- 1) Лекционный материал
- 2) <https://kotlinlang.org/docs/home.html>