

← AP1-Kt-T01

Reviews



✓ Subscribe

Intensive

[Show all](#)

AP1-Kt-T01

AP1-Kt-T02

AP1-Kt-P01

AP1-Kt-T03

Task

Project 01 — Kotlin_Bootcamp

Резюме: в этом проекте ты познакомишься с основными функциями языка **Kotlin**.

💡 [Нажми сюда](#), чтобы поделиться с нами обратной связью на этот проект. Это анонимно и поможет нашей команде сделать обучение лучше. Рекомендуем заполнить опрос сразу после выполнения проекта.

Содержание

1. [Chapter I](#)
 - [Инструкция](#)
2. [Chapter II](#)

- [Общая информация](#)

3. [Chapter III](#)

- [Проект: Умные утилиты](#)
- [Задание 0. Создание проекта](#)
- [Задание 1. Нахождение периметра треугольника](#)
- [Задание 2. Нахождение часов, минут и секунд](#)
- [Задание 3. Поиск числа Фибоначчи](#)
- [Задание 4. Поиск среднего арифметического отрицательных чисел](#)
- [Задание 5. Поиск чисел, у которых совпадает первая и последняя цифра](#)
- [Задание 6. Сортировка коллекции выбором](#)
- [Задание 7. Поиск максимального и минимального значений](#)
- [Задание 8. Упорядоченная последовательность по возрастанию](#)
- [Задание 9. Фильтрация строк](#)
- [Задание 10. Поиск имен совершеннолетних пользователей](#)
- [Задание 11. Поиск наибольшего общего делителя двух целых чисел](#)

Chapter I

Инструкция

1. На протяжении всего курса тебя будет сопровождать чувство неопределенности и острого дефицита информации — это нормально. Не забывай, что информация в репозитории и Google всегда с тобой. Как и пиры, и Rocket.Chat. Общайся. Ищи. Опирайся на здравый смысл. Не бойся ошибиться.
2. Будь внимателен к источникам информации. Проверяй. Думай. Анализируй. Сравнивай.
3. Внимательно читай задания. Перечитай несколько раз.
4. Читать примеры тоже лучше внимательно. В них может быть что-то, что не указано в явном виде в самом задании.
5. Тебе могут встретиться несоответствия, когда что-то новое в условиях задачи или примере противоречит уже известному. Если встретилось такое — попробуй разобраться. Если не получилось — запиши вопрос в открытые вопросы и выясни в процессе работы. Не оставляй открытые вопросы неразрешенными.
6. Если задание кажется непонятным или невыполнимым — так только кажется. Попробуй его декомпозировать. Скорее всего, отдельные части станут понятными.
7. На пути тебе встретятся самые разные задания. Те, что помечены звездочкой (*) — подходят для более дотошных. Они повышенной сложности и необязательны к выполнению. Но если ты их сделаешь, то получишь дополнительный опыт и знания.

8. Не пытайся обмануть систему и окружающих. В первую очередь ты обманешь себя.
9. Есть вопрос? Спроси соседа справа. Если это не помогло — соседа слева.
10. Когда пользуешься помощью — всегда разбирайся до конца: почему, как и зачем. Иначе помощь не будет иметь смысла.
11. Всегда делай push только в ветку develop! Ветка master будет проигнорирована. Работай в директории src.
12. В твоей директории не должно быть иных файлов, кроме тех, что обозначены в заданиях.

Chapter II

Общая информация

Kotlin — это быстрый, безопасный и надежный язык программирования для всего: от мобильных приложений и корпоративного ПО до приложений для работы с большими данными и серверных технологий.

В 2010 году программист питерской ИТ-компании JetBrains Андрей Бреслав вместе с коллегами начал разработку нового языка. На тот момент организация существовала 10 лет, а их главный продукт — IntelliJ IDEA, полностью написанный на Java, — был уже «очень тяжел». JetBrains не устраивал этот язык программирования, и было принято решение найти ему альтернативу. Поиски не увенчались успехом: ни один инструмент разработки не соответствовал желаемым требованиям.

Команда проекта, состоявшая из опытных специалистов, решила пойти другим путем и создать собственный язык, который бы отвечал всем потребностям компании и был способен решать широкий спектр задач.

Саму историю создания Kotlin разработчики называют сказочной, считая, что «мы делали то, что хотели». Главной целью команды было обратить на себя внимание Google и получить их поддержку, что удалось в 2016 году. Вскоре Android объявила об официальной поддержке Kotlin для разработки собственных приложений, а в 2019 году Google объявил этот язык приоритетным для создания мобильных приложений под свою мобильную ОС.

Главные преимущества **Kotlin**:

1. Портативность: программы на Kotlin могут выполняться на любой машине, которая поддерживает виртуальную машину Java (JVM).
2. Объектно-ориентированность: Kotlin является полностью объектно-ориентированным языком, что позволяет создавать модульные и масштабируемые программы. Объектно-ориентированный подход способствует повторному использованию кода и облегчает его сопровождение.

3. Безопасность: Kotlin предоставляет механизмы безопасности, такие как контроль доступа и проверка типов, что позволяет создавать более надежные приложения. Также виртуальная машина Java (JVM) обеспечивает изоляцию программ от операционной системы.
4. Многозадачность: Kotlin поддерживает многозадачность, что позволяет разрабатывать эффективные и отзывчивые приложения, способные выполнять несколько задач одновременно.
5. Большая стандартная библиотека: Kotlin поставляется с обширной стандартной библиотекой, включающей в себя классы и методы для решения широкого спектра задач, что упрощает разработку.
6. Корутины и асинхронное выполнение кода: Kotlin предоставляет разработчику такой инструмент, как Kotlin Coroutines — мощный механизм для написания асинхронного и многопоточного кода.
7. Простота использования: Kotlin обладает относительно простым синтаксисом, что упрощает разработку и понимание кода, особенно для новичков.

Темы для изучения:

- Точка входа в программу, структура программы;
- Компиляция/интерпретация программы;
- Управляющие структуры языка (следование, ветвление, повторение);
- Типы данных (назначение, размер, представление в памяти, операции);
- Организация ввода-вывода (stdin-stdout);
- Автоматическая, статическая и динамическая память, сборщик мусора;
- Сложные структуры данных (шаблоны и дженерики).

Chapter III

Проект: Умные утилиты

Рассматриваются основные функции языка при помощи проекта, который представляет собой набор инструментов для решения разных типов задач.

Внимание! Каждую задачу оформляй в качестве отдельного проекта.

Например, `T01/src/exercise0`, `T01/src/exercise1`, ..., `T01/src/exerciseN-1`, где N количество задач. Если предыдущее задание необходимо для следующего, просто скопируй предыдущий проект в директорию следующего и продолжай разработку в нем.

Задание 0. Создание проекта

В IntelliJ Idea создай новый проект:

- Выбери язык Kotlin.
- Выбери систему сборки Gradle.

- Выбери JDK 18, если такого нет, то загрузи любой JDK 18 версии.
- Выбери для Gradle DSL — Kotlin.

Задание 1. Нахождение периметра треугольника

Разработай математический модуль, который определяет периметр треугольника.

- Программа считывает координаты вершин треугольника.
- Программа работает с вещественными числами.
- Программа вычисляет и выводит периметр треугольника, если из введенных вершин образуется треугольник.
- Если из введенных вершин не образуется треугольник, то программа выводит: «It's not a triangle».
- Программа не завершается с ошибкой при некорректных входных данных. Она выводит: «Couldn't parse a number. Please, try again» и повторяет попытку ввода.
- Точность: 3 знака после запятой.

Входные данные	Выходные данные
1.0 2.0 2.0 1.0 5.0 5.0	Perimeter: 11.414
2.0 1.0 2.0 1.0 2.0 1.0	It's not a triangle
2.0 1.0 2.0 1.0 3.0 1.0	It's not a triangle

Задание 2. Нахождение часов, минут и секунд

Разработай математический модуль, который переводит секунды к формату hh:mm:ss, то есть находит количество часов, минут и секунд.

- Программа считывает секунды.
- Программа работает с целыми числами.

- Программа вычисляет и выводит количество часов, минут и секунд в формате hh:mm:ss.
- Если введено количество секунд меньше 0, то должно выводиться: «Incorrect time».
- Программа не завершается с ошибкой при некорректных входных данных. Она выводит: «Couldn't parse a number. Please, try again» и повторяет попытку ввода.
- Должна быть следующая структура программы:
 - метод ввода;
 - метод нахождения часов, минут и секунд;
 - метод вывода;
 - все упомянутые выше методы должны вызываться из main.

Входные данные	Выходные данные
3599	00:59:59
3601	01:00:01
-100	Incorrect time

Задание 3. Поиск числа Фибоначчи

Разработай математический модуль, который находит n число Фибоначчи.

- Программа считывает порядковый номер числа Фибоначчи.
- Программа работает с целыми числами.
- Предусмотрен выход за пределы памяти.
- Для нахождения числа Фибоначчи используй рекурсивный подход.
- Программа не завершается с ошибкой при некорректных входных данных. Она выводит: «Couldn't parse a number. Please, try again» и повторяет попытку ввода.

Входные данные	Выходные данные
10	55
100000000	Too large n

Задание 4. Поиск среднего арифметического отрицательных чисел

Разработай математический модуль, который находит среднее арифметическое отрицательных чисел.

- Программа считывает количество чисел.

- Программа считывает каждое новое число в коллекцию.
- Программа работает с целыми числами.
- Если отрицательные числа есть, то программа выводит их среднее арифметическое, иначе выводит: «There are no negative elements».
- Если введено отрицательное количество чисел или ноль, то должно выводиться: «Input error. Size <= 0».
- Программа не завершается с ошибкой при некорректных входных данных. Она выводит: «Couldn't parse a number. Please, try again» и повторяет попытку ввода.
- Используй цикл с предусловием for.

Входные данные	Выходные данные
4 1 2 3 4	There are no negative elements
-1	Input error. Size <= 0
4 1 -2 3 -4	-3

Задание 5. Поиск чисел, у которых совпадает первая и последняя цифра

Разработай математический модуль, ищущий числа, у которых совпадает первая и последняя цифра.

- Программа считывает количество чисел.
- Программа считывает каждое новое число в коллекцию.
- Программа работает с целыми числами.
- Если есть числа, у которых совпадает первая и последняя цифра, то должны выводиться они, иначе пусть выводится: «There are no such elements».
- Числа, у которых совпадает первая и последняя цифра, сохрани в отдельный массив.
- Если введено отрицательное количество чисел или ноль, то программа должна выводить: «Input error. Size <= 0».
- Программа не завершается с ошибкой при некорректных входных данных. Она выводит: «Couldn't parse a number. Please, try again» и повторяет попытку ввода.
- Используй цикл с предусловием while.
- Должен быть выделен отдельный метод для определения числа, у которого совпадает первая и последняя цифра.

Входные данные	Выходные данные
4 100 200 300 400	There are no such elements
-1	Input error. Size <= 0
5 1 202 300 200005 301213	1 202 301213

Задание 6. Сортировка коллекции выбором

Разработай математический модуль, который производит сортировку коллекции выбором по возрастанию.

- Программа считывает количество чисел.
- Программа считывает каждое новое число в коллекцию.
- Программа работает с вещественными числами.
- Программа выводит отсортированную коллекцию выбором по возрастанию.
- Если введено отрицательное количество чисел или ноль, то программа выводит: «Input error. Size <= 0»
- Программа не завершается с ошибкой при некорректных входных данных. Она выводит: «Couldn't parse a number. Please, try again» и повторяет попытку ввода.
- Должен быть выделен отдельный метод для сортировки выбором коллекции по возрастанию.
- Нельзя использовать библиотечные реализации, нужно написать свою.

Входные данные	Выходные данные
4 100.0 50.0 60.0 10.0	10.0 50.0 60.0 100.0
-1	Input error. Size <= 0

Задание 7. Поиск максимального и минимального значений

Разработай математический модуль, который производит поиск максимального и минимального значений в коллекции.

- Программа считывает путь до файла.
- Программа считывает из файла количество чисел.
- Программа считывает каждое новое число из файла в коллекцию, пока не превысит количество чисел либо не достигнет конца файла.

- Программа работает с вещественными числами.
- Программа выводит считанное количество чисел и сами числа.
- Программа сохраняет в файл «result.txt» найденные минимум и максимум и выводит в консоль сообщение: «Saving min and max values in file».
- Если считалось чисел меньше, чем указано в количестве, программа выводит: «Input error. Insufficient number of elements».
- Если файла не существует, то должно выводиться: «Input error. File isn't exist».
- Если отрицательное количество чисел или ноль, то должно выводиться: «Input error. Size <= 0».
- Программа не завершается с ошибкой при некорректных входных данных. Она пропускает ошибочный ввод и переходит к следующему считыванию значения.

file1.txt
4 100.0 50.0 60.0 10.0

file2.txt
0 100.0 50.0 60.0 10.0

file3.txt
10 100.0 50.0 60.0 10.0

file4.txt
5 20.0 50.0 f 60.0 g 10.0 1.0

Входные данные	Выходные данные в консоль	Выходные данные в result.txt
file1.txt	4 100.0 50.0 60.0 10.0 Saving min and max values in file	10.0 100.0
file2.txt	Input error. Size <= 0	
file3.txt	Input error. Insufficient number of elements	

Входные данные	Выходные данные в консоль	Выходные данные в result.txt
file4.txt	5 20.0 50.0 60.0 10.0 1.0 Saving min and max values in file	1.0 60.0
fileIsNotExist.txt	Input error. File isn't exist	

Задание 8. Упорядоченная последовательность по возрастанию

Разработай математический модуль, который определяет, является ли последовательность упорядоченной по возрастанию.

- Программа считывает каждое новое число.
- Программа работает с целыми числами.
- Нельзя использовать коллекции.
- Программа должна определить, является ли последовательность упорядоченной по возрастанию. В случае отрицательного ответа определи порядковый номер первого числа, нарушающего такую упорядоченность, и выведи: «The sequence is not ordered from the ordinal number of the number (порядковый номер)».
- Программа завершает проверку введенной последовательности при некорректном вводе и выводит: «The sequence is ordered in ascending order», если было введено хотя бы одно число.
- Если не ввели ни одного числа, то должно выводиться: «Input error».

Входные данные	Выходные данные
1 2 3 5 4	The sequence is not ordered from the ordinal number of the number 4
a	Input error
10 20 50 80 90 g	The sequence is ordered in ascending order

Задание 9. Фильтрация строк

Разработай модуль, который фильтрует список строк по подстроке.

- Программа считывает количество строк.
- Программа считывает каждую новую строку в список.
- После считывания всех строк программе на вход подается подстрока для фильтрации.

- Программа должна вывести список строк, в которых присутствует введенная подстрока.
- Нельзя использовать `kotlin.streams`.
- Должна быть своя реализация метода фильтрации.

Входные данные	Выходные данные
4 First car Second door Third message Fourth wood oo	Second door, Fourth wood
2 First car Second door kek	

Задание 10. Поиск имен совершеннолетних пользователей

Разработай модуль, который ищет имена совершеннолетних пользователей.

- Создай класс `User` с двумя полями: строка с именем пользователя, целочисленный возраст.
- Класс `User` должен быть в отдельном файле.
- Программа считывает количество пользователей.
- Каждый считанный пользователь добавляется в список.
- Если ввели отрицательный или нулевой возраст, то программа выводит: «Incorrect input. Age <= 0» и переходит к следующему вводу.
- Программа не завершается с ошибкой при некорректных входных данных. Она выводит: «Couldn't parse a number. Please, try again» и повторяет попытку ввода.
- Программа должна вывести имена совершеннолетних пользователей.
- Операции поиска и вывода имен совершеннолетних пользователей должны выполняться при помощи `Sequences`.

Входные данные	Выходные данные
3 Name1 16 Name2 19 Name3 18	Name2, Name3

Входные данные	Выходные данные
3 Name1 16 Name2 14 Name3 13	
4 Name1 -2 Name2 23 Name3 13 Name4 24	Incorrect input. Age <= 0 Name2, Name4

Задание 11. Поиск наибольшего общего делителя двух целых чисел

Разработай модуль, который вычисляет наибольший общий делитель двух целых чисел с помощью функции расширения типа данных Int.

- Программа считывает два числа.
- Программа вычисляет наибольший общий делитель введенных чисел.
- Программа должна корректно работать при вводе отрицательных чисел.
- Поиск наибольшего общего делителя должен быть представлен, как функция расширения типа данных Int.
- Программа не завершается с ошибкой при некорректных входных данных. Она выводит: «Couldn't parse a number. Please, try again» и повторяет попытку ввода.

Входные данные	Выходные данные
0 0	0
26 39	13
-3 -6	3
-6 3	3
3 5	1

