|  |
| --- |
|  |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |
| Институт информационных технологий |
| Кафедра инструментального и прикладного программного обеспечения |

|  |  |
| --- | --- |
|  | |
| **ОТЧЕТ ПО ПРАКТИЧЕСКИМ РАБОТАМ № 1 — 3** | |
| **по дисциплине** | |
| **«Шаблоны программных платформ языка Джава»** | |
| Выполнил студент группы **ИКБО-30-22** | ***Сенькевич Г.Д.*** |
| Принял ассистент кафедры ИиППО | ***Ермаков С.Р.*** |

|  |  |  |
| --- | --- | --- |
| Практические работы выполнены | « » 2024 г. |  |
| «Зачтено» | « » 2024 г. |  |

Москва 2024

**Практическая работа №1**

1. **Постановка задачи и персональный вариант**

Тема: Встроенные функциональные интерфейсы Java. Возможности Java 8. Лямбда-выражения. Области действия, замыкания. Предикаты. Функции. Компараторы.

Постановка задачи: Имплементировать интерфейс Comparator, сравнивающий двух студентов по набранным за семестр баллов.

1. **Код и пояснение к коду**

Код: <https://github.com/gavril-s/mirea-4/tree/master/java/pract/src/task1/opt6>

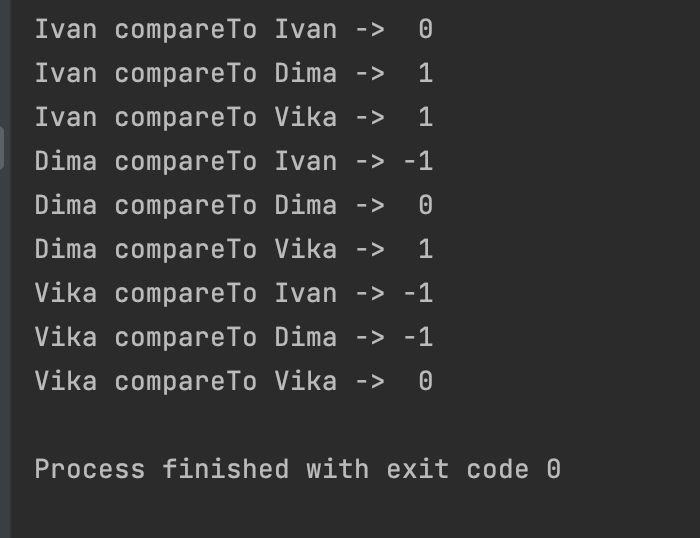
Пояснения:

Класс Student – простая реализация студента, содержащая только имя и оценку.

Класс StudentComparator – класс, реализующий интерфейс Compare<Student>.

Класс Main проводит тестирование работы класса StudentComparator.

1. **Результат работы программы**

****

**Практическая работа №2**

1. **Постановка задачи и персональный вариант**

Тема: Работа со Stream API в Java 8.

Постановка задачи: В ходе выполнения практической работы должно быть реализовано:

1) класс Human (int age, String firstName, String lastName, LocalDate birthDate, int weight);

2) приложение, которое создает список из объектов класса Human, а затем производит уменьшение веса каждого объекта на 5, фильтрация по дате рождения меньшей, чем 3 февраля 1999, конкатенация фамилий в строку через пробел.

1. **Код и пояснение к коду**

Код: <https://github.com/gavril-s/mirea-4/tree/master/java/pract/src/task2/opt6>

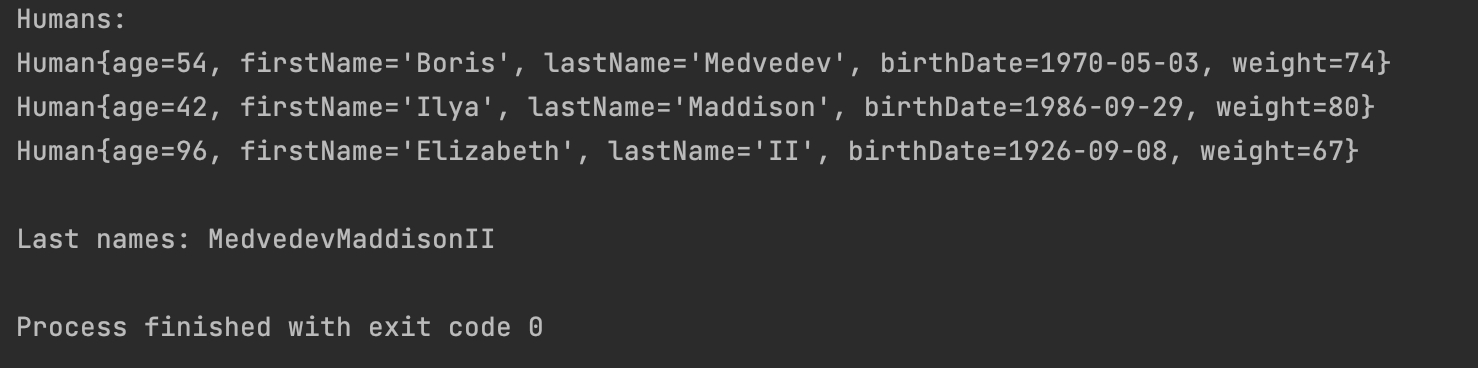
Пояснения:

Класс Human – реализация класса Human согласно условию.

Класс HumanManager – класс, выполняющий операции, указанные в условии, над списком экземпляров Human, используя Stream API.

Класс Main – тестирование класса HumanManager.

1. **Результат работы программы**

****

**Практическая работа №3**

1. **Постановка задачи и персональный вариант**

Тема: Знакомство с конкурентным программированием в Java. Потокобезопасность, ключевое слово syncrhonized, мьютексы, семафоры, мониторы, барьеры.

Постановка задачи: Set с использованием Lock, Map с использованием Semaphore.

1. **Код и пояснение к коду**

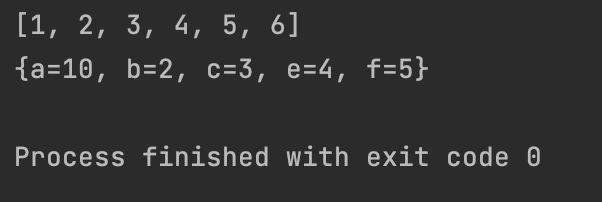
Код: <https://github.com/gavril-s/mirea-4/tree/master/java/pract/src/task3/opt6>

Пояснения:

Класс LockSet реализует множество с использованием Lock. Представляет из себя обёртку над классом HashSet стандартной библиотеки, которая добавляет взаимодействие с Lock в каждую операцию.

Класс SemaphoreMap реализует словарь с использованием Semaphore. Аналогично классу LockSet представляет из себя обёртку над классом HashMap стандартной библиотеки.

1. **Результат работы программы**



**Практическая работа №4**

1. **Постановка задачи и персональный вариант**

Тема: Работа с ExecutorService, CompletableFuture.

Постановка задачи: Реализовать собственную имплементацию ExecutorService с единственным параметром конструктора – количеством потоков.

1. **Код и пояснение к коду**

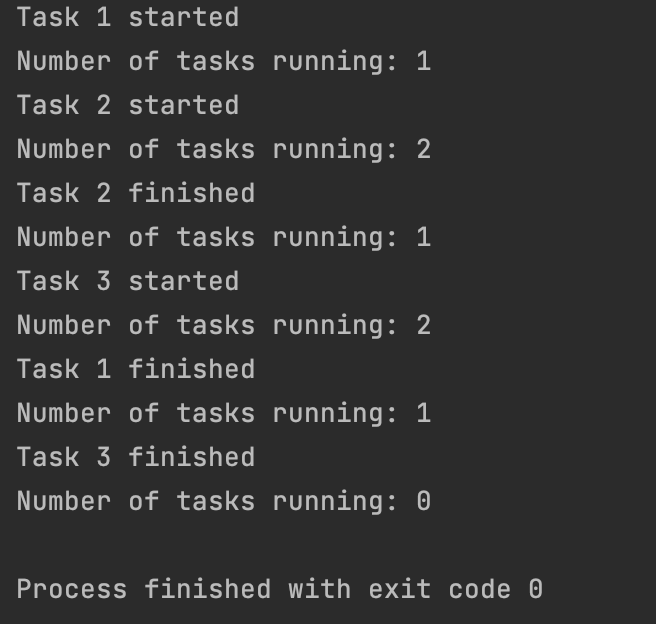
Код: [https://github.com/gavril-s/mirea-4/tree/master/java/pract/src/task4](https://github.com/gavril-s/mirea-4/tree/master/java/pract/src/task4/opt6)

Пояснения:

Класс ExecutorService реализован с помощью Semaphore, что позволяет поддерживать фиксированное число потоков.

Класс Main проводит тестирование класса Executor Service.

1. **Результат работы программы**

****