Университет ИТМО

Мегафакультет компьютерных технологий и управления

Факультет программной инженерии и компьютерной техники

Лабораторная работа по программированию №2

Вариант №2666

Группа: Р3107

Студент: Чусовлянов Максим Сергеевич

Преподаватель: Письмак Алексей Евгеньевич

г. Санкт-Петербург

2023

Оглавление

[Текст задания 3](#_fj05fxrbwn1h)

[Диаграмма классов 4](#_30j0zll)

[Исходный код программы 5](#_1fob9te)

[Результат работы программы 6](#)

[Вывод 7](#_3znysh7)

# 

# **Текст задания**

На основе базового класса Pokemon написать свои классы для заданных видов покемонов. Каждый вид покемона должен иметь один или два типа и стандартные базовые характеристики:

* очки здоровья (HP)
* атака (attack)
* защита (defense)
* специальная атака (special attack)
* специальная защита (special defense)
* скорость (speed)

Классы покемонов должны наследоваться в соответствии с цепочкой эволюции покемонов. На основе базовых классов PhysicalMove, SpecialMove и StatusMove реализовать свои классы для заданных видов атак.

Атака должна иметь стандартные тип, силу (power) и точность (accuracy). Должны быть реализованы стандартные эффекты атаки. Назначить каждому виду покемонов атаки в соответствии с вариантом. Уровень покемона выбирается минимально необходимым для всех реализованных атак.

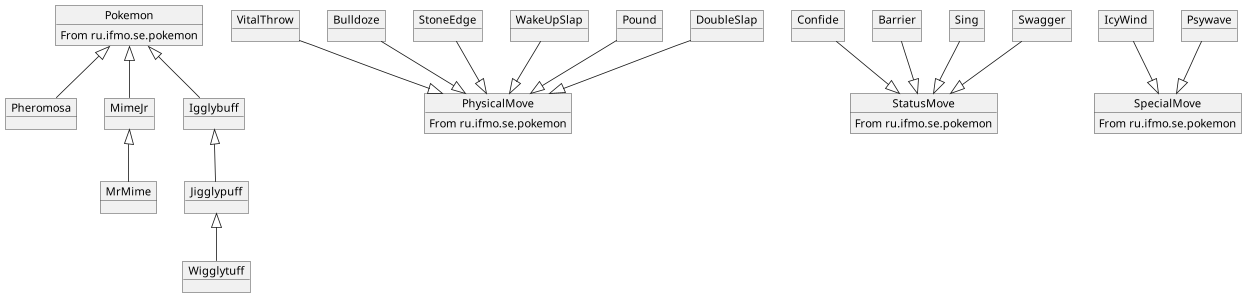
Используя класс симуляции боя Battle, создать 2 команды покемонов (каждый покемон должен иметь имя) и запустить бой.

Базовые классы и симулятор сражения находятся в [jar-архиве](https://se.ifmo.ru/documents/10180/660917/Pokemon.jar/a7ce60af-6ee6-47d0-a95e-e5ed9a697bd2). Информацию о покемонах, цепочках эволюции и атаках можно найти на сайтах.



# **Диаграмма классов**

Вывод в UML-формате см. в репозитори: <https://github.com/Vaneshik/VT-Labs/blob/main/java/lab2/UML/uml.plantuml>



# **Исходный код программы**

Репозиторий: <https://github.com/Vaneshik/VT-Labs/tree/main/java/lab2>

# **Результат работы программы**

См. в репозитории: <https://raw.githubusercontent.com/Vaneshik/VT-Labs/main/java/lab2/battle.log>

# 

# **Вывод**

Во время выполнения данной лабораторной работы, я изучил синтаксис PlantUML для создания UML-диаграмм, узнал об основах ООП и научился применять их на практике, научился подключать внешнюю jar-зависимость в проект и собирать такой проект при помощи средств JDK.