САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ

ИССЛЕДРОВАТЕЛЬСКИЙ

УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,

МЕХАНИКИ И ОПТИКИ

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Программирование»

**Отчет**

По лабораторной работе №2

Вариант 290008

Студент

Федоров Евгений Константинович

Преподаватель

Наумова Надежда Александровна

Санкт-Петербург, 2023 г.

**Текст задания:**

Цель работы: на простом примере разобраться с основными концепциями ООП и научиться использовать их в программах.

Что надо сделать (краткое описание)

1. Ознакомиться с [документацией](https://se.ifmo.ru/~tony/doc/), обращая особое внимание на классы Pokemon и Move. При дальнейшем выполнении лабораторной работы читать документацию еще несколько раз.
2. Скачать файл Pokemon.jar. Его необходимо будет использовать как для компиляции, так и для запуска программы. Распаковывать его не надо! Нужно научиться подключать внешние jar-файлы к своей программе.
3. Написать минимально работающую программу и посмотреть как она работает.

Battle b = new Battle();

Pokemon p1 = new Pokemon("Чужой", 1);

Pokemon p2 = new Pokemon("Хищник", 1);

b.addAlly(p1);

b.addFoe(p2);

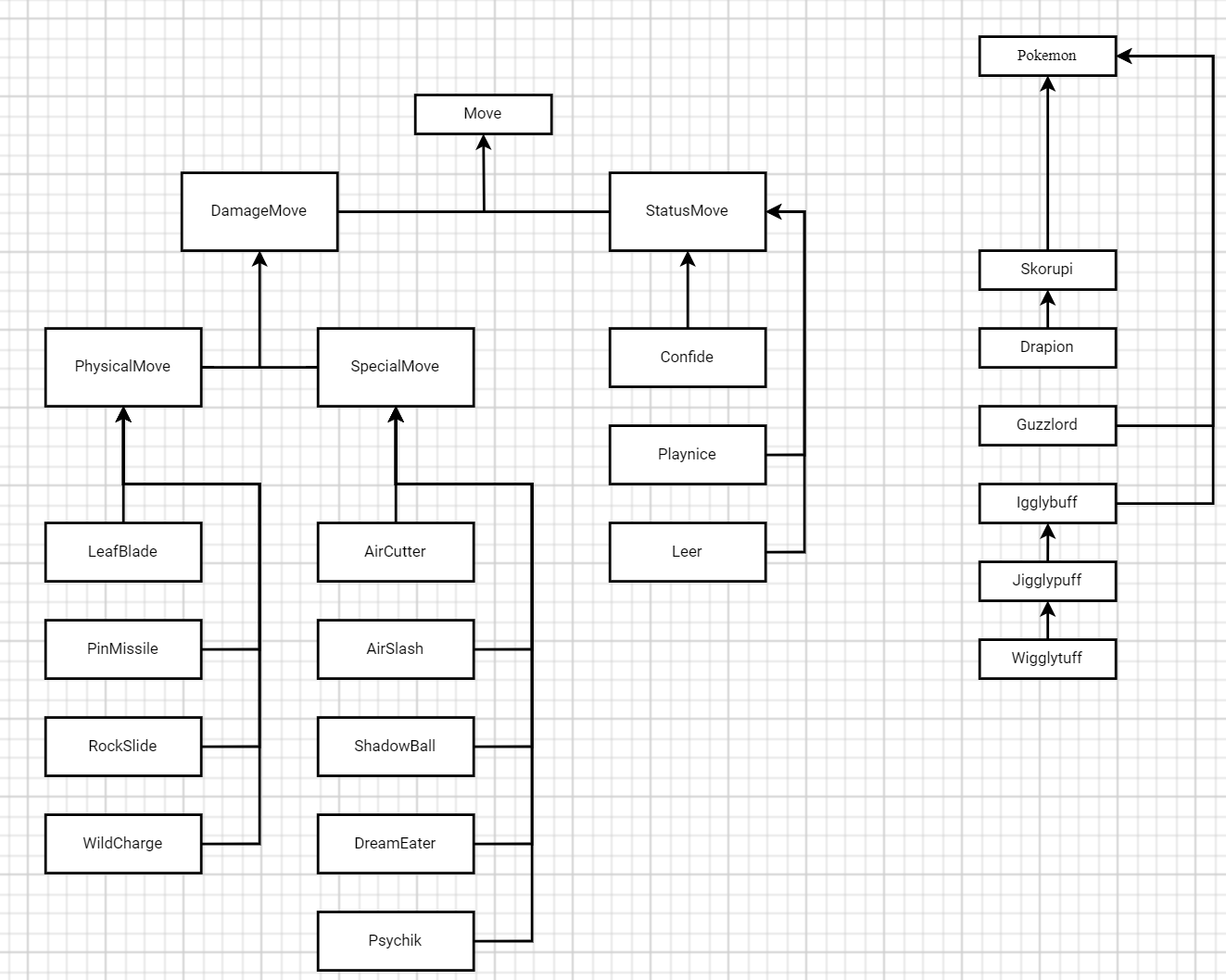
b.go();

1. Создать один из классов покемонов для своего варианта. Класс должен наследоваться от базового класса Pokemon. В конструкторе нужно будет задать типы покемона и его базовые характеристики. После этого попробуйте добавить покемона в сражение.
2. Создать один из классов атак для своего варианта (лучше всего начать с физической или специальной атаки). Класс должен наследоваться от класса PhysicalMove или SpecialMove. В конструкторе нужно будет задать тип атаки, ее силу и точность. После этого добавить атаку покемону и проверить ее действие в сражении. Не забудьте переопределить метод describe, чтобы выводилось нужное сообщение.
3. Если действие атаки отличается от стандартного, например, покемон не промахивается, либо атакующий покемон также получает повреждение, то в классе атаки нужно дополнительно переопределить соответствующие методы (см. документацию). При реализации атак, которые меняют статус покемона (наследники StatusMove), скорее всего придется разобраться с классом Effect. Он позволяет на один или несколько ходов изменить состояние покемона или модификатор его базовых характеристик.
4. Доделать все необходимые атаки и всех покемонов, распределить покемонов по командам, запустить сражение.

Покемоны:



**UML-диаграмма:**

****

**Код программы:**

***Покемоны:***

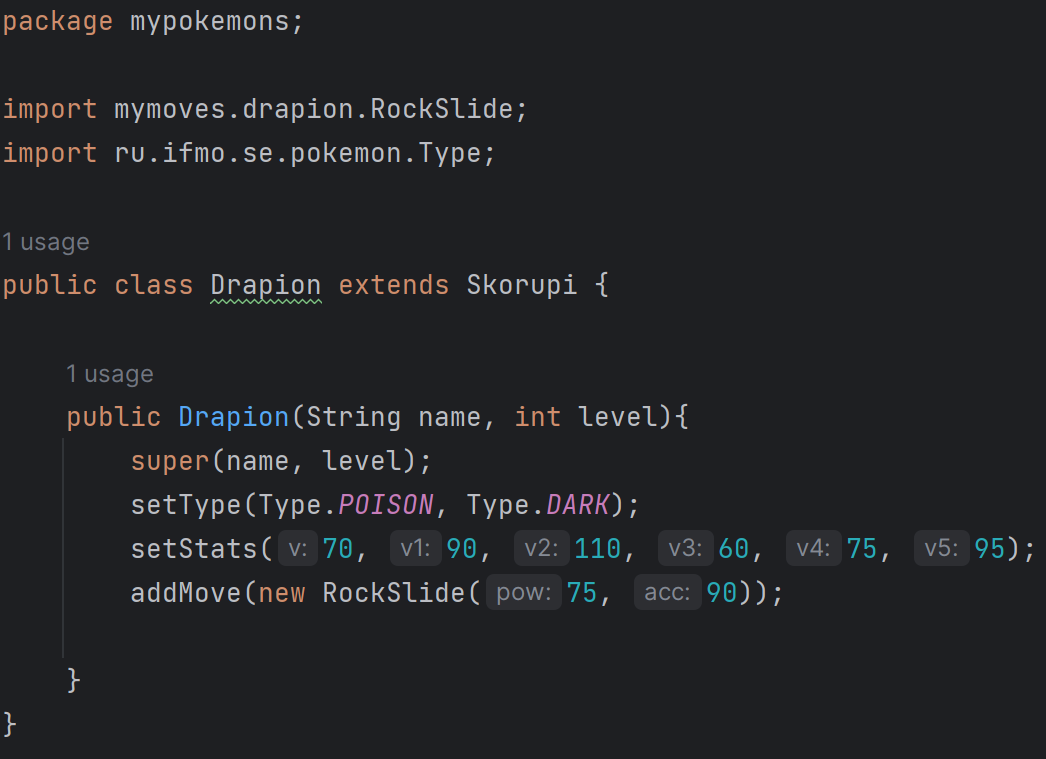
Guzzlord.java

**

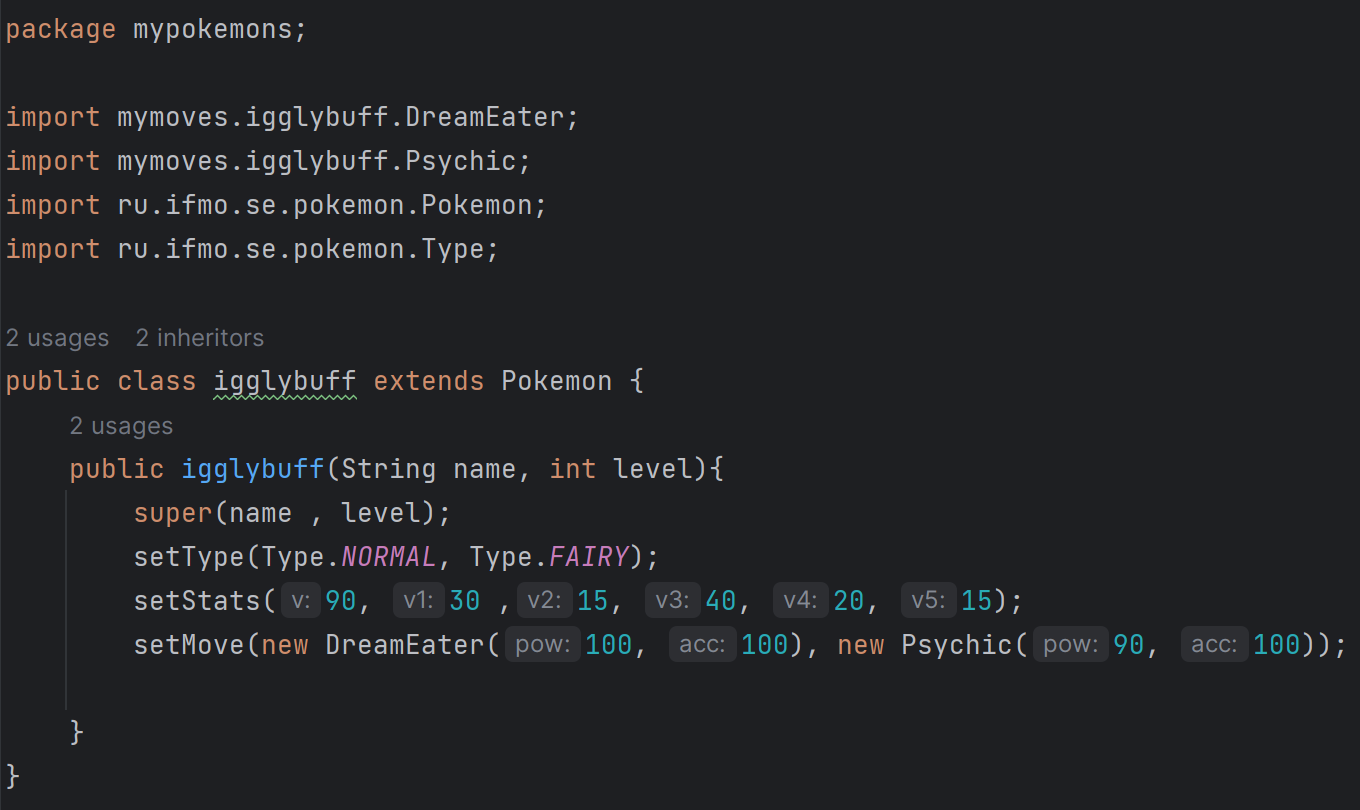
Skorupi.java



Drapion.java

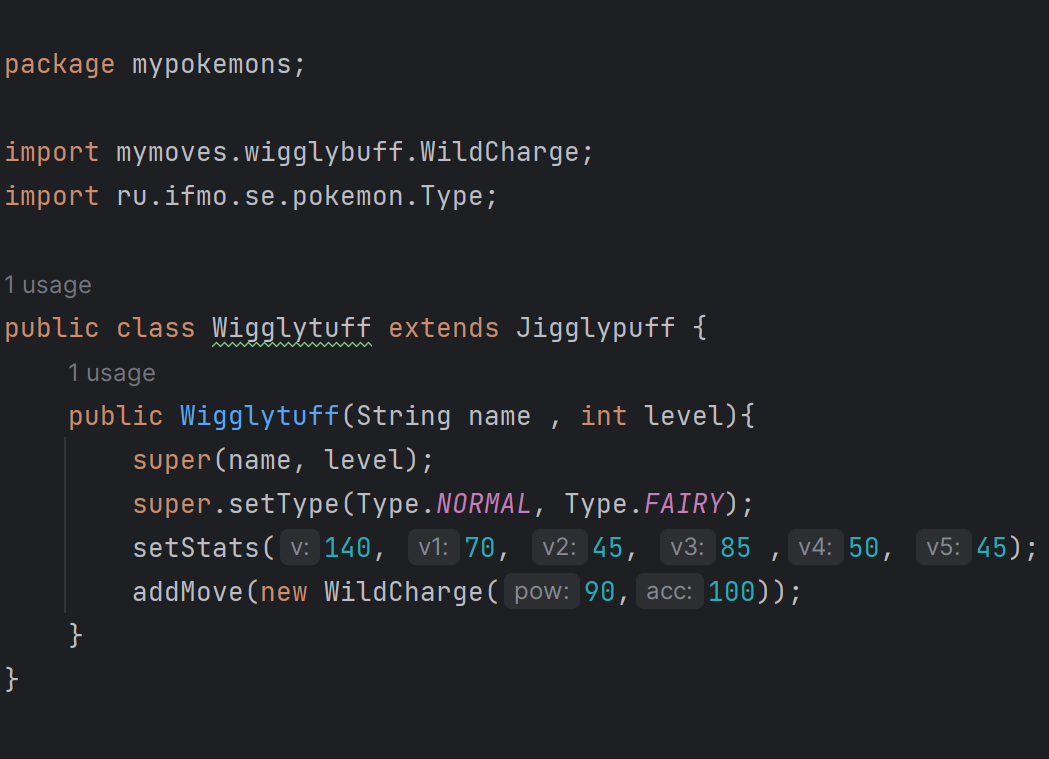


Igglybuff.java



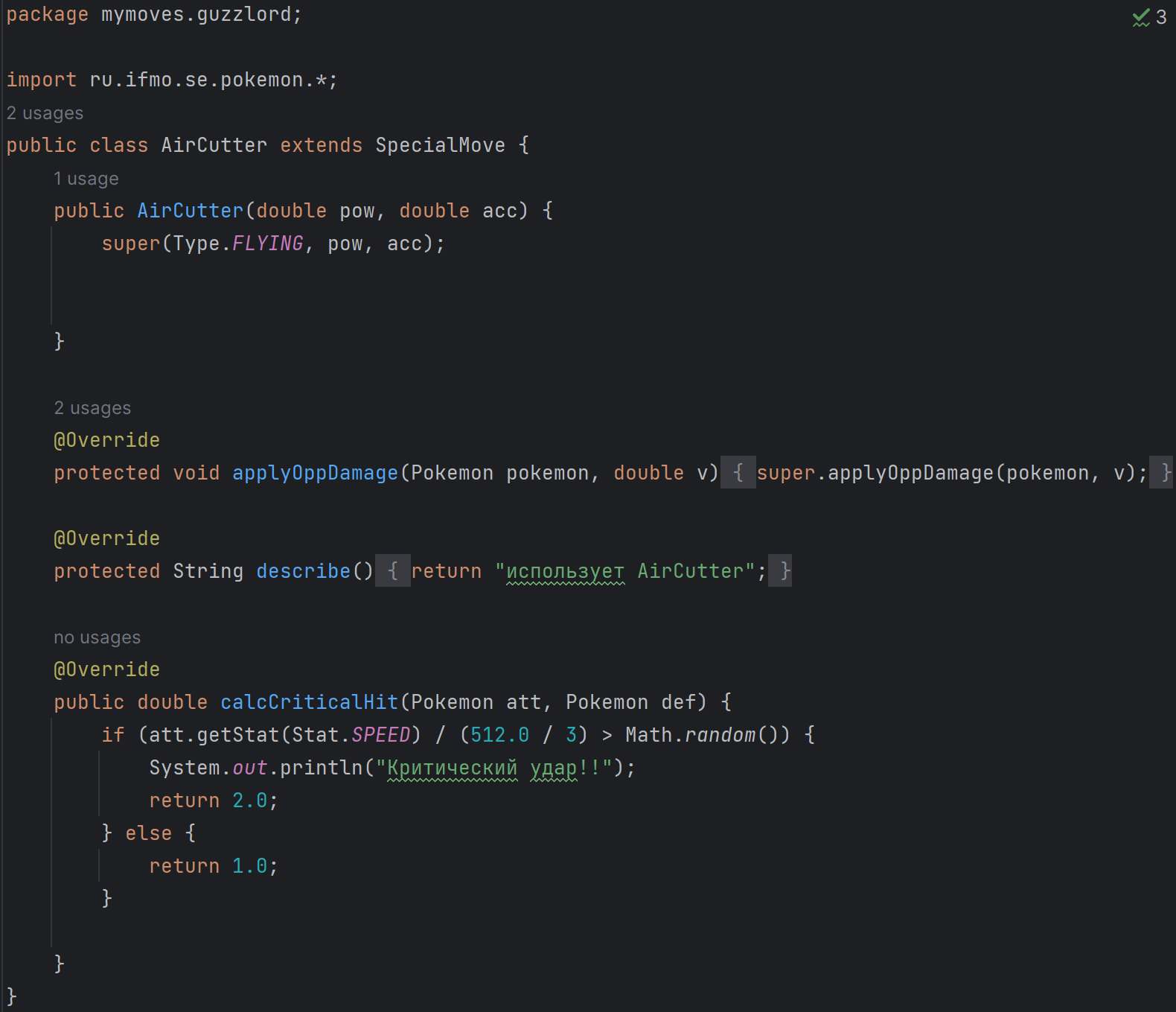
Jigglypuff.java

  
Wigglytuff.java

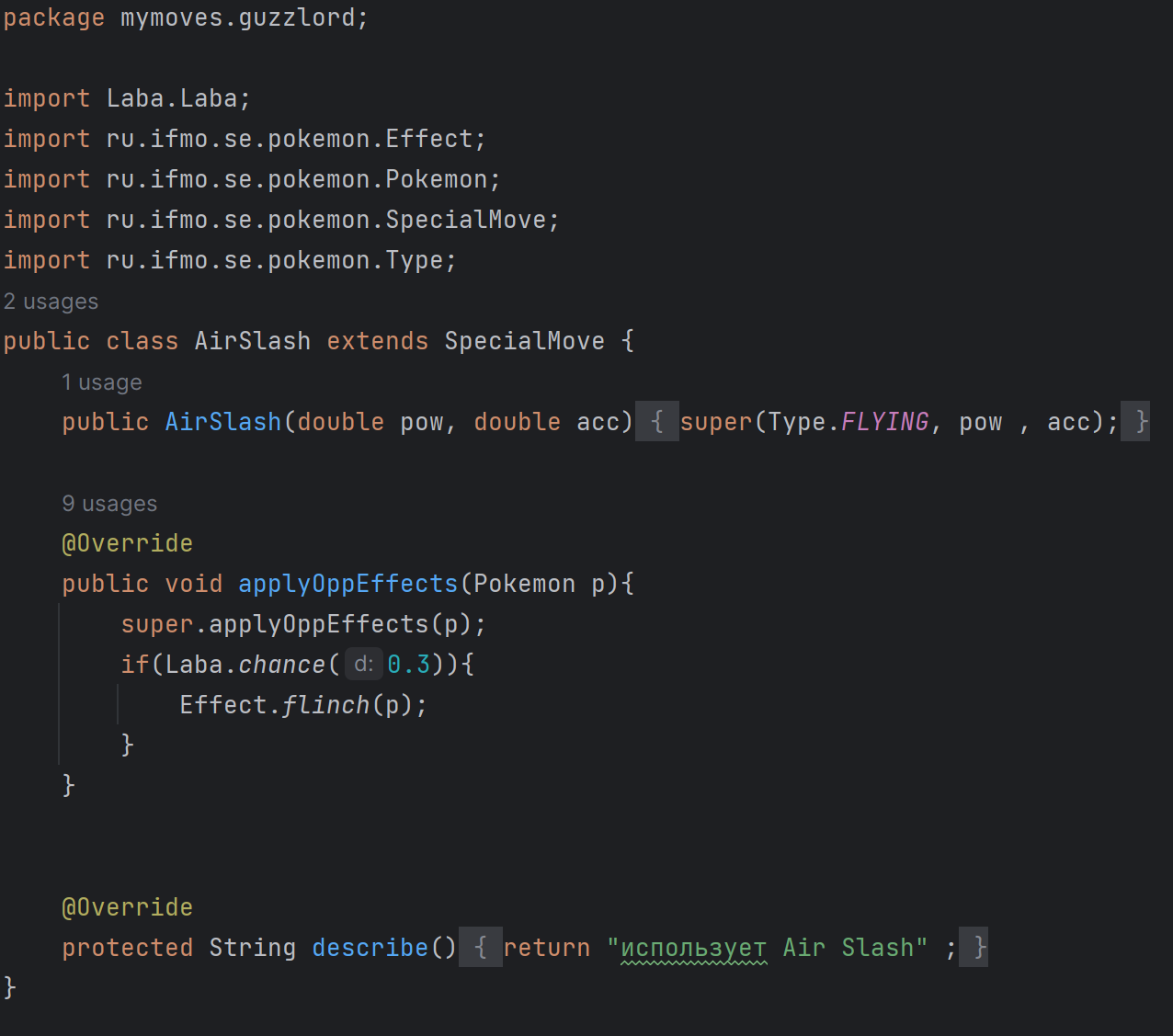


***Атаки:***

AirCutter.java



AirSlash.java



Confide.java



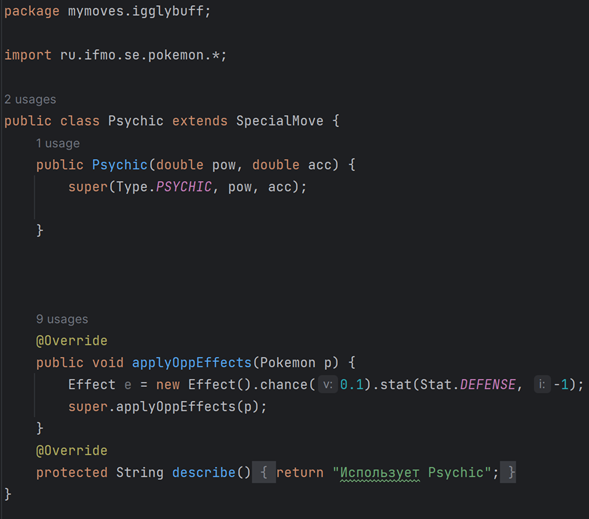
Leafblade.java



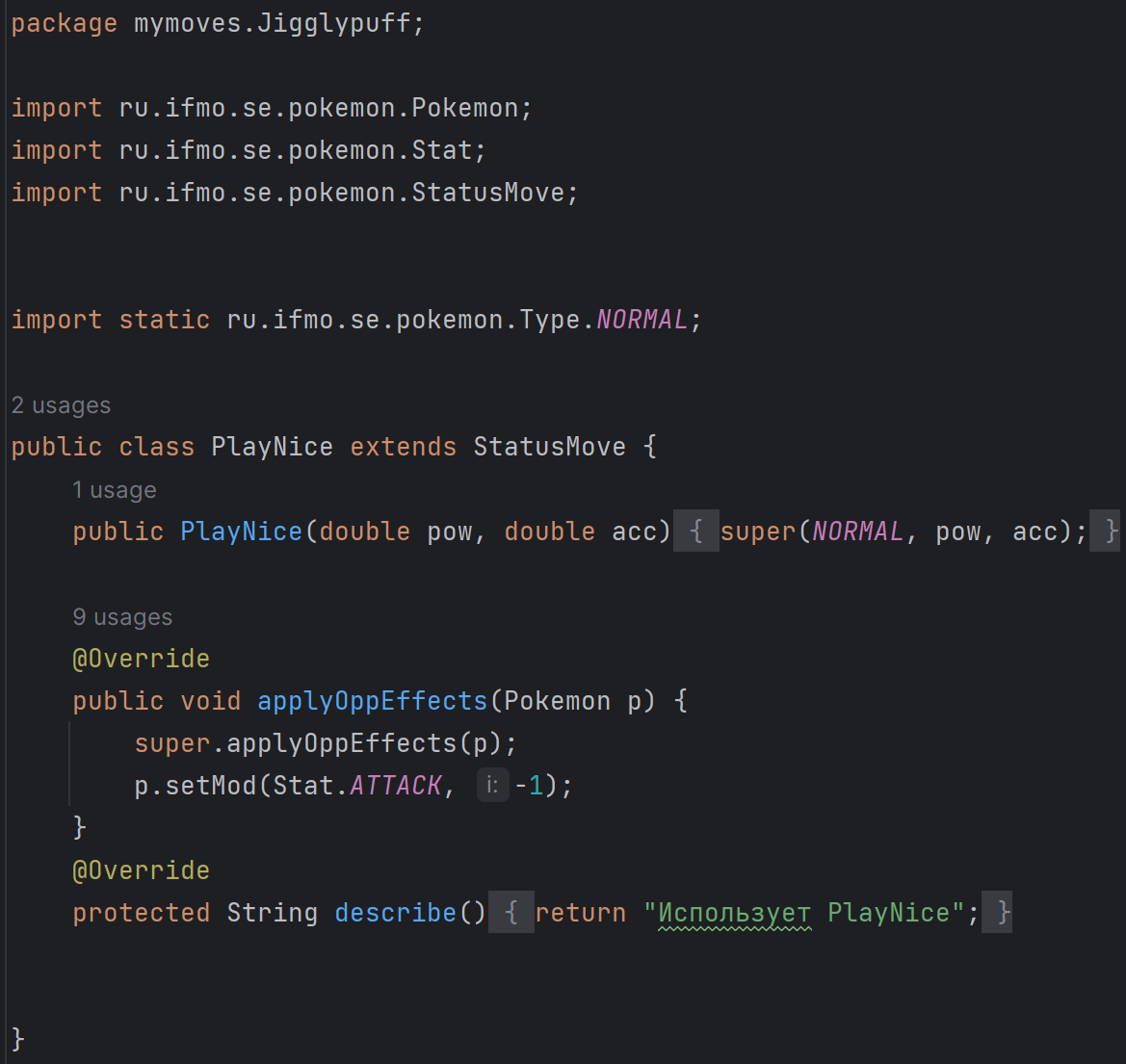
DreamEater.java



Psychic.java



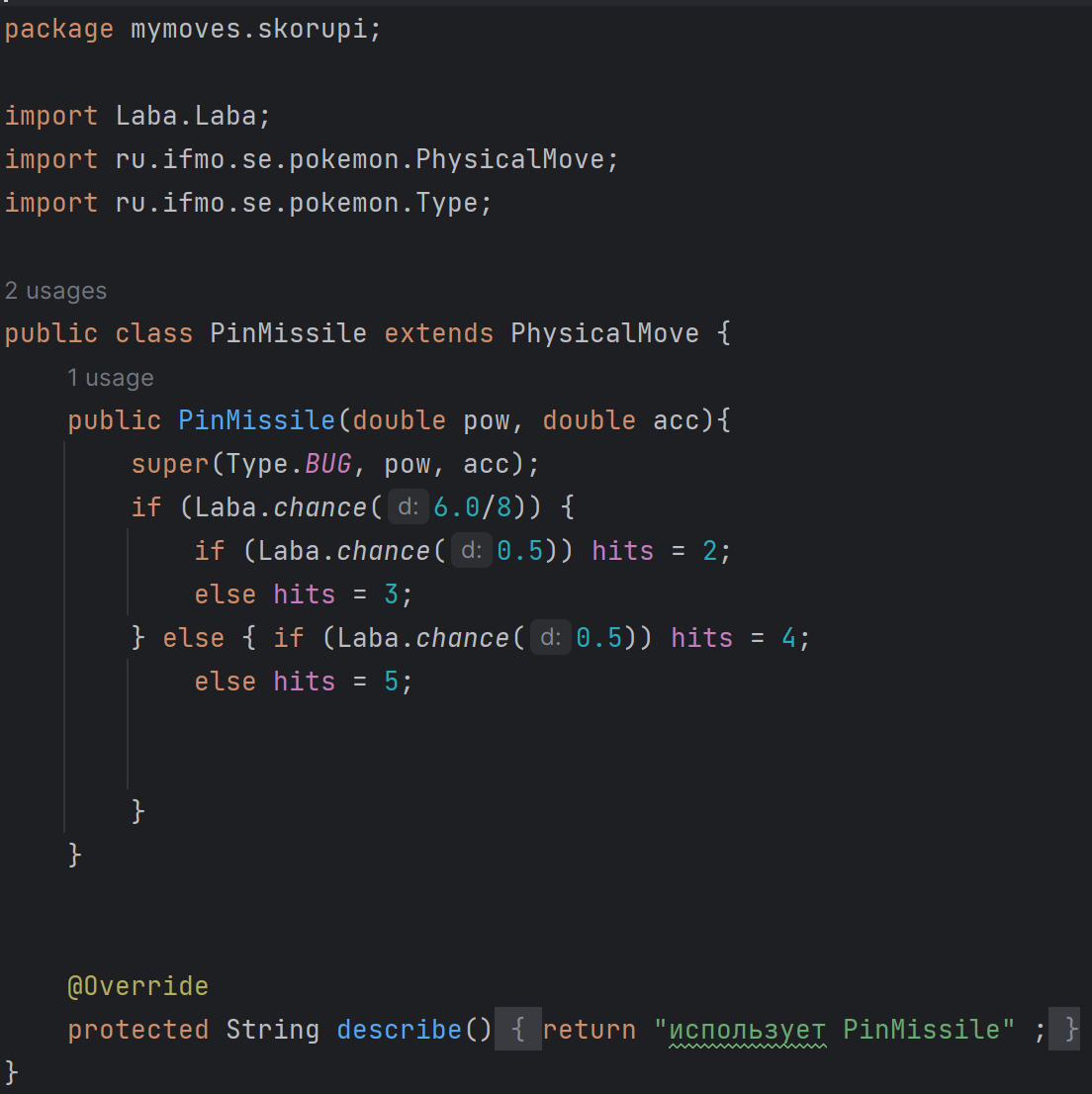
PlayNice.java

******

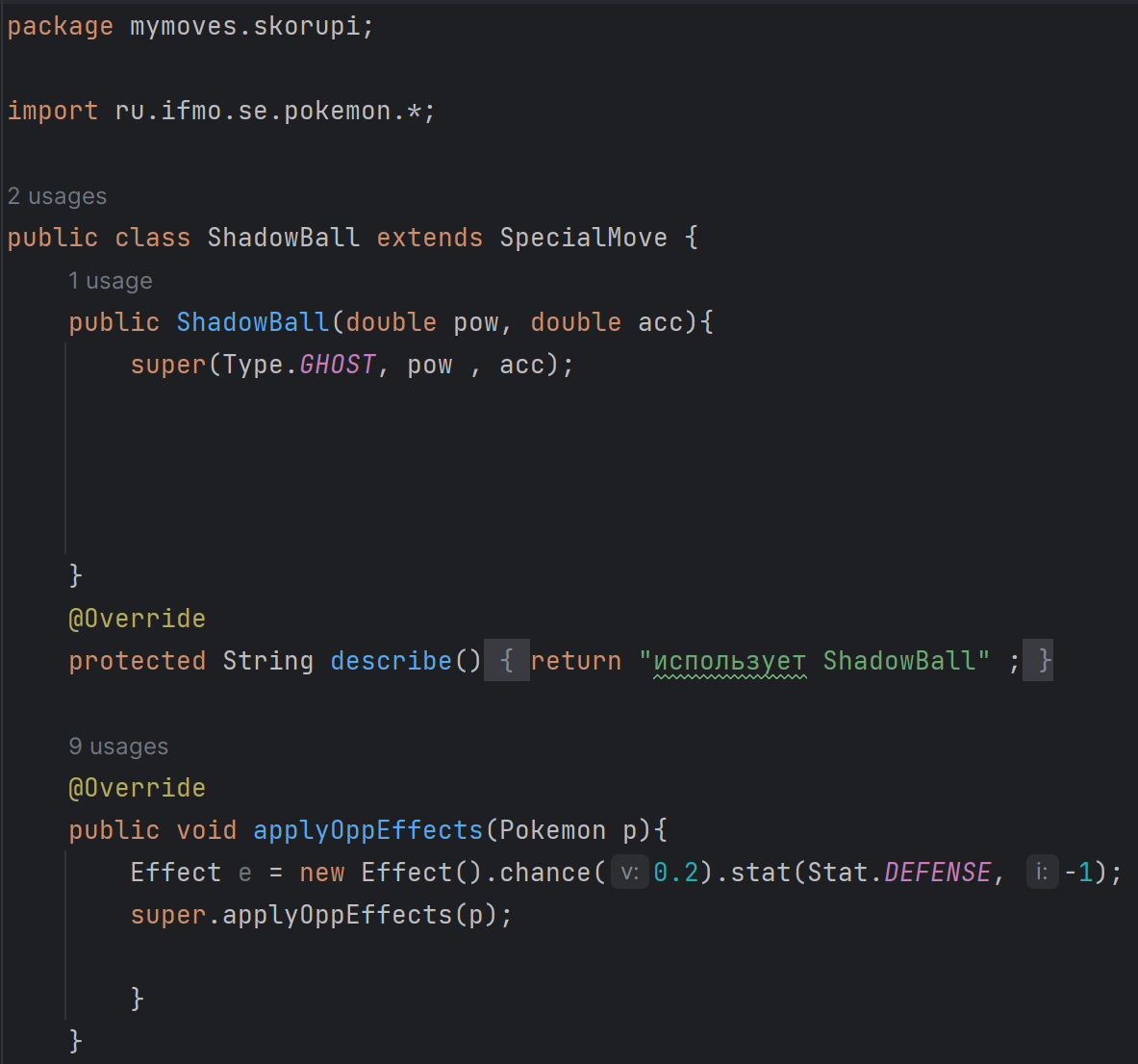
Leer.java



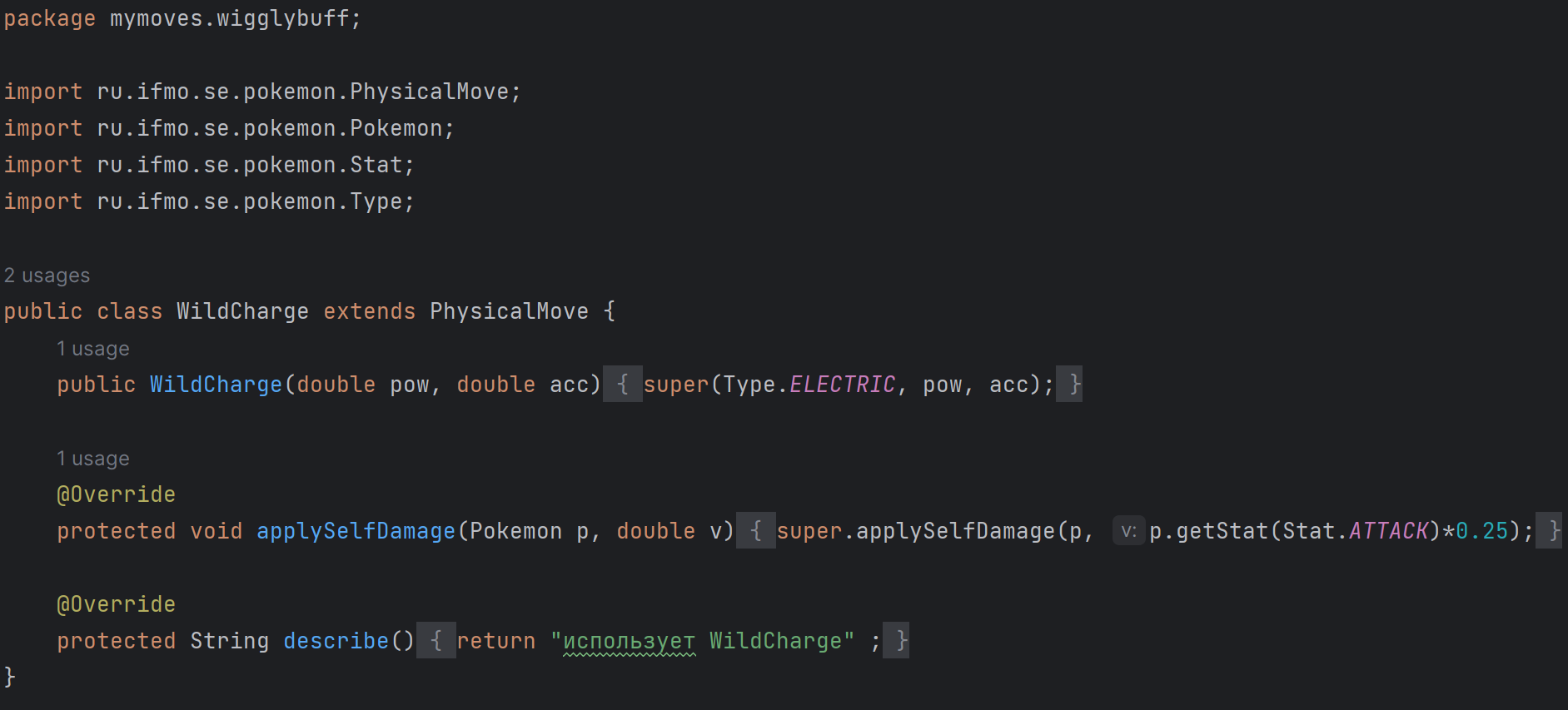
PinMissile.java



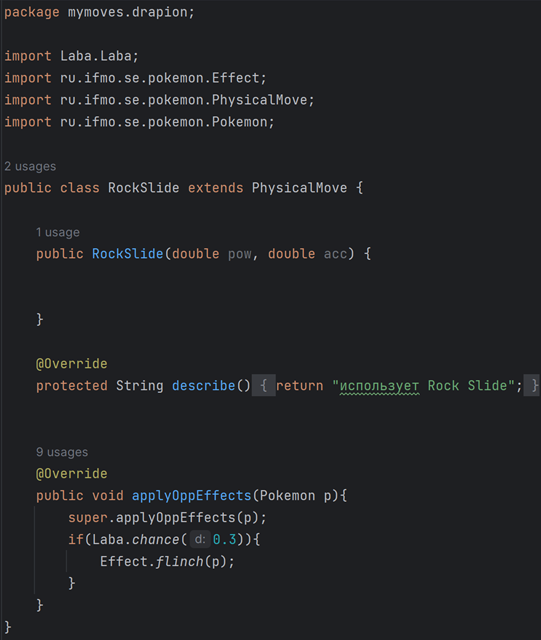
ShadowBall.java



WildCharge.java



RockSlide.java



***Главный файл:***

Laba.java



**Результат работы:**

guzzlord Guzzy из команды синих вступает в бой!

Drapion Drappy из команды белых вступает в бой!

Drapion Drappy использует ShadowBall.

guzzlord Guzzy теряет 4 здоровья.

guzzlord Guzzy промахивается

Drapion Drappy использует Rock Slide.

guzzlord Guzzy теряет 3 здоровья.

guzzlord Guzzy использует Air Slash.

Drapion Drappy теряет 6 здоровья.

Drapion Drappy использует ShadowBall.

guzzlord Guzzy теряет 4 здоровья.

guzzlord Guzzy использует AirCutter.

Drapion Drappy теряет 5 здоровья.

Drapion Drappy использует Leer.

guzzlord Guzzy уменьшает защиту.

guzzlord Guzzy промахивается

Drapion Drappy использует PinMissile.

guzzlord Guzzy теряет 6 здоровья.

Drapion Drappy использует PinMissile.

guzzlord Guzzy теряет 7 здоровья.

Drapion Drappy использует PinMissile.

guzzlord Guzzy теряет 10 здоровья.

Drapion Drappy использует PinMissile.

guzzlord Guzzy теряет 5 здоровья.

guzzlord Guzzy теряет сознание.

Skorupi Skorpi из команды синих вступает в бой!

Drapion Drappy использует Rock Slide.

Skorupi Skorpi теряет 2 здоровья.

Skorupi Skorpi использует PinMissile.

Критический удар!

Drapion Drappy теряет 12 здоровья.

Skorupi Skorpi использует PinMissile.

Drapion Drappy теряет 5 здоровья.

Drapion Drappy теряет сознание.

Jigglypuff Jiggle из команды белых вступает в бой!

Skorupi Skorpi использует ShadowBall.

Jigglypuff Jiggle теряет 1 здоровья.

Jigglypuff Jiggle не замечает воздействие типа GHOST

Jigglypuff Jiggle промахивается

Skorupi Skorpi использует PinMissile.

Jigglypuff Jiggle теряет 3 здоровья.

Skorupi Skorpi использует PinMissile.

Jigglypuff Jiggle теряет 2 здоровья.

Jigglypuff Jiggle Использует DreamEater.

Skorupi Skorpi теряет 11 здоровья.

Покемончик не спит

Skorupi Skorpi использует ShadowBall.

Jigglypuff Jiggle теряет 1 здоровья.

Jigglypuff Jiggle не замечает воздействие типа GHOST

Jigglypuff Jiggle промахивается

Skorupi Skorpi использует PinMissile.

Jigglypuff Jiggle теряет 3 здоровья.

Skorupi Skorpi использует PinMissile.

Jigglypuff Jiggle теряет 4 здоровья.

Jigglypuff Jiggle теряет сознание.

igglybuff Iggle из команды белых вступает в бой!

Skorupi Skorpi использует ShadowBall.

igglybuff Iggle теряет 1 здоровья.

igglybuff Iggle не замечает воздействие типа GHOST

igglybuff Iggle Использует DreamEater.

Skorupi Skorpi теряет 11 здоровья.

Покемончик не спит

Skorupi Skorpi теряет сознание.

Wigglytuff Wistle из команды синих вступает в бой!

Wigglytuff Wistle Использует Psychic.

igglybuff Iggle теряет 5 здоровья.

igglybuff Iggle Использует Psychic.

Wigglytuff Wistle теряет 6 здоровья.

Wigglytuff Wistle Использует DreamEater.

igglybuff Iggle теряет 6 здоровья.

Покемончик не спит

igglybuff Iggle Использует DreamEater.

Wigglytuff Wistle теряет 5 здоровья.

Покемончик не спит

Wigglytuff Wistle промахивается

igglybuff Iggle Использует Psychic.

Wigglytuff Wistle теряет 5 здоровья.

Wigglytuff Wistle теряет сознание.

В команде синих не осталось покемонов.

Команда белых побеждает в этом бою!

Вывод:

Во время выполнения лабораторной работы я изучил основные концепции ООП, такие как Наследование, Инкапсуляция, Полиморфизм, и применил их на практике. Научился работать с документацией в формате javadoc, создал диаграмму UML для структуры проекта, разобрался с модификаторами доступа.