

Министерство образования и науки РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Санкт-Петербургский национальный исследовательский университет

ИТМО»

Лабораторная работа №2

по дисциплине “Программирование”

Студент:

Новиков Даниил Дмитриевич, Р3131

Преподаватель:

Бобрусь Александр Владимирович

Санкт-Петербург

2024

Задание




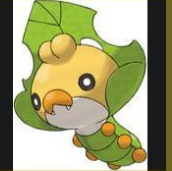


Цель работы: на простом примере разобраться с основными концепциями ООП и научиться использовать их в программах.

Что надо сделать (краткое описание)

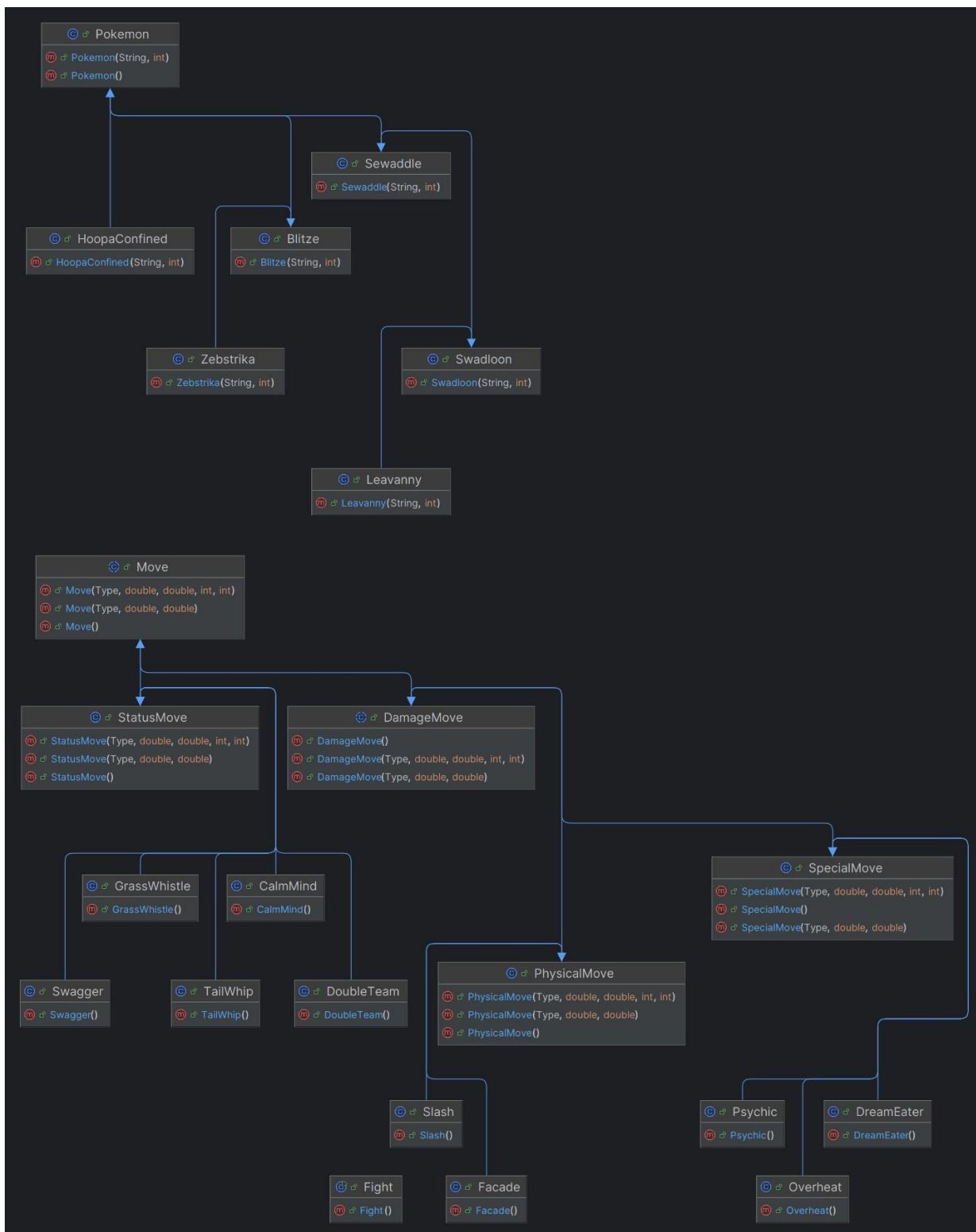
1. Ознакомиться с [документацией](#), обращая особое внимание на классы `Pokemon` и `Move`. При дальнейшем выполнении лабораторной работы читать документацию еще несколько раз.
2. Скачать файл `Pokemon.jar`. Его необходимо будет использовать как для компиляции, так и для запуска программы. Распаковывать его не надо! Нужно научиться подключать внешние `jar`-файлы к своей программе.
3. Написать минимально работающую программу и посмотреть как она работает.
4. `Battle b = new Battle();`
5. `Pokemon p1 = new Pokemon("Чужой", 1);`
6. `Pokemon p2 = new Pokemon("Хищник", 1);`
7. `b.addAlly(p1);`
8. `b.addFoe(p2);`
9. `b.go();`
10. 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.
11. 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.
12. 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.

13. Доделать все необходимые атаки и всех покемонов, распределить покемонов по командам, запустить сражение.

Hoopa-Confined  Атаки: <ul style="list-style-type: none">✂ Swagger✂ Dream Eater✂ Psychic✂ Facade	Blitzle  Атаки: <ul style="list-style-type: none">✂ Double Team✂ Swagger✂ Tail Whip	Zebstrika  Атаки: <ul style="list-style-type: none">✂ Double Team✂ Swagger✂ Tail Whip✂ Overheat	Sewaddle  Атаки: <ul style="list-style-type: none">✂ Swagger✂ Calm Mind	Swadloon  Атаки: <ul style="list-style-type: none">✂ Swagger✂ Calm Mind✂ Grass Whistle
Leavanny  Атаки: <ul style="list-style-type: none">✂ Swagger✂ Calm Mind✂ Grass Whistle✂ Slash				

UML диаграмма



Исходный код программы

<https://github.com/Buratishkin/ITMO/tree/main/prog/lab2/code>

Результат работы программы

<https://github.com/Buratishkin/ITMO/tree/main/prog/lab2/test.txt>

Выводы к работе

В этой работе я:

- читал документацию для библиотеки Pokemon.jar
- создал классы для всех своих покемонов
- реализовал для них различные атаки
- изучил подробно класс Effect
- изучил принципы ООП в Java
- изучил различные модификаторы доступа
- изучил пакеты и инструкцию import