

Национальный исследовательский университет ИТМО

Факультет Программной Инженерии и Компьютерной  
техники

**Программирование**  
**Лабораторная работа №2**  
**Введение в ООП**  
**Вариант №27922**

Выполнил:  
Кузьмин Дмитрий Анатольевич

Группа:  
Р3109

Преподаватель:  
Гаврилов А. В.,  
Мустафаева А. В.

# Содержание

1	Задание	2
2	Ход работы	3
3	Вывод	4

# 1 Задание

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. 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();
```

4. 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.
5. 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.

6. 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.
7. Доделать все необходимые атаки и всех покемонов, распределить покемонов по командам, запустить сражение.



Рис. 1: Мои покемоны

## 2 Ход работы

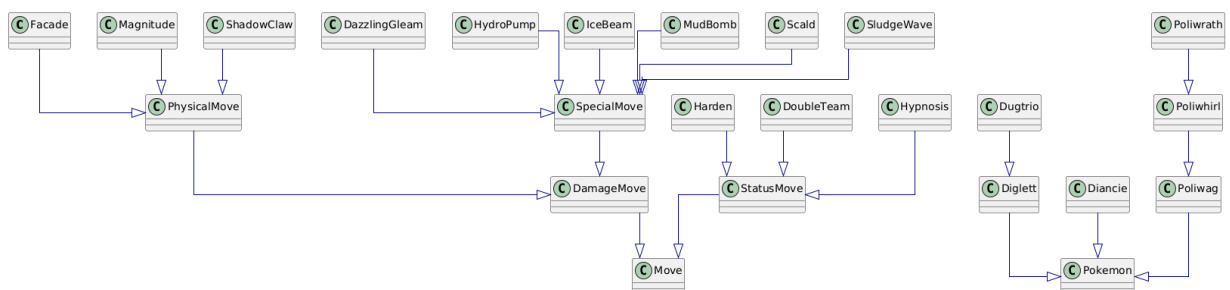


Рис. 2: Диаграмма классов реализованной объектной модели

[Ссылка на код](#)

Пример результата работы программы:

```
Diancie Котик из команды полосатых вступает в бой!  
Diglett Зайчик из команды синих вступает в бой!  
Diglett Зайчик использует Mud_Bomb.  
Diancie Котик теряет 12 здоровья.  
Diancie Котик уменьшает точность.  
  
Diancie Котик использует Facade.  
Diglett Зайчик теряет 4 здоровья.  
  
Diglett Зайчик использует Mud_Bomb.  
Diancie Котик теряет 13 здоровья.  
Diancie Котик теряет сознание.  
В команде полосатых не осталось покемонов.  
Команда синих побеждает в этом бою!
```

Рис. 3: Пример боя покемонов

### 3 Вывод

В ходе лабораторной работы я разобрался в ООП. Я научился создавать классы, работать с внешним jar файлом и понял как работает наследование и модификаторы доступа.