Министерство образования и науки РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики»

**факультет программной инженерии и компьютерной техники**

**ЛАБОРАТОРНАЯ РАБОТА № 2**

по дисциплине

‘Программирование’

Вариант №53272

*Выполнил:*

Студент группы P3109

Литвиненко Никита Алексеевич

*Преподаватель:*

Шешуков Дмитрий

Михайлович



Санкт-Петербург, 2021

Оглавление

[Цель 2](#_Toc91247635)

[Задание 3](#_Toc91247636)

[Диаграмма классов реализованной объектной модели 5](#_Toc91247637)

[Исходный код программы 6](#_Toc91247638)

[Main.java 6](#_Toc91247639)

[Pokemon 6](#_Toc91247640)

[Drapion.java 6](#_Toc91247641)

[Lombre.java 7](#_Toc91247642)

[Lotad.java 7](#_Toc91247643)

[Ludicolo.java 8](#_Toc91247644)

[Pheromosa.java 8](#_Toc91247645)

[Skorupi.java 8](#_Toc91247646)

[Attack 9](#_Toc91247647)

[Blizzard.java 9](#_Toc91247648)

[BrutalSwing.java 9](#_Toc91247649)

[Confide.java 9](#_Toc91247650)

[CrossPosion.java 10](#_Toc91247651)

[DarkPulse.java 10](#_Toc91247652)

[EnergyBall.java 11](#_Toc91247653)

[FocusEnergy.java 11](#_Toc91247654)

[HydroPump.java 11](#_Toc91247655)

[MegaPunch.java 12](#_Toc91247656)

[PinMissile.java 12](#_Toc91247657)

[Swagger.java 12](#_Toc91247658)

[Результат работы программы 13](#_Toc91247659)

[Выводы 14](#_Toc91247660)

# Цель

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

# Задание

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);

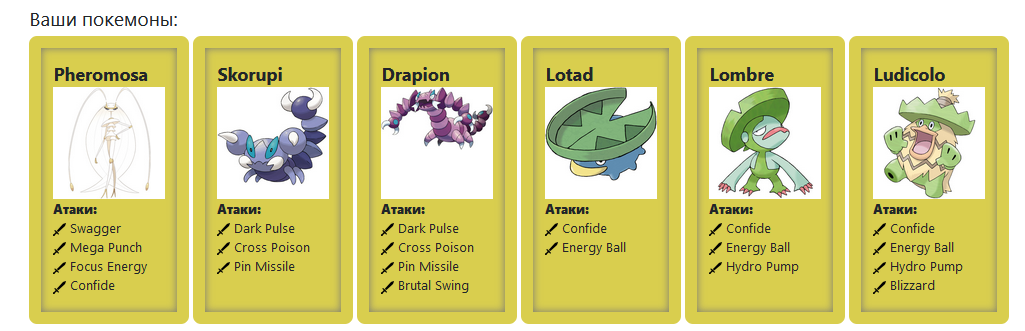
Pokemon p2 = new Pokemon("Хищник", 1);

b.addAlly(p1);

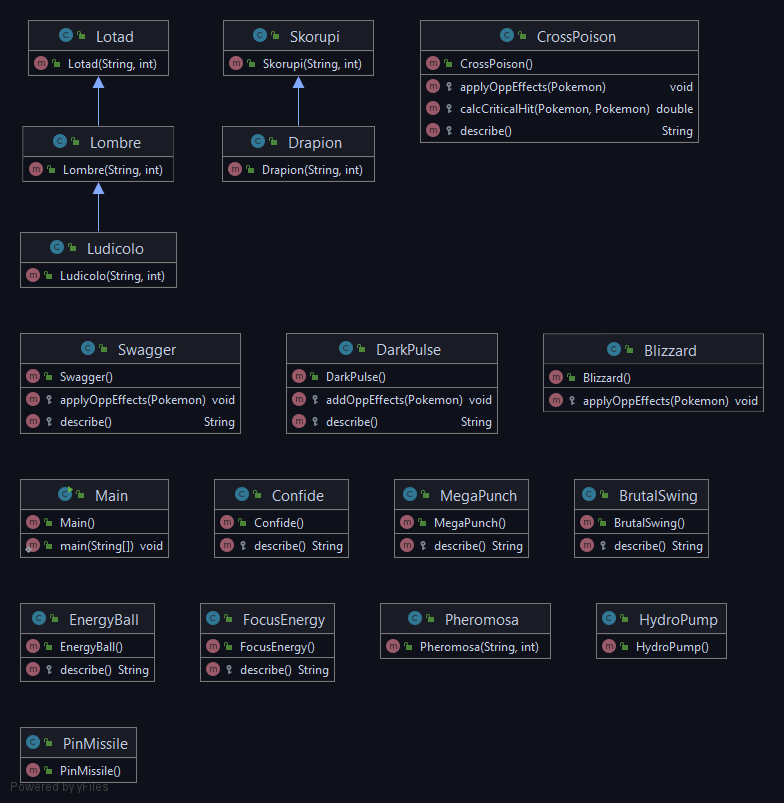
b.addFoe(p2);

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.



# Диаграмма классов реализованной объектной модели



# Исходный код программы

## Main.java

package com.company;

import com.company.pokemon.\*;

import ru.ifmo.se.pokemon.Battle;

public class Main {

public static void main(String[] args) {

Battle b = new Battle();

Pheromosa p1 = new Pheromosa("Shinji", 1);

Drapion p2 = new Drapion("Asuka", 1);

Skorupi p3 = new Skorupi("Rei", 2);

Lotad p4 = new Lotad("Misato", 1);

Lombre p5 = new Lombre("Kaworu", 2);

Ludicolo p6 = new Ludicolo("Gendo", 3);

b.addAlly(p1);

b.addAlly(p2);

b.addAlly(p3);

b.addFoe(p4);

b.addFoe(p5);

b.addFoe(p6);

b.go();

}

}

## Pokemon

### Drapion.java

package com.company.pokemon;

import com.company.attack.\*;

import ru.ifmo.se.pokemon.\*;

public class Drapion extends Skorupi{

public Drapion(String name, int level) {

super(name, level);

this.setStats(70, 90, 110, 60, 75,95);

this.setType(Type.POISON, Type.DARK);

this.setMove(new DarkPulse(), new CrossPoison(), new PinMissile(), new BrutalSwing());

}

}

### Lombre.java

package com.company.pokemon;

import com.company.attack.Confide;

import com.company.attack.HydroPump;

import ru.ifmo.se.pokemon.Pokemon;

public class Lombre extends Lotad{

public Lombre(String name, int level) {

super(name, level);

this.setStats(60,50,50,60,70,50);

this.setMove(new Confide(), new HydroPump());

}

}

### Lotad.java

package com.company.pokemon;

import com.company.attack.Confide;

import ru.ifmo.se.pokemon.\*;

public class Lotad extends Pokemon{

public Lotad(String name, int level) {

super(name, level);

this.setStats(40,30,30,40,50,30);

this.setType(Type.WATER, Type.GRASS);

this.setMove(new Confide());

}

}

### Ludicolo.java

package com.company.pokemon;

import com.company.attack.Blizzard;

import com.company.attack.Confide;

import com.company.attack.HydroPump;

import com.company.pokemon.\*;

public class Ludicolo extends Lombre {

public Ludicolo(String name, int level) {

super(name, level);

this.setStats(80,70,70,90,100,70);

this.setMove(new Confide(), new HydroPump(), new Blizzard());

}

}

### Pheromosa.java

package com.company.pokemon;

import com.company.attack.\*;

import ru.ifmo.se.pokemon.\*;

public class Pheromosa extends Pokemon{

public Pheromosa(String name, int level) {

super(name, level);

this.setStats(71, 137, 37, 137, 37, 151);

this.setType(Type.BUG, Type.FIGHTING);

this.setMove(new Swagger(), new MegaPunch(), new FocusEnergy(), new Confide());

}

}

### Skorupi.java

package com.company.pokemon;

import com.company.attack.\*;

import ru.ifmo.se.pokemon.\*;

public class Skorupi extends Pokemon{

public Skorupi(String name, int level) {

super(name, level);

this.setStats(40, 50, 90, 30, 55, 65);

this.setMove(new DarkPulse(), new CrossPoison(), new PinMissile());

}

}

## Attack

### Blizzard.java

package com.company.attack;

import ru.ifmo.se.pokemon.\*;

public class Blizzard extends SpecialMove {

public Blizzard() {

super(Type.ICE, 110, 70);

}

protected void applyOppEffects(Pokemon p){

Effect.freeze(p);

}

}

### BrutalSwing.java

package com.company.attack;

import ru.ifmo.se.pokemon.\*;

public class BrutalSwing extends PhysicalMove {

public BrutalSwing(){

super(Type.DARK, 60, 100);

}

protected String describe(){

return "использует Brutal Swing";

}

}

### Confide.java

package com.company.attack;

import ru.ifmo.se.pokemon.StatusMove;

import ru.ifmo.se.pokemon.Type;

public class Confide extends StatusMove {

public Confide(){

super(Type.NORMAL, 0,0);

}

protected String describe(){

return "использует Confide";

}

}

### CrossPosion.java

package com.company.attack;

import ru.ifmo.se.pokemon.\*;

public class CrossPoison extends PhysicalMove {

public CrossPoison(){

super(Type.POISON, 70, 100);

}

protected void applyOppEffects(Pokemon p){

if (Math.random() <= 0.1){

Effect.poison(p);

}

}

protected double calcCriticalHit(Pokemon att, Pokemon def) {

return 3.0;

}

protected String describe(){

return "использует Cross Poison";

}

}

### DarkPulse.java

package com.company.attack;

import ru.ifmo.se.pokemon.\*;

public class DarkPulse extends SpecialMove {

public DarkPulse() {

super(Type.DARK, 80, 100);

}

protected void addOppEffects(Pokemon p){

if (Math.random() <= 0.2){

Effect.flinch(p);

}

}

protected String describe() {

return "использует Dark Pulse";

}

}

### EnergyBall.java

package com.company.attack;

import ru.ifmo.se.pokemon.SpecialMove;

import ru.ifmo.se.pokemon.Type;

public class EnergyBall extends SpecialMove {

public EnergyBall(){

super(Type.GRASS, 90, 100);

}

protected String describe(){

return "использует Energy Ball";

}

}

### FocusEnergy.java

package com.company.attack;

import ru.ifmo.se.pokemon.StatusMove;

import ru.ifmo.se.pokemon.Type;

public class FocusEnergy extends StatusMove {

public FocusEnergy(){

super(Type.NORMAL, 0, 0);

}

protected String describe(){

return "использует Focus Energy";

}

}

### HydroPump.java

package com.company.attack;

import ru.ifmo.se.pokemon.\*;

public class HydroPump extends SpecialMove {

public HydroPump(){

super(Type.WATER, 110, 80);

}

}

### MegaPunch.java

package com.company.attack;

import ru.ifmo.se.pokemon.\*;

public class MegaPunch extends PhysicalMove {

public MegaPunch(){

super(Type.NORMAL, 80, 85);

}

protected String describe(){

return "использует Mega Punch";

}

}

### PinMissile.java

package com.company.attack;

import ru.ifmo.se.pokemon.\*;

public class PinMissile extends PhysicalMove {

public PinMissile() {

super(Type.BUG, 25, 95);

}

protected String describe(){

return "использует Pin Missile";

}

}

### Swagger.java

package com.company.attack;

import ru.ifmo.se.pokemon.\*;

public class Swagger extends StatusMove {

public Swagger() {

super(Type.FIGHTING, 0, 85);

}

protected void applyOppEffects(Pokemon p) {

p.setMod(Stat.ATTACK, 2);

Effect.confuse(p);

}

protected String describe(){

return "использует Swagger";

}

}

# Результат работы программы

Pheromosa Shinji from the team Purple enters the battle!

Lotad Misato from the team Black enters the battle!

Pheromosa Shinji использует Mega Punch.

Lotad Misato loses 5 hit points.

Lotad Misato misses

Pheromosa Shinji misses

Lotad Misato misses

Pheromosa Shinji использует Mega Punch.

Lotad Misato loses 7 hit points.

Lotad Misato faints.

Lombre Kaworu from the team Black enters the battle!

Pheromosa Shinji использует Swagger.

Lombre Kaworu increases attack.

Lombre Kaworu misses

Pheromosa Shinji misses

Lombre Kaworu attacks.

Pheromosa Shinji loses 14 hit points.

Pheromosa Shinji faints.

Drapion Asuka from the team Purple enters the battle!

Lombre Kaworu misses

Drapion Asuka attacks.

Lombre Kaworu loses 9 hit points.

Lombre Kaworu attacks.

Drapion Asuka loses 12 hit points.

Drapion Asuka использует Cross Poison.

Lombre Kaworu loses 46 hit points.

Lombre Kaworu faints.

Ludicolo Gendo from the team Black enters the battle!

Ludicolo Gendo misses

Drapion Asuka attacks.

Ludicolo Gendo loses 8 hit points.

Ludicolo Gendo attacks.

Drapion Asuka loses 9 hit points.

Drapion Asuka is frozen

Drapion Asuka faints.

Skorupi Rei from the team Purple enters the battle!

Ludicolo Gendo misses

Skorupi Rei attacks.

Ludicolo Gendo loses 7 hit points.

Ludicolo Gendo attacks.

Skorupi Rei loses 9 hit points.

Skorupi Rei is frozen

Skorupi Rei использует Cross Poison.

Ludicolo Gendo loses 35 hit points.

Ludicolo Gendo faints.

Team Black loses its last Pokemon.

The team Purple wins the battle!

Process finished with exit code 0

# Выводы

Объектно-ориентированное программирование – это очень удобно, однако, пример “Каждая эволюция одного покемона – это еще один наследник этого класса”, как мне кажется, не совсем удачный.