Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 «Программная инженерия» –

Системное и прикладное программное обеспечение

**Отчёт**

**По лабораторной работе №2**

**По программированию**

**Вариант: 94663**

Выполнил:

студент 1 курса

Батманов Даниил Евгеньевич

Группа: Р3107

Принял:

Гаврилов Антон Валерьевич

Отчёт принят «\_\_»\_\_\_\_\_2022 г.

Оценка: \_\_\_\_\_\_\_\_\_\_\_

г. Санкт-Петербург, 2022

**Оглавление**

[Задание 3](#_Toc118222211)

[Диаграмма классов реализованной объектной модели 4](#_Toc118222212)

[Исходный код программы 5](#_Toc118222213)

[Результат работы программы 17](#_Toc118222214)

[Вывод 34](#_Toc118222215)

[Список литературы 35](#_Toc118222216)

# 

# 

# 

# Задание

На основе базового класса Pokemon написать свои классы для заданных видов покемонов. Каждый вид покемона (рисунок 1) 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.

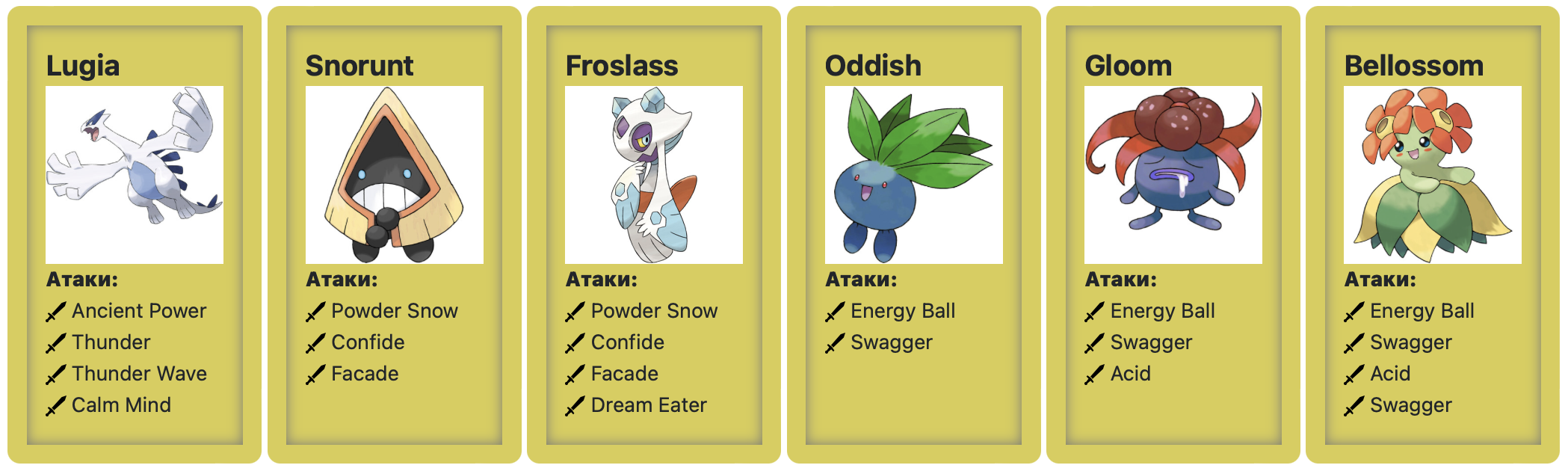


Рисунок 1

# Диаграмма классов реализованной объектной модели

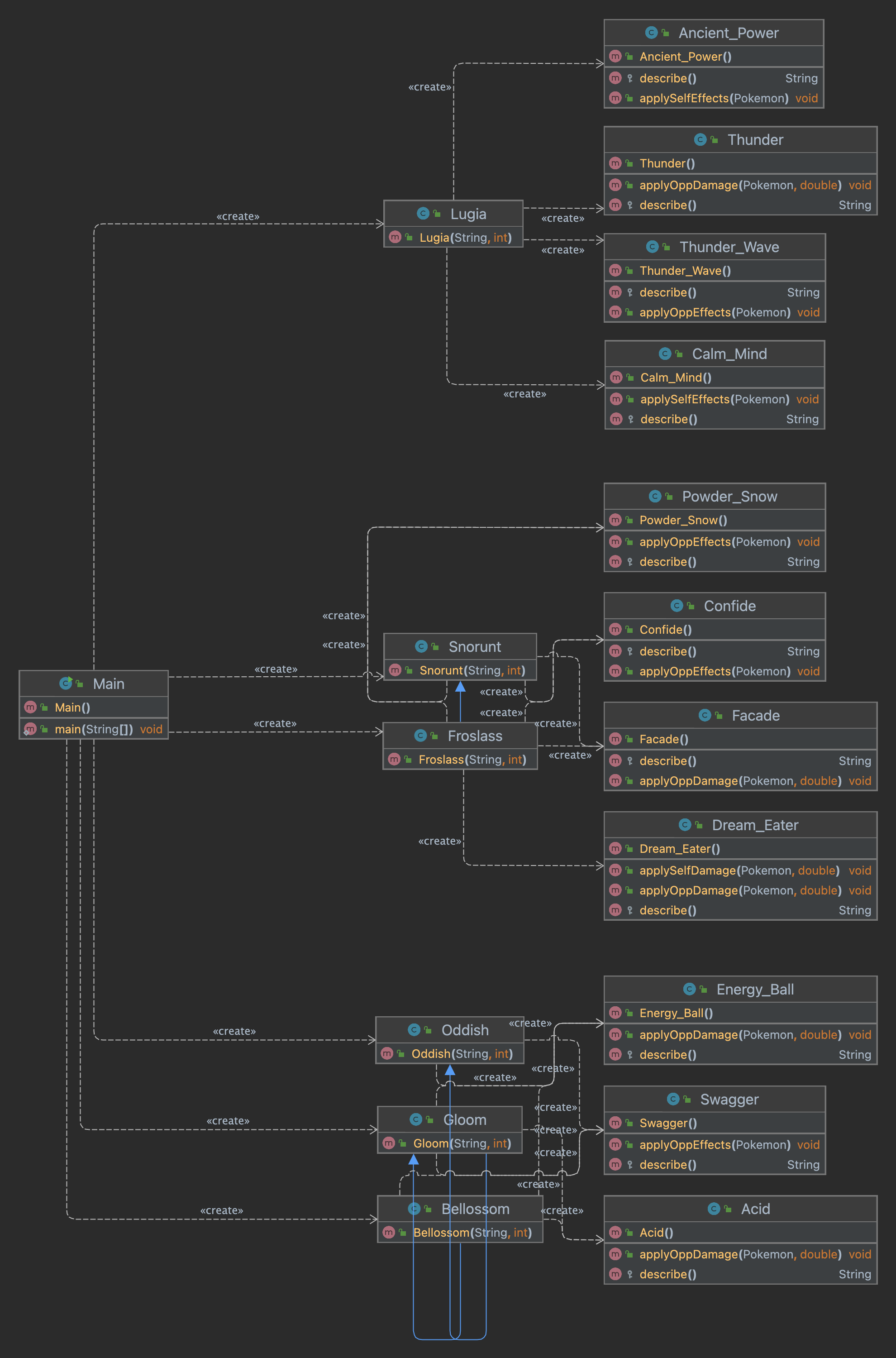


Рисунок 2

# Исходный код программы

**Main.java**

// lab2, variant: 94663

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

import Pokemons.\*;

import Moves.\*;

public class Main {

public static void main(String[] yolo) {

Battle b = new Battle();

Lugia p1 = new Lugia("(Лужа)", 10);

Snorunt p2 = new Snorunt("(Снорант)", 10);

Froslass p3 = new Froslass("(Фросласс)", 10);

Oddish p4 = new Oddish("(Оддиш)", 10);

Gloom p5 = new Gloom("(Глум)", 10);

Bellossom p6 = new Bellossom("(Белоссом)", 10);

b.addAlly(p1);

b.addAlly(p2);

b.addAlly(p3);

b.addFoe(p4);

b.addFoe(p5);

b.addFoe(p6);

b.go();

}

}

**Pokemons**

*Lugia*

package Pokemons;

import Moves.\*;

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

public class Lugia extends Pokemon {

public Lugia(String name, int level) {

super(name, level);

setType(Type.PSYCHIC, Type.FLYING);

setStats(106, 90, 130, 90, 154, 110);

setMove(new Ancient\_Power(), new Thunder(), new Thunder\_Wave(), new Calm\_Mind());

}

}

*Snorunt*

package Pokemons;

import Moves.\*;

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

public class Snorunt extends Pokemon {

public Snorunt(String name, int level) {

super(name, level);

setType(Type.ICE);

setStats(50, 50, 50, 50, 50, 50);

setMove(new Powder\_Snow(), new Confide(), new Facade());

}

}

*Froslass*

package Pokemons;

import Moves.\*;

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

public class Froslass extends Snorunt {

public Froslass(String name, int level) {

super(name, level);

setType(Type.ICE, Type.GHOST);

setStats(70, 80, 70, 80, 70, 110);

setMove(new Powder\_Snow(), new Confide(), new Facade(), new Dream\_Eater());

}

}

*Oddish*

package Pokemons;

import Moves.\*;

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

public class Oddish extends Pokemon {

public Oddish(String name, int level) {

super(name, level);

setType(Type.GRASS, Type.POISON);

setStats(45, 50, 55, 75, 65, 30);

setMove(new Energy\_Ball(), new Swagger());

}

}

*Gloom*

package Pokemons;

import Moves.\*;

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

public class Gloom extends Oddish {

public Gloom(String name, int level) {

super(name, level);

setType(Type.GRASS, Type.POISON);

setStats(60, 65, 70, 85, 75, 40);

setMove(new Energy\_Ball(), new Swagger(), new Acid());

}

}

*Bellossom*

package Pokemons;

import Moves.\*;

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

public class Bellossom extends Gloom {

public Bellossom(String name, int level) {

super(name, level);

setType(Type.GRASS);

setStats(75, 80, 95, 90, 100, 50);

setMove(new Energy\_Ball(), new Swagger(), new Acid(), new Swagger());

}

}

**Moves**

*Ancient power*

package Moves;

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

public class Ancient\_Power extends SpecialMove {

public Ancient\_Power() {

super(Type.ROCK, 60, 100);

}

private boolean flag;

@Override

public void applySelfEffects(Pokemon p) {

int chance = (int)(Math.random() \* 101);

if (chance <= 10) flag = true;

if (flag) {

p.setMod(Stat.ATTACK, +1);

p.setMod(Stat.DEFENSE, +1);

p.setMod(Stat.SPECIAL\_ATTACK, +1);

p.setMod(Stat.SPECIAL\_DEFENSE, +1);

p.setMod(Stat.SPEED, +1);

}

}

@Override

protected String describe() {

if (flag) return "повезло, покемон использует Ancient Power: увеличивает свои характеристики на 1 и атакует";

else return "наносит урон";

}

}

*Thunder*

package Moves;

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

public class Thunder extends SpecialMove {

public Thunder() {

super(Type.ELECTRIC, 110, 70);

}

private boolean flag;

@Override

public void applyOppDamage(Pokemon def, double damage) {

int chance = (int)(Math.random() \* 101);

if (chance <= 30 && !def.hasType(Type.ELECTRIC)) flag = true;

if (flag) Effect.paralyze(def);

}

@Override

protected String describe() {

if (flag) return "повезло, покемон использует Thunder: парализует и наносит урон";

else return "наносит урон";

}

}

*Thunder Wave*

package Moves;

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

public class Thunder\_Wave extends StatusMove {

public Thunder\_Wave() {

super(Type.ELECTRIC, 0, 90);

}

@Override

public void applyOppEffects(Pokemon p) {

Effect.paralyze(p);

}

@Override

protected String describe() {

return "использует Thunder Wave: парализует";

}

}

*Calm Mind*

package Moves;

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

public class Calm\_Mind extends StatusMove {

public Calm\_Mind() {

super(Type.PSYCHIC, 0, 0);

}

@Override

public void applySelfEffects(Pokemon p) {

p.setMod(Stat.SPECIAL\_ATTACK, +1);

p.setMod(Stat.SPECIAL\_DEFENSE, +1);

}

@Override

protected String describe() {

return "использует Calm Mind: увеличивает Специальную Атаку и Специальную Защиту на 1";

}

}

*Powder Snow*

package Moves;

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

public class Powder\_Snow extends SpecialMove {

public Powder\_Snow() {

super(Type.ICE, 40, 100);

}

private boolean flag;

@Override

public void applyOppEffects(Pokemon p) {

int chance = (int)(Math.random() \* 101);

if (chance <= 10) flag = true;

if (flag) Effect.freeze(p);

}

@Override

protected String describe() {

if (flag) return "повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон";

else return "наносит урон";

}

}

*Confide*

package Moves;

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

public class Confide extends StatusMove {

public Confide() {

super(Type.NORMAL, 0, 0);

}

@Override

public void applyOppEffects(Pokemon p) {

p.setMod(Stat.SPECIAL\_ATTACK, -1);

}

@Override

protected String describe() {

return "использует Confide и уменьшает специальную атаку на 1";

}

}

*Facade*

package Moves;

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

public class Facade extends PhysicalMove {

public Facade() {

super(Type.NORMAL, 70, 100);

}

private boolean flag;

@Override

public void applyOppDamage(Pokemon def, double damage) {

Status condition = def.getCondition();

if (condition.equals(Status.POISON) || condition.equals(Status.PARALYZE)) {

flag = true;

def.setMod(Stat.HP, +2 \* (int)Math.round(damage));

}

}

@Override

protected String describe() {

if (flag) return "наносит двойной урон";

else return "наносит урон";

}

}

*Dream Eater*

package Moves;

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

public class Dream\_Eater extends SpecialMove {

public Dream\_Eater() {

super(Type.PSYCHIC, 100, 100);

}

private boolean flag;

@Override

public void applyOppDamage(Pokemon def, double damage) {

Status condition = def.getCondition();

flag = true;

if (condition.equals(Status.SLEEP)) def.setMod(Stat.HP, (int)Math.round(damage));

}

@Override

public void applySelfDamage(Pokemon att, double damage) {

if (flag) att.setMod(Stat.HP, -((int)Math.round(damage) / 2));

}

@Override

protected String describe() {

if (flag) return "использует Dream Eater: наносит урон и востанавливает себе половину нанесенного урона";

else return "ничего не делает";

}

}

*Energy Ball*

package Moves;

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

public class Energy\_Ball extends SpecialMove {

public Energy\_Ball() {

super(Type.GRASS, 90, 100);

}

private boolean flag;

@Override

public void applyOppDamage(Pokemon def, double damage) {

int chance = (int)(Math.random() \* 101);

if (chance <= 10) flag = true;

if (flag) def.setMod(Stat.SPECIAL\_DEFENSE, -1);

}

@Override

protected String describe() {

if (flag) return "повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон";

else return "наносит урон";

}

}

*Swagger*

package Moves;

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

public class Swagger extends StatusMove {

public Swagger() {

super(Type.NORMAL, 0, 85);

}

@Override

public void applyOppEffects(Pokemon p) {

p.confuse();

p.setMod(Stat.ATTACK, 2);

}

@Override

protected String describe() {

return "использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза";

}

}

*Acid*

package Moves;

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

public class Acid extends SpecialMove {

public Acid() {

super(Type.POISON, 40,100);

}

private boolean flag;

@Override

public void applyOppDamage(Pokemon def, double damage) {

int chance = (int)(Math.random() \* 101);

if (chance <= 10) flag = true;

if (flag) def.setMod(Stat.SPECIAL\_DEFENSE, -1);

}

@Override

protected String describe() {

if (flag) return "повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон";

else return "наносит урон";

}

}

# Результат работы программы

Lugia (Лужа) из команды белых вступает в бой!

Oddish (Оддиш) из команды полосатых вступает в бой!

Lugia (Лужа) промахивается

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Lugia (Лужа) увеличивает атаку.

Lugia (Лужа) наносит урон.

Критический удар!

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Lugia (Лужа) увеличивает атаку.

Lugia (Лужа) наносит урон.

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Lugia (Лужа) увеличивает атаку.

Lugia (Лужа) наносит урон.

Oddish (Оддиш) теряет 8 здоровья.

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Lugia (Лужа) увеличивает атаку.

Lugia (Лужа) использует Thunder Wave: парализует.

Oddish (Оддиш) парализован

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Lugia (Лужа) увеличивает атаку.

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Lugia (Лужа) увеличивает атаку.

Lugia (Лужа) промахивается

Oddish (Оддиш) наносит урон.

Критический удар!

Lugia (Лужа) растерянно попадает по себе.

Lugia (Лужа) теряет 13 здоровья.

Lugia (Лужа) наносит урон.

Oddish (Оддиш) наносит урон.

Критический удар!

Lugia (Лужа) растерянно попадает по себе.

Lugia (Лужа) теряет 15 здоровья.

Lugia (Лужа) растерянно попадает по себе.

Lugia (Лужа) теряет 10 здоровья.

Oddish (Оддиш) наносит урон.

Критический удар!

Lugia (Лужа) наносит урон.

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Lugia (Лужа) увеличивает атаку.

Lugia (Лужа) растерянно попадает по себе.

Lugia (Лужа) теряет 9 здоровья.

Lugia (Лужа) теряет сознание.

Snorunt (Снорант) из команды белых вступает в бой!

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Snorunt (Снорант) увеличивает атаку.

Snorunt (Снорант) наносит урон.

Oddish (Оддиш) теряет 20 здоровья.

Oddish (Оддиш) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Snorunt (Снорант) увеличивает атаку.

Snorunt (Снорант) наносит урон.

Oddish (Оддиш) теряет 24 здоровья.

Oddish (Оддиш) теряет сознание.

Gloom (Глум) из команды полосатых вступает в бой!

Snorunt (Снорант) наносит урон.

Gloom (Глум) теряет 11 здоровья.

Gloom (Глум) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Snorunt (Снорант) увеличивает атаку.

Snorunt (Снорант) растерянно попадает по себе.

Snorunt (Снорант) теряет 7 здоровья.

Gloom (Глум) наносит урон.

Snorunt (Снорант) наносит урон.

Gloom (Глум) теряет 17 здоровья.

Gloom (Глум) наносит урон.

Snorunt (Снорант) наносит урон.

Gloom (Глум) теряет 17 здоровья.

Gloom (Глум) теряет сознание.

Bellossom (Белоссом) из команды полосатых вступает в бой!

Bellossom (Белоссом) наносит урон.

Snorunt (Снорант) промахивается

Snorunt (Снорант) промахивается

Bellossom (Белоссом) наносит урон.

Bellossom (Белоссом) наносит урон.

Snorunt (Снорант) наносит двойной урон.

Bellossom (Белоссом) наносит урон.

Bellossom (Белоссом) наносит урон.

Snorunt (Снорант) наносит двойной урон.

Snorunt (Снорант) наносит двойной урон.

Snorunt (Снорант) наносит двойной урон.

Snorunt (Снорант) наносит двойной урон.

Bellossom (Белоссом) наносит урон.

Bellossom (Белоссом) наносит урон.

Snorunt (Снорант) наносит урон.

Snorunt (Снорант) теряет 5 здоровья.

Snorunt (Снорант) наносит урон.

Snorunt (Снорант) теряет 4 здоровья.

Bellossom (Белоссом) наносит урон.

Bellossom (Белоссом) наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) наносит урон.

Snorunt (Снорант) уменьшает специальную защиту.

Snorunt (Снорант) наносит двойной урон.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Snorunt (Снорант) уменьшает специальную защиту.

Snorunt (Снорант) промахивается

Snorunt (Снорант) промахивается

Snorunt (Снорант) промахивается

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Bellossom (Белоссом) увеличивает атаку.

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Bellossom (Белоссом) увеличивает атаку.

Snorunt (Снорант) повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон.

Snorunt (Снорант) теряет 3 здоровья.

Snorunt (Снорант) повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон.

Snorunt (Снорант) теряет 3 здоровья.

Snorunt (Снорант) повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон.

Snorunt (Снорант) теряет 2 здоровья.

Snorunt (Снорант) повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон.

Snorunt (Снорант) теряет 3 здоровья.

Snorunt (Снорант) промахивается

Snorunt (Снорант) промахивается

Bellossom (Белоссом) повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон.

Snorunt (Снорант) уменьшает специальную защиту.

Snorunt (Снорант) наносит двойной урон.

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Bellossom (Белоссом) увеличивает атаку.

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Bellossom (Белоссом) увеличивает атаку.

Snorunt (Снорант) промахивается

Snorunt (Снорант) промахивается

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Snorunt (Снорант) промахивается

Snorunt (Снорант) промахивается

Snorunt (Снорант) повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон.

Bellossom (Белоссом) теряет 5 здоровья.

Bellossom (Белоссом) замерзает

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Snorunt (Снорант) уменьшает специальную защиту.

Bellossom (Белоссом) оттаивает

Snorunt (Снорант) промахивается

Bellossom (Белоссом) растерянно попадает по себе.

Bellossom (Белоссом) теряет 10 здоровья.

Snorunt (Снорант) повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон.

Snorunt (Снорант) восстанавливает 6 здоровья.

Snorunt (Снорант) повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон.

Критический удар!

Snorunt (Снорант) восстанавливает 11 здоровья.

Snorunt (Снорант) повезло, покемон использует Powder Snow: замораживает противника и наносит ему урон.

Bellossom (Белоссом) теряет 5 здоровья.

Bellossom (Белоссом) замерзает

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Snorunt (Снорант) увеличивает атаку.

Bellossom (Белоссом) оттаивает

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Snorunt (Снорант) наносит двойной урон.

Критический удар!

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Snorunt (Снорант) увеличивает атаку.

Bellossom (Белоссом) повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Snorunt (Снорант) уменьшает специальную защиту.

Snorunt (Снорант) промахивается

Snorunt (Снорант) промахивается

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Snorunt (Снорант) увеличивает атаку.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Bellossom (Белоссом) уменьшает специальную защиту.

Snorunt (Снорант) наносит двойной урон.

Bellossom (Белоссом) повезло, покемон использует Energy Ball: уменьшает Специальную Защиту противника на 1 и наносит урон.

Snorunt (Снорант) уменьшает специальную защиту.

Snorunt (Снорант) промахивается

Bellossom (Белоссом) повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон.

Snorunt (Снорант) уменьшает специальную защиту.

Snorunt (Снорант) промахивается

Snorunt (Снорант) промахивается

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Bellossom (Белоссом) увеличивает атаку.

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Bellossom (Белоссом) увеличивает атаку.

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Bellossom (Белоссом) увеличивает атаку.

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Bellossom (Белоссом) увеличивает атаку.

Snorunt (Снорант) промахивается

Bellossom (Белоссом) использует Swagger: вызывает растерянность и увеличивает атаку соперника в 2 раза.

Snorunt (Снорант) увеличивает атаку.

Snorunt (Снорант) растерянно попадает по себе.

Snorunt (Снорант) теряет 10 здоровья.

Snorunt (Снорант) растерянно попадает по себе.

Snorunt (Снорант) теряет 12 здоровья.

Оба покемона теряют сознание.

Froslass (Фросласс) из команды белых вступает в бой!

Froslass (Фросласс) наносит урон.

Bellossom (Белоссом) повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон.

Froslass (Фросласс) уменьшает специальную защиту.

Froslass (Фросласс) ничего не делает.

Froslass (Фросласс) теряет 17 здоровья.

Bellossom (Белоссом) повезло, покемон использует Acid: уменьшает Специальную Защиту противника на 1 и наносит урон.

Froslass (Фросласс) уменьшает специальную защиту.

Froslass (Фросласс) ничего не делает.

Froslass (Фросласс) теряет 22 здоровья.

Froslass (Фросласс) теряет сознание.

В команде белых не осталось покемонов.

Команда полосатых побеждает в этом бою!

Process finished with exit code 0

# Вывод

В ходе выполнения данной лабораторной работы я научился подключать внешние jar-файлы к программе, создавать Package’ы и классы, а затем работать с созданными классами, с внешними базами данных для поиска информации для создания этих классов.

# Список литературы

**Database Pockemon** http://pokemondb.net/ [Книга].

**Oracle** Java Documentation. - Т. https://docs.oracle.com/en/java/.

**Pockemon Документация** https://se.ifmo.ru/~tony/doc/ [Книга].

**Михалевский Петр** Легкий старт в Java <...>. - [б.м.] : Stepik. - Т. https://stepik.org/course/90684/syllabus.