Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего образования

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И  
ОПТИКИ  
Факультет систем управления и робототехники

Отчёт по лабораторной работе №1

“Программирование”

Вариант №312637

Преподаватель:

Сорокин Роман Борисович

Выполнила:

Парфенова Ольга Сергеевна

Гр. R3142

Санкт-Петербург, 2020

1. Текст задания

На основе базового класса Pokemon написать свои классы для заданных видов покемонов. Каждый вид покемона должен иметь один или два типа и стандартные базовые характеристики:

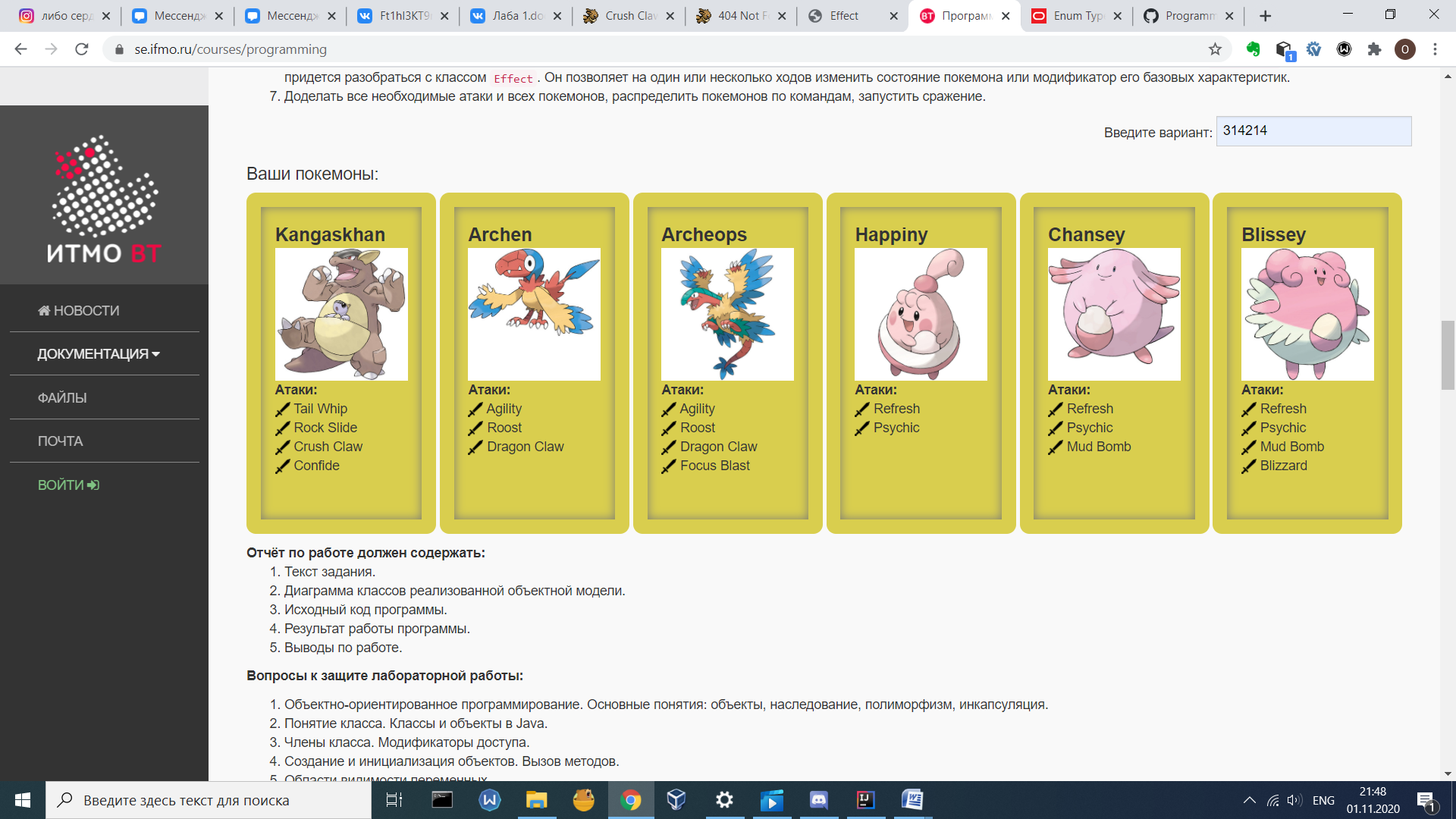
* очки здоровья (HP)
* атака (attack)
* защита (defense)
* специальная атака (special attack)
* специальная защита (special defense)
* скорость (speed)

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.

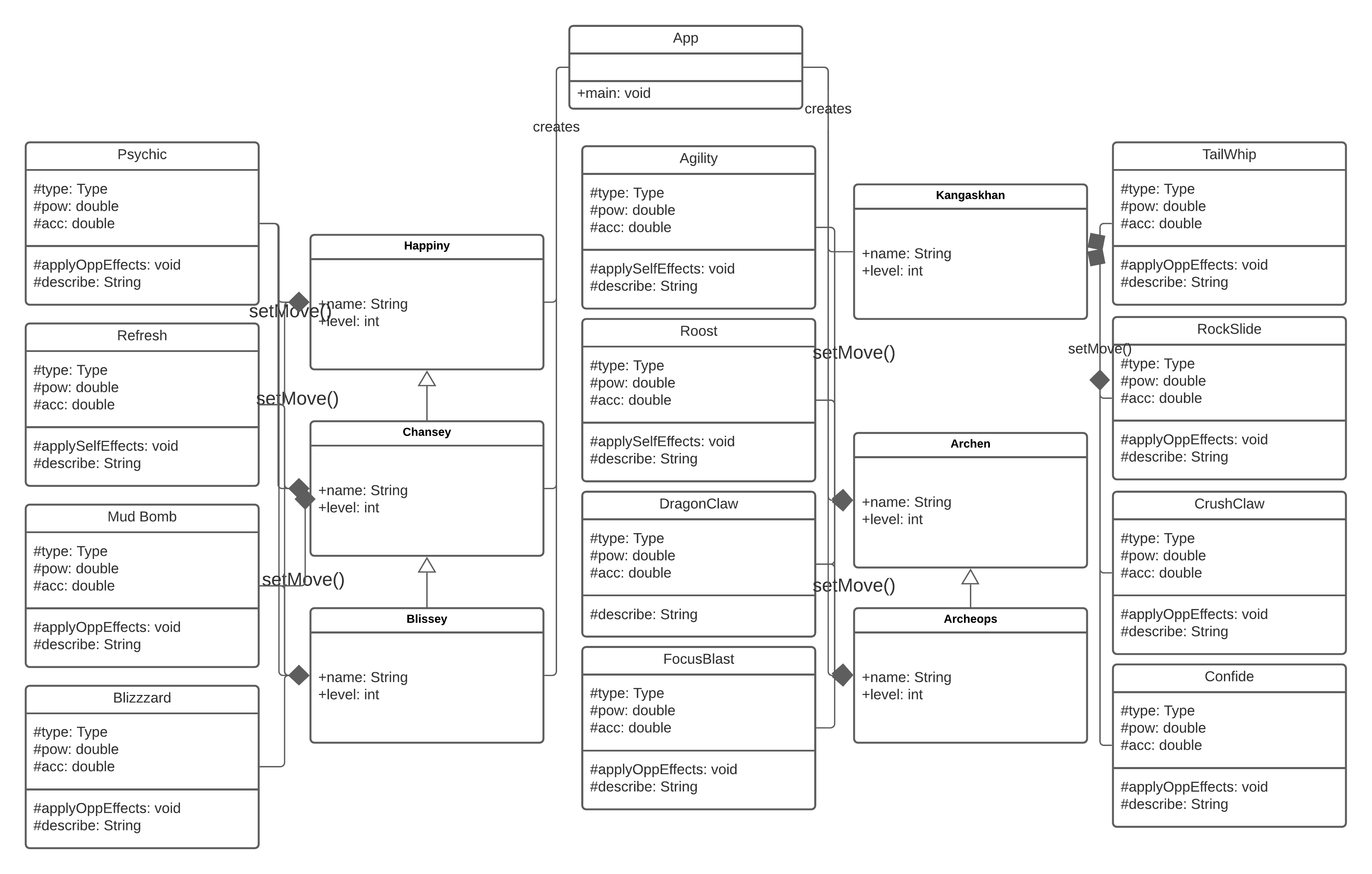
Атака должна иметь стандартные тип, силу (power) и точность (accuracy). Должны быть реализованы стандартные эффекты атаки. Назначить каждому виду покемонов атаки в соответствии с вариантом. Уровень покемона выбирается минимально необходимым для всех реализованных атак.

Используя класс симуляции боя Battle, создать 2 команды покемонов (каждый покемон должен иметь имя) и запустить бой.

Мои покемоны:



2. Диаграмма классов



3. Исходный код программы

**App.java**

package Lab2\_gradle;

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

public class App {

public static void main(String[] args) {

Battle b = new Battle();

b.addAlly(new Happiny("Happiny", 1));

b.addAlly(new Archeops("Archeops", 1));

b.addAlly(new Kangaskhan("Kangaskhan", 1));

b.addFoe(new Chansey("Chansey", 1));

b.addFoe(new Archen("Archen", 1));

b.addFoe(new Blissey("Blissey", 1));

b.go();

}

}

**Pokemons.java**

package Lab2\_gradle;

import ru.ifmo.se.pokemon.Pokemon;

import ru.ifmo.se.pokemon.Type;

class Happiny extends Pokemon {

public Happiny(String name, int level) {

super(name, level);

setStats(100, 5, 5, 15, 65, 30);

setType(Type.NORMAL);

setMove(new Refresh(), new Psychic());

}

}

class Chansey extends Happiny {

public Chansey(String name, int level) {

super(name, level);

setStats(250, 5, 5, 35, 105, 50);

setType(Type.NORMAL);

setMove(new Refresh(), new Psychic(), new MudBomb());

}

}

class Blissey extends Chansey {

public Blissey(String name, int level) {

super(name, level);

setStats(255, 10, 10, 75, 135, 55);

setType(Type.NORMAL);

setMove(new Refresh(), new Psychic(), new MudBomb(), new Blizzard());

}

}

class Archen extends Pokemon {

public Archen(String name, int level) {

super(name, level);

setStats(55, 112, 45, 74, 45, 70);

setType(Type.ROCK, Type.FLYING);

setMove(new Agility(), new Roost(), new DragonClaw());

}

}

class Archeops extends Archen {

public Archeops(String name, int level) {

super(name, level);

setStats(75, 140, 65, 112, 65, 110);

setType(Type.FLYING, Type.ROCK);

setMove(new Agility(), new Roost(), new DragonClaw(), new FocusBlast());

}

}

class Kangaskhan extends Pokemon {

public Kangaskhan(String name, int level) {

super(name, level);

setStats(105, 95, 80, 40, 80, 90);

setType(Type.NORMAL);

setMove(new TailWhip(), new RockSlide(), new CrushClaw(), new Confide(), new Confide());

}

}

**BlisseyAttacks.java**

package Lab2\_gradle;

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

class Psychic extends SpecialMove {

protected Psychic() {

super(Type.PSYCHIC, 90, 100);

}

protected void applyOppEffects(Pokemon pokemon) {

Effect e = new Effect().chance(0.1).stat(Stat.SPECIAL\_DEFENSE, -1);

pokemon.addEffect(e);

}

protected String describe() {

return ("применяет Psychic");

}

}

class Refresh extends StatusMove {

protected Refresh() {

super(Type.NORMAL, 0, 100);

}

protected void applySelfEffects(Pokemon pokemon) {

Status c = pokemon.getCondition();

Effect e = new Effect().condition(Status.NORMAL);

if (c.equals(Status.PARALYZE) || c.equals(Status.BURN) || c.equals(Status.POISON)) pokemon.addEffect(e);

}

protected String describe() {

return ("применяет Refresh");

}

}

class MudBomb extends SpecialMove {

protected MudBomb() {

super(Type.GROUND, 65, 85);

}

protected void applyOppEffects(Pokemon pokemon) {

Effect e = new Effect().chance(0.3).stat(Stat.ACCURACY, -1);

pokemon.addEffect(e);

}

protected String describe() {

return ("применяет Mud Bomb");

}

}

class Blizzard extends SpecialMove {

protected Blizzard() {

super(Type.ICE, 110, 70);

}

protected void applyOppEffects(Pokemon pokemon) {

Effect e = new Effect().chance(0.1);

Effect.freeze(pokemon);

}

protected String describe() {

return ("применяет Blizzard");

}

}

**ArcheopsAttacks.java**

ackage Lab2\_gradle;

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

class Agility extends StatusMove {

protected Agility() {

super(Type.PSYCHIC, 0, 100);

}

protected void applySelfEffects(Pokemon pokemon) {

pokemon.setMod(Stat.SPEED, 2);

}

protected String describe() {

return ("применяет Agility");

}

}

class Roost extends StatusMove {

protected Roost() {

super(Type.FLYING, 0, 100);

}

protected void applySelfEffects(Pokemon pokemon) {

double max\_hp = pokemon.getStat(Stat.HP);

pokemon.setMod(Stat.HP, (int) -max\_hp / 2);

}

protected String describe() {

return ("применяет Roost");

}

}

class DragonClaw extends PhysicalMove {

protected DragonClaw() {

super(Type.DRAGON, 80, 100);

}

protected String describe() {

return ("применяет Dragon Claw");

}

}

class FocusBlast extends SpecialMove {

protected FocusBlast() {

super(Type.FIGHTING, 120, 70);

}

protected void applyOppEffects(Pokemon pokemon) {

Effect e = new Effect().chance(0.1).stat(Stat.SPECIAL\_DEFENSE, -1);

pokemon.addEffect(e);

}

protected String describe() {

return ("применяет Focus Blast");

}

}

**KangaskhanAttacks.java**

package Lab2\_gradle;

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

class TailWhip extends StatusMove {

protected TailWhip() {

super(Type.NORMAL, 0, 100);

}

protected void applyOppEffects(Pokemon pokemon) {

pokemon.setMod(Stat.DEFENSE, -1);

}

protected String describe() {

return ("применяет Tail Whip");

}

}

class RockSlide extends PhysicalMove {

protected RockSlide() {

super(Type.ROCK, 75, 90);

}

protected void applyOppEffects(Pokemon pokemon) {

Effect e = new Effect().chance(0.3);

e.flinch(pokemon);

}

protected String describe() {

return ("применяет Rock Slide");

}

}

class CrushClaw extends PhysicalMove {

protected CrushClaw() {

super(Type.NORMAL, 75, 95);

}

protected void applyOppEffects(Pokemon pokemon) {

Effect e = new Effect().chance(0.5).stat(Stat.DEFENSE, -1);

pokemon.addEffect(e);

}

protected String describe() {

return ("применяет Crush Claw");

}

}

class Confide extends StatusMove {

protected Confide() {

super(Type.NORMAL, 0, 100);

}

protected void applyOppEffects(Pokemon pokemon) {

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

}

protected String describe() {

return ("применяет Confide");

}

}

4. Результат работы программы

Happiny Happiny из команды зеленых вступает в бой!

Chansey Chansey из команды фиолетовых вступает в бой!

Chansey Chansey применяет Refresh.

Happiny Happiny применяет Refresh.

Chansey Chansey применяет Mud Bomb.

Happiny Happiny теряет 4 здоровья.

Happiny Happiny применяет Refresh.

Chansey Chansey применяет Psychic.

Happiny Happiny теряет 6 здоровья.

Happiny Happiny применяет Psychic.

Chansey Chansey теряет 4 здоровья.

Chansey Chansey применяет Refresh.

Happiny Happiny применяет Refresh.

Chansey Chansey применяет Refresh.

Happiny Happiny применяет Refresh.

Chansey Chansey применяет Mud Bomb.

Happiny Happiny теряет 4 здоровья.

Happiny Happiny теряет сознание.

Archeops Archeops из команды зеленых вступает в бой!

Archeops Archeops применяет Agility.

Archeops Archeops увеличивает скорость.

Chansey Chansey применяет Mud Bomb.

Archeops Archeops теряет 1 здоровья.

Archeops Archeops не замечает воздействие типа GROUND

Archeops Archeops применяет Focus Blast.

Chansey Chansey теряет 11 здоровья.

Chansey Chansey применяет Psychic.

Archeops Archeops теряет 5 здоровья.

Archeops Archeops применяет Roost.

Archeops Archeops восстанавливает 6 здоровья.

Chansey Chansey применяет Refresh.

Archeops Archeops применяет Focus Blast.

Chansey Chansey теряет 8 здоровья.

Chansey Chansey теряет сознание.

Archen Archen из команды фиолетовых вступает в бой!

Archeops Archeops применяет Agility.

Archeops Archeops увеличивает скорость.

Archen Archen применяет Roost.

Archen Archen восстанавливает 6 здоровья.

Archeops Archeops применяет Agility.

Archeops Archeops увеличивает скорость.

Archen Archen применяет Agility.

Archen Archen увеличивает скорость.

Archeops Archeops применяет Roost.

Archeops Archeops восстанавливает 6 здоровья.

Archen Archen применяет Dragon Claw.

Archeops Archeops теряет 7 здоровья.

Archeops Archeops применяет Agility.

Archeops Archeops увеличивает скорость.

Archen Archen применяет Agility.

Archen Archen увеличивает скорость.

Archeops Archeops применяет Focus Blast.

Archen Archen теряет 7 здоровья.

Archen Archen применяет Agility.

Archen Archen увеличивает скорость.

Archeops Archeops применяет Focus Blast.

Archen Archen теряет 5 здоровья.

Archen Archen применяет Dragon Claw.

Archeops Archeops теряет 5 здоровья.

Archeops Archeops применяет Focus Blast.

Archen Archen теряет 8 здоровья.

Archen Archen теряет сознание.

Blissey Blissey из команды фиолетовых вступает в бой!

Archeops Archeops применяет Roost.

Archeops Archeops восстанавливает 6 здоровья.

Blissey Blissey применяет Psychic.

Archeops Archeops теряет 4 здоровья.

Archeops Archeops применяет Focus Blast.

Blissey Blissey теряет 14 здоровья.

Blissey Blissey применяет Blizzard.

Archeops Archeops теряет 14 здоровья.

Archeops Archeops замерзает

Archeops Archeops теряет сознание.

Kangaskhan Kangaskhan из команды зеленых вступает в бой!

Kangaskhan Kangaskhan применяет Tail Whip.

Blissey Blissey уменьшает защиту.

Blissey Blissey применяет Psychic.

Kangaskhan Kangaskhan теряет 5 здоровья.

Kangaskhan Kangaskhan применяет Tail Whip.

Blissey Blissey уменьшает защиту.

Blissey Blissey применяет Blizzard.

Kangaskhan Kangaskhan теряет 5 здоровья.

Kangaskhan Kangaskhan замерзает

Blissey Blissey применяет Psychic.

Kangaskhan Kangaskhan теряет 5 здоровья.

Kangaskhan Kangaskhan теряет сознание.

В команде зеленых не осталось покемонов.

Команда фиолетовых побеждает в этом бою!

5. Выводы

В ходе работы над данной программой я узнала основы объектно-ориентированного программирования, научилась работать с классами, их методами и модификаторами доступа, модификаторами final и static, импортировать внешние jar-файлы и использовать систему сборки Gradle.