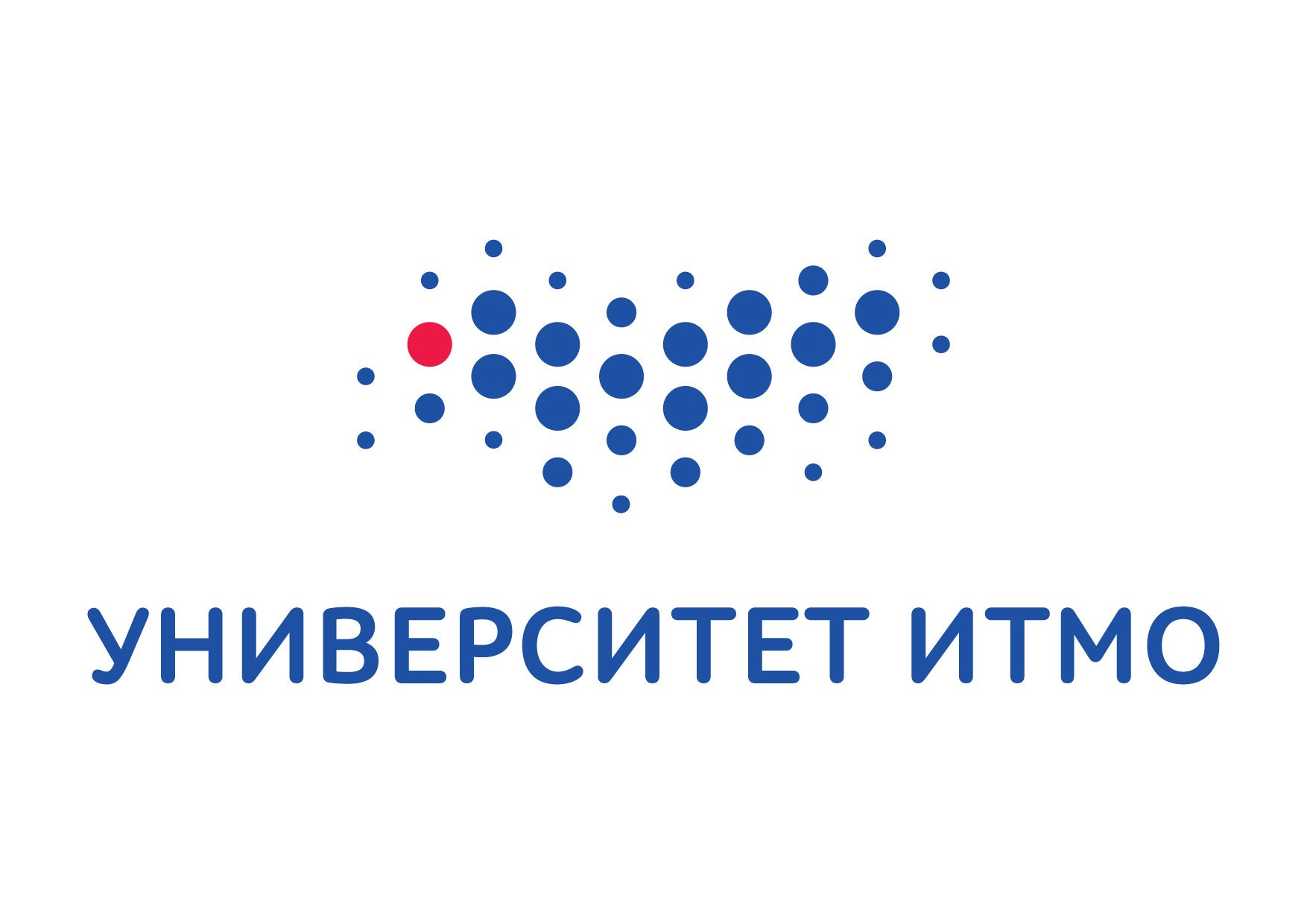
Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет Программной Инженерии и Компьютерной Техники

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



### Лабораторная работа № 2

### по дисциплине “ Программирование”

Вариант 1234

Выполнила: Армут Алина Юсуфовна

Р3114

Преподаватель:

Райла Мартин

г. Санкт-Петербург

2022

**Задание:**

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.

Базовые классы и симулятор сражения находятся в [jar-архиве](https://se.ifmo.ru/documents/10180/660917/Pokemon.jar/a7ce60af-6ee6-47d0-a95e-e5ed9a697bd2) (обновлен 9.10.2018, исправлен баг с добавлением атак и кодировкой). Документация в формате javadoc - [здесь](https://se.ifmo.ru/~tony/doc/).

Информацию о покемонах, цепочках эволюции и атаках можно найти на сайтах <http://poke-universe.ru>, <http://pokemondb.net>,<http://veekun.com/dex/pokemon>

**Покемоны**

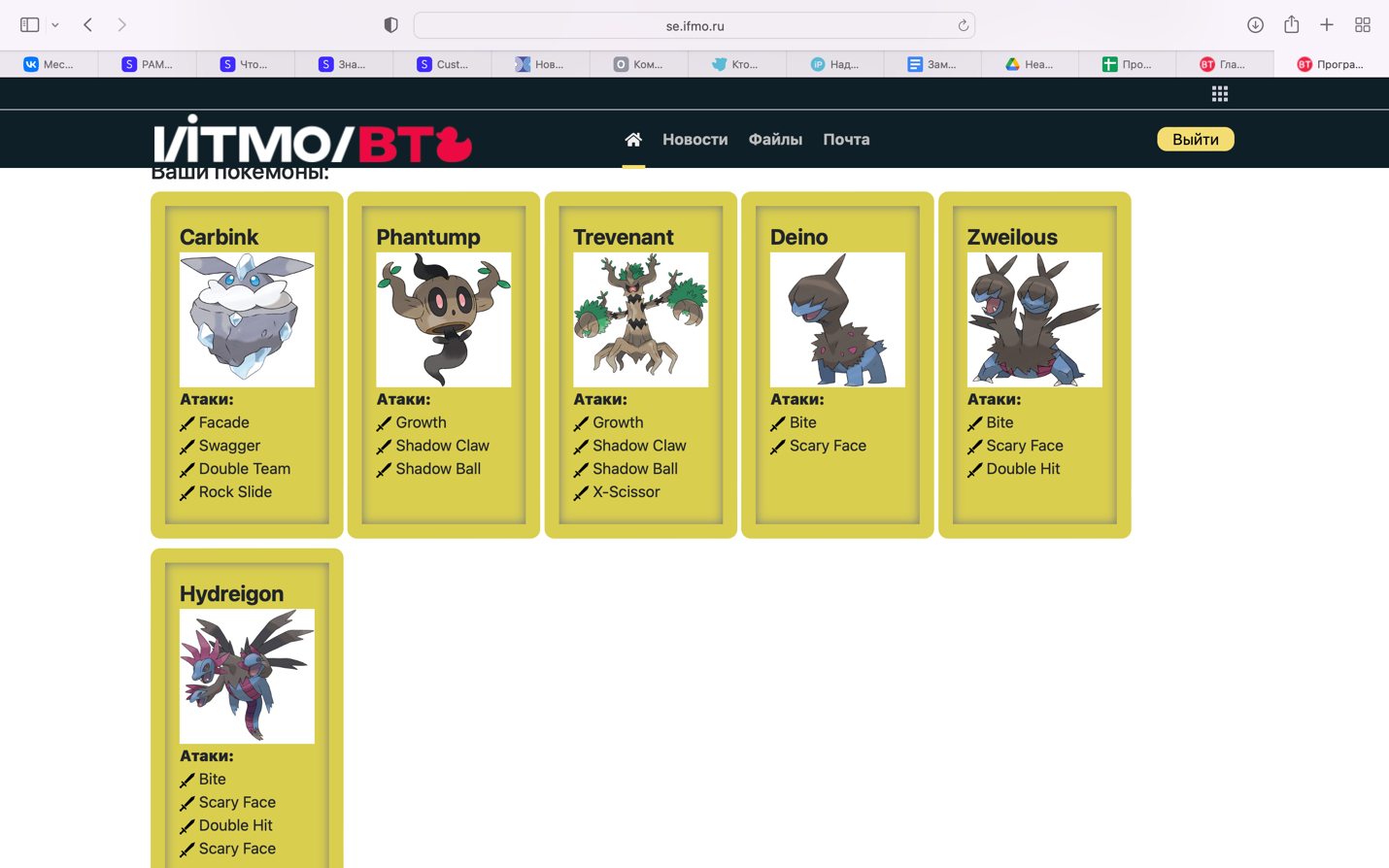
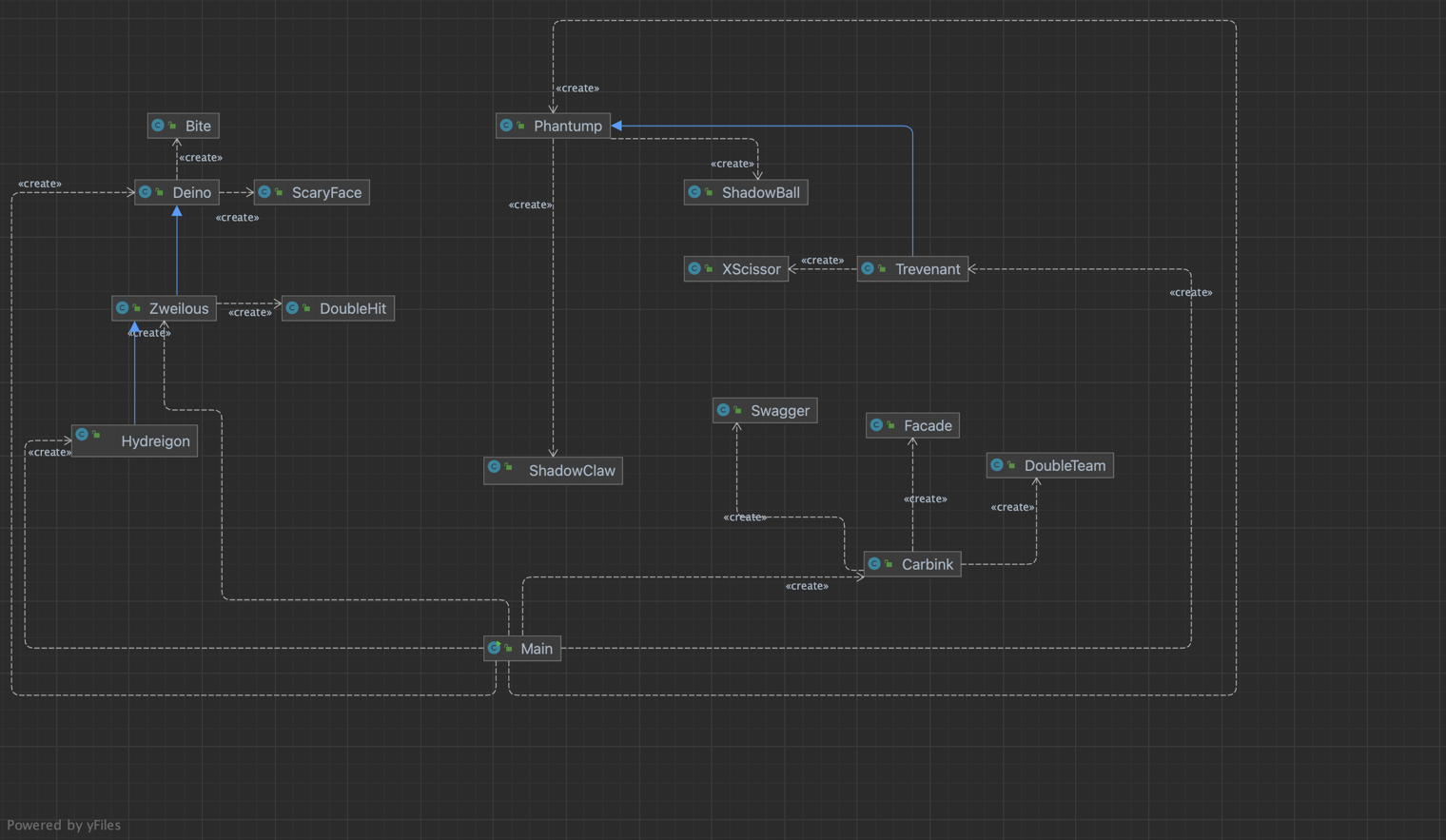
****

Диаграмма классов:



**Исходный код:**

Main.java

import ru.ifmo.se.pokemon.\*;  
import pokemons.\*;  
  
  
public class Main {  
 public static void main(String[] args) {  
 Battle b = new Battle();  
 Carbink p1 = new Carbink("Vasya", 1);  
 Hydreigon p2 = new Hydreigon("Katya",3);  
 Deino p3 = new Deino("Sasha",1);  
 Phantump p4 = new Phantump("Mia",2);  
 Trevenant p5 = new Trevenant("Roma",3);  
 Zweilous p6 = new Zweilous("Kirill",2);  
  
 b.addAlly(p1);  
 b.addAlly(p2);  
 b.addAlly(p3);  
 b.addFoe(p4);  
 b.addFoe(p5);  
 b.addFoe(p6);  
 b.go();  
  
 }  
}

Carbink.java

package pokemons;  
  
import ru.ifmo.se.pokemon.\*;  
import attacks.\*;  
  
  
public class Carbink extends Pokemon {  
  
  
 public Carbink(String name, int level) {  
 super(name,level);  
 setStats(50,50,150,50,50,500);  
 setType(Type.*ROCK*,Type.*FAIRY*);  
 setMove(new Facade(), new Swagger(), new DoubleTeam(), new Swagger());  
  
  
 }  
}

Phantump.java

package pokemons;  
  
import attacks.\*;  
import ru.ifmo.se.pokemon.\*;  
  
public class Phantump extends Pokemon {  
  
  
 public Phantump(String name, int level) {  
 super(name,level);  
 setStats(43,70,48,50,60,38);  
 setType(Type.*GHOST*,Type.*GRASS*);  
 setMove(new Growth(), new ShadowClaw(), new ShadowBall());  
  
  
 }  
}

Trevenant.java

package pokemons;  
  
import ru.ifmo.se.pokemon.\*;  
import attacks.\*;  
  
public class Trevenant extends Phantump {  
  
  
 public Trevenant(String name, int level) {  
 super(name,level);  
 setStats(85,110,76,65,82,56);  
 setType(Type.*GHOST*,Type.*GRASS*);  
 addMove(new XScissor());  
  
  
 }  
}

Deino.java

package pokemons;  
  
import ru.ifmo.se.pokemon.\*;  
import attacks.\*;  
  
  
public class Deino extends Pokemon {  
  
  
 public Deino(String name, int level) {  
 super(name,level);  
 setStats(52,65,50,45,50 ,38);  
 setType(Type.*DARK*,Type.*FAIRY*);  
 setMove(new Bite(), new ScaryFace());  
 }  
}

Zweilous.java

package pokemons;  
  
import ru.ifmo.se.pokemon.\*;  
import attacks.\*;  
  
public class Zweilous extends Deino {  
  
  
 public Zweilous(String name, int level) {  
 super(name,level);  
 setStats(72,85,70,65,70,58);  
 setType(Type.*DARK*,Type.*DRAGON*);  
 addMove(new DoubleHit());  
  
  
 }  
}

Hydreigon.java

package pokemons;  
  
import ru.ifmo.se.pokemon.\*;  
  
  
public class Hydreigon extends Zweilous {  
  
  
 public Hydreigon(String name, int level) {  
 super(name,level);  
 setStats(92,105,90,125,90,98);  
 setType(Type.*DARK*,Type.*DRAGON*);  
  
  
  
 }  
}

Bite.java

package attacks;  
  
import ru.ifmo.se.pokemon.\*;  
  
public class Bite extends PhysicalMove {  
 public Bite() {  
 super(Type.*DARK*, 60, 100);  
 }  
 @Override  
 protected void applyOppEffects(Pokemon p) {  
 if (Math.*random*() < 0.3) {  
 Effect.*flinch*(p);  
 }  
 }  
 @Override  
 protected String describe(){  
 return " использует Bite";  
 }  
  
}

DoubleHit.java

package attacks;  
  
import ru.ifmo.se.pokemon.\*;  
  
public class DoubleHit extends PhysicalMove {  
 public DoubleHit() {  
 super(Type.*NORMAL*, 35, 90);  
 }  
 @Override  
 protected void applyOppDamage (Pokemon def, double damage){  
 def.setMod(Stat.*HP*, (int) (damage\*2));  
  
 }  
 @Override  
 protected String describe(){  
 return "использует DoubleHit";  
 }  
}

DoubleTeam.java

package attacks;  
  
  
import ru.ifmo.se.pokemon.\*;  
  
public class DoubleTeam extends StatusMove {  
 public DoubleTeam() {  
 super(Type.*NORMAL*, 0, 100);  
 }  
 @Override  
 protected void applyOppEffects ( Pokemon p){  
 p.setMod(Stat.*EVASION*, 1 );  
 }  
 @Override  
 protected String describe(){  
 return " использует DoubleTeam";  
 }  
  
}

Facade.java

package attacks;  
  
import ru.ifmo.se.pokemon.\*;  
  
  
public class Facade extends PhysicalMove {  
 public Facade() {  
 super(Type.*NORMAL*, 70, 100);  
  
 }  
 @Override  
 protected void applyOppDamage(Pokemon def, double damage) {  
 super.applyOppDamage(def, damage);  
 if ((def.getCondition() == Status.*BURN*) || (def.getCondition() == Status.*PARALYZE*) || (def.getCondition() == Status.*POISON*)) {  
 def.setMod(Stat.*HP*, (int) Math.*round*(2\*damage));  
  
 }  
  
 }  
 @Override  
 protected String describe() {  
 return "Использует Facade";  
 }  
  
}

Growth.java

package attacks;  
  
  
import ru.ifmo.se.pokemon.\*;  
  
  
public class Growth extends StatusMove {  
 public Growth() {  
 super(Type.*NORMAL*, 0, 0);  
 }  
 @Override  
 protected void applySelfEffects(Pokemon p){  
 p.setMod(Stat.*ATTACK*,1);  
 p.setMod(Stat.*SPECIAL\_ATTACK*,1);  
 }  
 @Override  
 protected String describe(){  
 return "использует Growth";  
 }  
}

RockSlide.java

package attacks;  
  
  
import ru.ifmo.se.pokemon.\*;  
  
public class RockSlide extends PhysicalMove {  
 public RockSlide() {  
 super(Type.*ROCK*,75 , 90);  
 }  
 @Override  
 protected void applyOppEffects(Pokemon p) {  
 if (Math.*random*() < 0.3) {  
 Effect.*flinch*(p);  
 }  
 }  
 @Override  
 protected String describe(){  
 return "использует RockSlide";  
 }  
  
}

ScaryFace.java

package attacks;  
  
  
import ru.ifmo.se.pokemon.\*;  
  
public class ScaryFace extends StatusMove {  
 public ScaryFace() {  
 super(Type.*NORMAL*, 0,100 );  
 }  
 @Override  
 protected void applyOppEffects ( Pokemon p) {  
 p.setMod(Stat.*SPEED*, 2);  
 }  
 @Override  
 protected String describe(){  
 return "используе ScaryFace";  
 }  
}

ShadowBall.java

package attacks;  
  
  
import ru.ifmo.se.pokemon.\*;  
  
public class ShadowBall extends SpecialMove {  
 public ShadowBall() {  
 super(Type.*GHOST*, 80, 100);  
 }  
 @Override  
 protected void applyOppEffects(Pokemon p){  
 if (Math.*random*() <= 0.2) p.setMod(Stat.*SPECIAL\_DEFENSE*, -1);  
 }  
 @Override  
 protected String describe(){  
 return "использует ShadowBall";  
 }  
}

ShadowClaw.java

package attacks;  
  
import ru.ifmo.se.pokemon.\*;  
  
public class ShadowClaw extends PhysicalMove {  
 public ShadowClaw() {  
 super(Type.*GHOST*, 70, 100);  
 }  
 @Override  
 protected void applyOppDamage (Pokemon def, double damage) {  
 def.setMod(Stat.*HP*,(int)Math.*round*(damage)/8);  
  
 }  
 @Override  
 protected String describe(){  
 return "Использует ShadowClaw";  
 }  
}

Swagger.java

package attacks;  
  
import ru.ifmo.se.pokemon.\*;  
  
public class Swagger extends StatusMove {  
 public Swagger() {  
 super(Type.*NORMAL*, 0, 85);  
 }  
 @Override  
 protected void applyOppEffects(Pokemon p) {  
 p.setMod(Stat.*ATTACK*,2);  
 Effect.*confuse*(p);  
 }  
 @Override  
 protected String describe(){  
 return "Использует Swagger";  
 }  
}

XScissor.java

package attacks;  
  
import ru.ifmo.se.pokemon.\*;  
  
public class XScissor extends PhysicalMove {  
 public XScissor() {  
 super(Type.*BUG*, 80, 100);  
 }  
 @Override  
 protected void applyOppDamage (Pokemon def, double damage) {  
 def.setMod(Stat.*HP*, (int) Math.*round*(damage));  
 }  
 @Override  
 protected String describe(){  
 return "Использует X-Scissor";  
 }  
}

**Результат работы:**

Carbink Vasya из команды синих вступает в бой!

Phantump Mia из команды черных вступает в бой!

Carbink Vasya Использует Facade.

Phantump Mia теряет 1 здоровья.

Phantump Mia не замечает воздействие типа NORMAL

Phantump Mia Использует ShadowClaw.

Carbink Vasya теряет 1 здоровья.

Carbink Vasya Использует Swagger.

Phantump Mia не замечает воздействие типа NORMAL

Phantump Mia промахивается

Carbink Vasya Использует Swagger.

Phantump Mia не замечает воздействие типа NORMAL

Phantump Mia использует ShadowBall.

Carbink Vasya теряет 6 здоровья.

Carbink Vasya уменьшает специальную защиту.

Carbink Vasya Использует Swagger.

Phantump Mia не замечает воздействие типа NORMAL

Phantump Mia Использует ShadowClaw.

Carbink Vasya теряет 1 здоровья.

Carbink Vasya Использует Swagger.

Phantump Mia не замечает воздействие типа NORMAL

Phantump Mia использует ShadowBall.

Carbink Vasya теряет 6 здоровья.

Carbink Vasya теряет сознание.

Hydreigon Katya из команды синих вступает в бой!

Hydreigon Katya используе ScaryFace.

Phantump Mia не замечает воздействие типа NORMAL

Phantump Mia Использует ShadowClaw.

Hydreigon Katya использует Bite.

Phantump Mia теряет 17 здоровья.

Phantump Mia теряет сознание.

Trevenant Roma из команды черных вступает в бой!

Hydreigon Katya используе ScaryFace.

Trevenant Roma не замечает воздействие типа NORMAL

Trevenant Roma Использует ShadowClaw.

Hydreigon Katya используе ScaryFace.

Trevenant Roma не замечает воздействие типа NORMAL

Trevenant Roma промахивается

Hydreigon Katya использует DoubleHit.

Trevenant Roma теряет 2 здоровья.

Trevenant Roma не замечает воздействие типа NORMAL

Trevenant Roma Использует X-Scissor.

Hydreigon Katya теряет 12 здоровья.

Hydreigon Katya использует DoubleHit.

Trevenant Roma теряет 2 здоровья.

Trevenant Roma не замечает воздействие типа NORMAL

Trevenant Roma использует ShadowBall.

Hydreigon Katya теряет 4 здоровья.

Hydreigon Katya использует Bite.

Trevenant Roma теряет 14 здоровья.

Trevenant Roma Использует ShadowClaw.

Hydreigon Katya использует DoubleHit.

Trevenant Roma теряет 2 здоровья.

Trevenant Roma не замечает воздействие типа NORMAL

Trevenant Roma теряет сознание.

Zweilous Kirill из команды черных вступает в бой!

Hydreigon Katya использует Bite.

Zweilous Kirill теряет 6 здоровья.

Zweilous Kirill использует DoubleHit.

Hydreigon Katya теряет 6 здоровья.

Hydreigon Katya теряет сознание.

Deino Sasha из команды синих вступает в бой!

Deino Sasha используе ScaryFace.

Zweilous Kirill увеличивает скорость.

Zweilous Kirill использует Bite.

Критический удар!

Deino Sasha теряет 3 здоровья.

Deino Sasha используе ScaryFace.

Zweilous Kirill увеличивает скорость.

Zweilous Kirill используе ScaryFace.

Deino Sasha увеличивает скорость.

Deino Sasha используе ScaryFace.

Zweilous Kirill увеличивает скорость.

Zweilous Kirill используе ScaryFace.

Deino Sasha увеличивает скорость.

Deino Sasha используе ScaryFace.

Zweilous Kirill увеличивает скорость.

Zweilous Kirill использует Bite.

Deino Sasha теряет 2 здоровья.

Deino Sasha использует Bite.

Zweilous Kirill теряет 4 здоровья.

Deino Sasha используе ScaryFace.

Zweilous Kirill увеличивает скорость.

Deino Sasha использует Bite.

Zweilous Kirill теряет 2 здоровья.

Zweilous Kirill использует DoubleHit.

Deino Sasha теряет 10 здоровья.

Deino Sasha теряет сознание.

В команде синих не осталось покемонов.

Команда черных побеждает в этом бою!

**Вывод:**

**В процессе выполнения лабораторной работы я получила навыки использования объектно-ориентированного подхода программирования при использование языка Java.**