Факультет программной инженерии

И

Компьютерных технологий

Лабораторная работа №2

Вариант 491

Преподаватель: Письмак А.Е.

Выполнил: Патутин В.М

Группа:P3114

Санкт-Петербург

2019

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://poke-universe.ru/), [http://pokemondb.net](http://pokemondb.net/),<http://veekun.com/dex/pokemon>

**Комментарии**

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

Что надо сделать (краткое описание)

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);

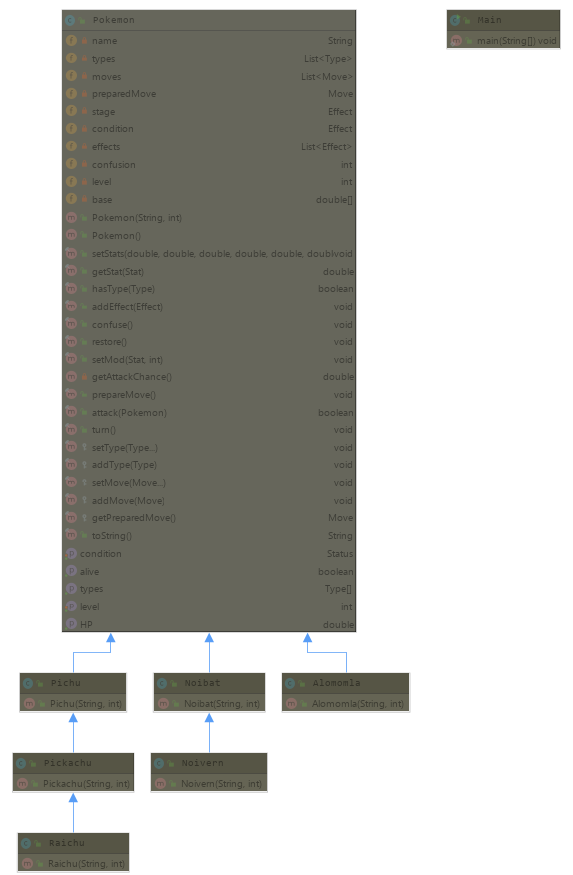
b.addFoe(p2);

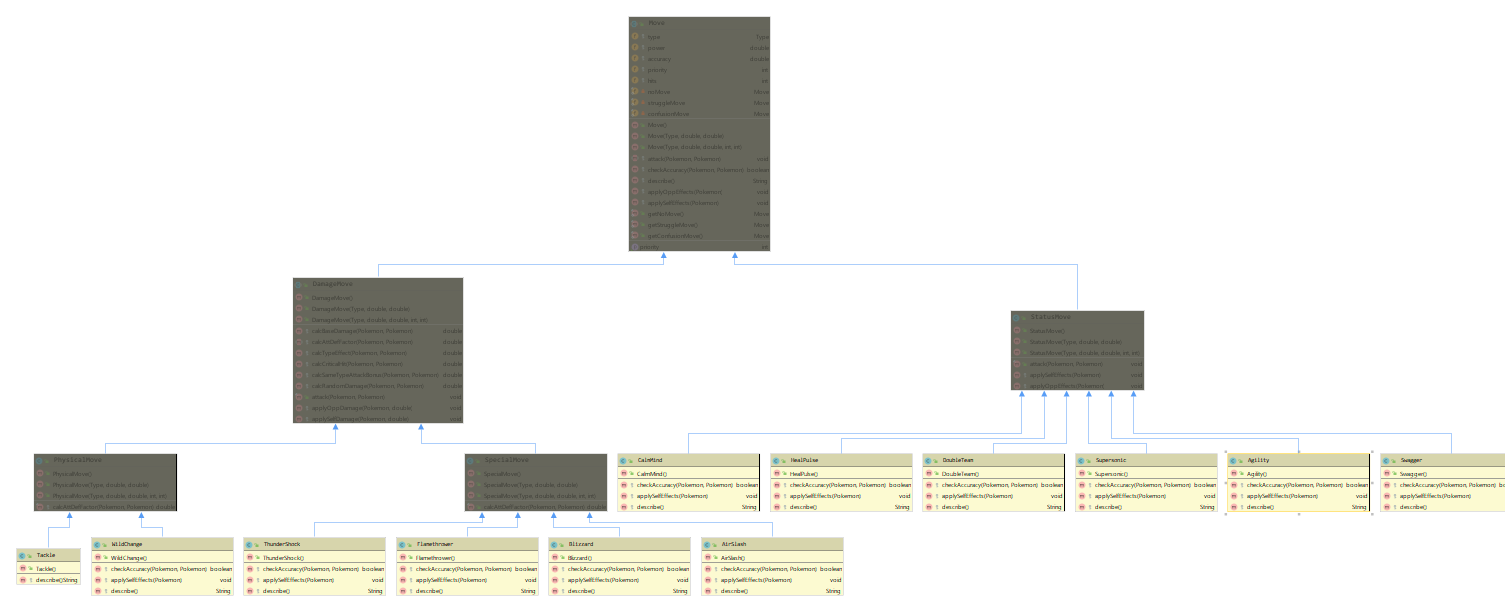
b.go();

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





Исходный код

**Main:**

**package** Laba2;  
**import** Pokemoshi.\*;  
**import** ru.ifmo.se.pokemon.Battle;  
**import** ru.ifmo.se.pokemon.Pokemon;  
  
**public class** Main {  
 **public static void** main (String [] args){  
 Battle b = **new** Battle();  
 *//команда 1* Pokemon player1 = **new** Alomomla(**"Вова"**, 1);  
 Pokemon player2 = **new** Noibat(**"Саня"**, 1);  
 Pokemon player3 = **new** Noivern(**"Настя"**, 1);  
 *//команда 2* Pokemon player4 = **new** Pichu(**"физичка"**, 1);  
 Pokemon player5 = **new** Pickachu(**"математичка "**, 1);  
 Pokemon player6 = **new** Raichu(**"ЧИШ"**, 1);  
  
 b.addAlly(player1);  
 b.addAlly(player2);  
 b.addAlly(player3);  
  
 b.addFoe(player4);  
 b.addFoe(player5);  
 b.addFoe(player6);  
  
  
 b.go();  
  
 }  
}

**Покемоны:**

Alomomla:

**package** Pokemoshi;  
  
**import** Attacke1.Blizzard;  
**import** Attacke1.CalmMind;  
**import** Attacke1.HealPulse;  
**import** Attacke1.Swagger;  
**import** ru.ifmo.se.pokemon.Pokemon;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** Alomomla **extends** Pokemon {  
 **public** Alomomla(String name, **int** level){  
 **super**(name, level);  
  
 **this**.setStats(165,75,80,40,45,65);  
 **this**.setType(Type.***WATER***);  
  
 **this**.setMove(**new** Blizzard(), **new** Swagger(), **new** CalmMind(), **new** HealPulse());  
  
 }  
}

Noibat:

**package** Pokemoshi;  
  
**import** Attacke1.AirSlash;  
**import** Attacke1.Supersonic;  
**import** Attacke1.Tackle;  
**import** ru.ifmo.se.pokemon.Pokemon;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** Noibat **extends** Pokemon {  
 **public** Noibat(String name, **int** level){  
 **super**(name, level);  
  
 **this**.setStats(40,30,35,45,40,55);  
 **this**.setType(Type.***DRAGON***);  
  
 **this**.setMove(**new** Supersonic(), **new** Tackle(), **new** AirSlash());  
  
 }  
}

Noivern:

**package** Pokemoshi;  
  
**import** Attacke1.Flamethrower;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** Noivern **extends** Noibat {  
 **public** Noivern(String name, **int** level){  
 **super**(name, level);  
  
 **this**.setStats(85,70,80,97,80,123);  
 **this**.setType(Type.***DRAGON***);  
  
 **this**.setMove(**new** Flamethrower());  
  
 }  
}

Pichu:

**package** Pokemoshi;  
  
**import** Attacke1.DoubleTeam;  
**import** Attacke1.ThunderShock;  
**import** ru.ifmo.se.pokemon.Pokemon;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** Pichu **extends** Pokemon {  
 **public** Pichu(String name, **int** level){  
 **super**(name, level);  
  
 **this**.setStats(20,40,15,35,35,60);  
 **this**.setType(Type.***ELECTRIC***);  
  
 **this**.setMove(**new** ThunderShock(), **new** DoubleTeam());  
  
 }  
}

Pickachu:

**package** Pokemoshi;  
  
**import** Attacke1.Agility;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** Pickachu **extends** Pichu {  
 **public** Pickachu(String name, **int** level){  
 **super**(name, level);  
  
 **this**.setStats(35,55,40,50,50,90);  
 **this**.setType(Type.***ELECTRIC***);  
  
 **this**.setMove(**new** Agility());  
  
 }  
}

Raichu:

**package** Pokemoshi;  
  
**import** Attacke1.WildChange;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** Raichu **extends** Pickachu {  
 **public** Raichu(String name, **int** level){  
 **super**(name, level);  
  
 **this**.setStats(60,90,55,90,80,110);  
 **this**.setType(Type.***ELECTRIC***);  
  
 **this**.setMove(**new** WildChange());  
  
 }  
}

**Атаки:**

Agility:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.\*;  
  
**public class** Agility **extends** StatusMove {  
 **public** Agility(){  
 **super**(Type.***PSYCHIC***,0,0);  
 }  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon att){  
 Effect effect = **new** Effect();  
 effect.stat(Stat.***SPEED***, +2);  
 att.addEffect(effect);  
 }  
  
 **protected** String describe() {  
 **return "Унизила при классе"**;  
 }  
}

AirSlash:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.Effect;  
**import** ru.ifmo.se.pokemon.Pokemon;  
**import** ru.ifmo.se.pokemon.SpecialMove;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** AirSlash **extends** SpecialMove {  
 **public** AirSlash(){  
 **super**(Type.***FLYING***,75,95);  
 }  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon def){  
 Effect effect = **new** Effect();  
 effect.chance(0.3);  
 effect.*flinch*(def);  
 def.addEffect(effect);  
 }  
  
 **protected** String describe() {  
 **return "Почти выиграл олимпиаду!"**;  
 }  
}

Blizzard:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.SpecialMove;  
**import** ru.ifmo.se.pokemon.Type;  
**import** ru.ifmo.se.pokemon.Pokemon;  
**import** ru.ifmo.se.pokemon.Effect;  
  
**public class** Blizzard **extends** SpecialMove {  
  
 **public** Blizzard (){**super**(Type.***ICE***,110,70);}  
  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon def){  
 Effect effect = **new** Effect();  
 effect.chance(0.1);  
 effect.*freeze*(def);  
 def.addEffect(effect);  
 }  
  
 **protected** String describe()  
 {  
 **return "Лови задачи МГУ, которые я решаю"**;  
 }  
}

CalmMind:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.\*;  
  
**public class** CalmMind **extends** StatusMove {  
 **public** CalmMind(){**super**(Type.***PSYCHIC***,0,0);}  
  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon att){  
 Effect effect = **new** Effect();  
 effect.stat(Stat.***SPECIAL\_ATTACK***, +1);  
 effect.stat(Stat.***SPECIAL\_DEFENSE***, +1);  
 att.addEffect(effect);  
 }  
  
 **protected** String describe()  
 {  
 **return "Иду к Тыну"**;  
 }  
}

DoubleTeam:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.\*;  
  
**public class** DoubleTeam **extends** StatusMove {  
 **public** DoubleTeam(){  
 **super**(Type.***NORMAL***,0,0);  
 }  
  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon att) {  
 Effect effect = **new** Effect();  
 effect.stat(Stat.***SPEED***, +6);  
 att.addEffect(effect);  
 }  
 **protected** String describe() {  
 **return "Боится, потому что пришла директриса"**;  
 }  
}

Flamethrower:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.Effect;  
**import** ru.ifmo.se.pokemon.Pokemon;  
**import** ru.ifmo.se.pokemon.SpecialMove;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** Flamethrower **extends** SpecialMove {  
 **public** Flamethrower(){  
 **super**(Type.***FIRE***,90,100);  
 }  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon def){  
 Effect effect = **new** Effect();  
 effect.chance(0.1);  
 effect.*burn*(def);  
 def.addEffect(effect);  
 }  
  
 **protected** String describe() {  
 **return "Ваши уроки бесполезны, они вообще никак и ничем не помогают, вы просто получаете бабки"**;  
 }  
}

HealPulse:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.\*;  
  
**public class** HealPulse **extends** StatusMove {  
 **public** HealPulse(){**super**(Type.***PSYCHIC***,0,0);}  
  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon att){  
 Effect effect = **new** Effect();  
 effect.stat(Stat.***HP***, +14);  
 att.addEffect(effect);  
 }  
  
 **protected** String describe()  
 {  
 **return "Смотрю сериальчик "**;  
 }  
}

Supersonic:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.\*;  
  
**public class** Supersonic **extends** StatusMove {  
 **public** Supersonic(){  
 **super**(Type.***NORMAL***,0,55);  
 }  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon def){  
 Effect effect = **new** Effect();  
 effect.chance(0.33);  
 **int** q= (**int**)(Math.*random*()\*4.3);  
 effect.turns(q);  
 effect.*confuse*(def);  
 def.addEffect(effect);  
 }  
  
 **protected** String describe() {  
 **return "А вы знали, что можно решить это проще?"**;  
 }  
}

WildChange:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.\*;  
  
**public class** WildChange **extends** PhysicalMove {  
 **public** WildChange(){  
 **super**(Type.***ELECTRIC***,90,100);  
 }  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon def) {  
 Effect effect1 = **new** Effect();  
 effect1.stat(Stat.***HP***, +13);  
 def.addEffect(effect1);  
 }  
  
  
  
 **protected** String describe() {  
 **return ":'Неаттестация по 4 предметам'"** + **" Игрок:'Мне пофиг'"**;  
 }  
}

Tackle:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.PhysicalMove;  
**import** ru.ifmo.se.pokemon.Type;  
  
**public class** Tackle **extends** PhysicalMove {  
 **public** Tackle(){  
 **super**(Type.***NORMAL***,40,100);  
 }  
  
 **protected** String describe() {  
 **return "Закатил слова от банальных слов "**;  
 }  
}

Swagger:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.\*;  
  
**public class** Swagger **extends** StatusMove {  
 **public** Swagger(){  
 **super**(Type.***NORMAL***,0,85);  
 }  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon def){  
 Effect effect = **new** Effect();  
 effect.chance(0.33);  
 effect.*confuse*(def);  
 **int** q= (**int**)(Math.*random*()\*4.3);  
 effect.turns(q);  
 effect.stat(Stat.***ATTACK***, +2);  
 effect.stat(Stat.***DEFENSE***, +2);  
 effect.stat(Stat.***SPECIAL\_ATTACK***, +2);  
 effect.stat(Stat.***SPECIAL\_DEFENSE***, +2);  
 effect.stat(Stat.***SPEED***, +2);  
 def.addEffect(effect);  
 }  
  
 **protected** String describe() {  
 **return "А вы уверены, что ваше решение правильное?"**;  
 }  
}

ThunderShock:

**package** Attacke1;  
  
**import** ru.ifmo.se.pokemon.\*;  
  
**public class** ThunderShock **extends** SpecialMove {  
 **public** ThunderShock(){  
 **super**(Type.***ELECTRIC***,40,100);  
 }  
 **protected boolean** checkAccuracy(Pokemon att, Pokemon def){  
 **return true**;  
 }  
  
 **protected void** applySelfEffects(Pokemon def){  
 Effect effect = **new** Effect();  
 effect.chance(0.1);  
 effect.*paralyze*(def);  
 def.addEffect(effect);  
 }  
  
 **protected** String describe() {  
 **return "Вызвала батю в школу"**;  
 }  
}

**Результат работы программы**

Alomomla Вова из команды красных вступает в бой!

Pichu физичка из команды фиолетовых вступает в бой!

Alomomla Вова Иду к Тыну.

Pichu физичка Боится, потому что пришла директриса.

Pichu физичка Боится, потому что пришла директриса.

Alomomla Вова Иду к Тыну.

Pichu физичка Вызвала батю в школу.

Alomomla Вова теряет 12 здоровья.

Alomomla Вова Иду к Тыну.

Pichu физичка Боится, потому что пришла директриса.

Alomomla Вова Иду к Тыну.

Pichu физичка Вызвала батю в школу.

Alomomla Вова теряет 9 здоровья.

Alomomla Вова теряет сознание.

Noibat Саня из команды красных вступает в бой!

Pichu физичка Боится, потому что пришла директриса.

Noibat Саня Почти выиграл олимпиаду!.

Pichu физичка теряет 3 здоровья.

Pichu физичка Боится, потому что пришла директриса.

Pichu физичка Боится, потому что пришла директриса.

Pichu физичка Боится, потому что пришла директриса.

Pichu физичка Вызвала батю в школу.

Noibat Саня теряет 4 здоровья.

Noibat Саня Почти выиграл олимпиаду!.

Pichu физичка теряет 3 здоровья.

Pichu физичка Вызвала батю в школу.

Noibat Саня теряет 2 здоровья.

Pichu физичка Боится, потому что пришла директриса.

Noibat Саня Почти выиграл олимпиаду!.

Pichu физичка теряет 2 здоровья.

Pichu физичка Боится, потому что пришла директриса.

Pichu физичка Боится, потому что пришла директриса.

Pichu физичка Боится, потому что пришла директриса.

Pichu физичка Вызвала батю в школу.

Noibat Саня теряет 2 здоровья.

Noibat Саня А вы знали, что можно решить это проще?.

Pichu физичка Вызвала батю в школу.

Noibat Саня теряет 4 здоровья.

Noibat Саня теряет сознание.

Noivern Настя из команды красных вступает в бой!

Pichu физичка Вызвала батю в школу.

Noivern Настя теряет 3 здоровья.

Noivern Настя Ваши уроки бесполезны, они вообще никак и ничем не помогают, вы просто получаете бабки.

Pichu физичка теряет 4 здоровья.

Noivern Настя воспламеняется

Pichu физичка теряет сознание.

Pickachu математичка из команды фиолетовых вступает в бой!

Noivern Настя Ваши уроки бесполезны, они вообще никак и ничем не помогают, вы просто получаете бабки.

Pickachu математичка теряет 6 здоровья.

Pickachu математичка Унизила при классе.

Pickachu математичка Унизила при классе.

Noivern Настя Ваши уроки бесполезны, они вообще никак и ничем не помогают, вы просто получаете бабки.

Pickachu математичка теряет 5 здоровья.

Pickachu математичка Унизила при классе.

Noivern Настя Ваши уроки бесполезны, они вообще никак и ничем не помогают, вы просто получаете бабки.

Pickachu математичка теряет 5 здоровья.

Pickachu математичка теряет сознание.

Raichu ЧИШ из команды фиолетовых вступает в бой!

Noivern Настя Ваши уроки бесполезны, они вообще никак и ничем не помогают, вы просто получаете бабки.

Raichu ЧИШ теряет 6 здоровья.

Raichu ЧИШ :'Неаттестация по 4 предметам' Игрок:'Мне пофиг'.

Noivern Настя теряет 4 здоровья.

Raichu ЧИШ теряет 13 здоровья.

В команде фиолетовых не осталось покемонов.

Команда красных побеждает в этом бою!

**Вывод:**

Я научился работать с классами и методами в java и познал основы ООП. И конечно же разобрался с атаками покемонов. Научился наследовать классы.