



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
"МИРЭА - Российский технологический университет"

РТУ МИРЭА

Институт информационных технологий (ИТ)
Кафедра инструментального и прикладного программного обеспечения

ОТЧЁТ
ПО ПРАКТИЧЕСКОЙ РАБОТЕ №2
по дисциплине
«Программирование на языке Java»

**Тема: Использование UML диаграмм в объектно-ориентированном
программировании**

Выполнил студент группы ИКБО-16-20

Пак С.А.

Принял ассистент кафедры ИиППО

Русляков А.А.

Практические работы выполнены «_____» 2021г.

«Зачтено» «_____» 2021г

Москва 2021

СОДЕРЖАНИЕ

ТЕОРЕТИЧЕСКОЕ ВВЕДЕНИЕ.....	3
РЕШЕНИЕ ЗАДАЧИ.....	4
1. Постановка задачи.....	4
2. Программный код.....	4
3. Вывод программы.....	9
ВЫВОД.....	10

ТЕОРЕТИЧЕСКОЕ ВВЕДЕНИЕ

Язык моделирования Unified Modeling Language (UML) является стандартом де-факто с 1998 года для проектирования и документирования объектно-ориентированных программ. Средствами UML в виде диаграмм можно графически изобразить класс и экземпляр класса.

Графически представляем класс в виде прямоугольника, разделенного на три области – область именованного класса, область инкапсуляции данных и область операций (методы).

Имя (или сущность) : определяет класс.

Переменные (или атрибуты, состояние, поля данных класса): содержит статические атрибуты класса, или описывают свойства класса (сущности предметной области).

Методы (или поведение, функции, работа с данными): описывают динамическое поведение класса. Другими словами, класс инкапсулирует статические свойства (данные) и динамические модели поведения (операции, которые работают с данными) в одном месте (“коробке” или прямоугольнике).

РЕШЕНИЕ ЗАДАЧИ

1. Постановка задачи

Задания:

1. По диаграмме класса UML описывающей сущность Автор. Необходимо написать программу, которая состоит из двух классов Author и TestAuthor. Класс Author должен содержать реализацию методов, представленных на диаграмме класса;

2. По UML диаграмме класса написать программу, которая состоит из двух классов. Один из них Ball должен реализовывать сущность мяч, а другой с названием TestBall тестировать работу созданного класса. Класс Ball должен содержать реализацию методов, представленных на UML. Диаграмма на рисунке описывает сущность Мяч написать программу.

Класс Ball моделирует движущийся мяч. В состав класса входят:

- Две переменные с модификатором private (поля данных класса): x, y, которые описывают положение мяча на поле;
- Конструкторы, public методы получения и записи значений для private переменных;
- Метод setXY (), который задает положение мяча и метод setXYSpeed(), чтобы задать скорость мяча;
- Метод move() , позволяет переместить мяч, так что что увеличивает x и y на данном участке на xDisp и yDisp, соответственно;
- Метод toString(), который возвращает "Ball @ (x , y) ".

2. Программный код

Файл Author.java:

```
package ru.mirea.classes;

public class Author {
    private String name;
    private String email;
    private char gender;

    /**
     * Конструктор, создающий полноценный объект класса Author
     * @param name        имя автора
     * @param email       email автора
     * @param gender      пол автора
     */
    public Author(String name, String email, char gender) {
        this.name = name;
        this.email = email;
    }
}
```

```

        this.gender = gender;
    }

    /**
     * Геттер для поля name
     * @return      имя автора
     */
    public String getName() {
        return this.name;
    }

    /**
     * Геттер для поля email
     * @return      email автора
     */
    public String getEmail() {
        return this.email;
    }

    /**
     * Сеттер для поля email
     * @param email      новый email автора
     */
    public void setEmail(String email) {
        this.email = email;
    }

    /**
     * Геттер для поля gender
     * @return      пол автора
     */
    public char getGender() {
        return this.gender;
    }

    /**
     * Объединяет информацию об объекте в одну строку
     * @return      строка с информацией об объекте
     */
    public String toString() {
        return this.name + " (" + this.gender + ") at " + this.email;
    }
}

```

Файл AuthorTest.java:

```
package ru.mirea;
```

```

import java.util.Scanner;
import ru.mirea.classes.Author;

public class AuthorTest {
    private static final Scanner IN = new Scanner(System.in);

    public static void main(String[] args) {
        System.out.print("Введите имя автора: ");
        String name = IN.nextLine();

        System.out.print("Введите email автора: ");
        String email = IN.next();

        System.out.print("Введите пол автора: ");
        char gender = IN.next().toCharArray()[0];

        Author author = new Author(name, email, gender);

        System.out.println();
        System.out.println("Информация об авторе:");
        System.out.println(author);
    }
}

```

Файл Ball.java:

```

package ru.mirea.classes;

public class Ball {
    private double x;
    private double y;

    /**
     * Конструктор по умолчанию
     */
    public Ball() {
        this.x = 0.0;
        this.y = 0.0;
    }

    /**
     * Конструктор, создающий полноценный объект класса Ball
     * @param x координата мяча по оси абсцисс
     * @param y координата мяча по оси ординат
     */
}

```

```

public Ball(double x, double y) {
    this.x = x;
    this.y = y;
}

/**
 * Геттер для поля x
 * @return      координата мяча по оси абсцисс
 */
public double getX() {
    return this.x;
}

/**
 * Сеттер для поля x
 * @param x      новая координата по оси абсцисс
 */
public void setX(double x) {
    this.x = x;
}

/**
 * Геттер для поля y
 * @return      координата мяча по оси ординат
 */
public double getY() {
    return this.y;
}

/**
 * Сеттер для поля y
 * @param y      новая координата мяча по оси ординат
 */
public void setY(double y) {
    this.y = y;
}

/**
 * Задаёт положение мяча на координатной плоскости
 * @param x      координата по оси абсцисс
 * @param y      координата по оси ординат
 */
public void setXY(double x, double y) {
    this.x = x;
    this.y = y;
}

```

```

/**
 * Задаёт движение мяча на координатной плоскости
 * @param xDisp      приращение координаты по оси абсцисс
 * @param yDisp      приращение координаты по оси ординат
 */
public void move(double xDisp, double yDisp) {
    this.x += xDisp;
    this.y += yDisp;
}

/**
 * Объединяет информацию об объекте в одну строку
 * @return          строка с информацией об объекте
 */
public String toString() {
    return "\tКоордината по оси X: " + this.x
        + "\n\tКоордината по оси Y: " + this.y;
}
}

```

Файл BallTest.java:

```

package ru.mirea;

import java.util.Scanner;
import ru.mirea.classes.Ball;

public class BallTest {
    private static final Scanner IN = new Scanner(System.in);

    public static void main(String[] args) {
        System.out.print("Введите координаты мяча: ");

        double x = IN.nextDouble();
        double y = IN.nextDouble();

        Ball ball = new Ball();
        ball.setXY(x, y);

        System.out.println("Мяч сдвинулся на 4.5 единицы по оси X,
3.23 единицы по оси Y");
        ball.move(4.5, 3.23);

        System.out.println();
        System.out.println("Теперь его координаты таковы:");
        System.out.println(ball);
    }
}

```


}

3. Вывод программы

Во всех программах пользователь поочерёдно вводит значения для полей объектов, и в конце выводится вся информация о них (рис.1, 2).

```
7 ezhik@spark:production/Java-MIREA-prac-2 $ java ru.mirea.AuthorTest
6 Введите имя автора: Lewis Carroll
5 Введите email автора: lewis@mail.com
4 Введите пол автора: m
3
2 Информация об авторе:
1 Lewis Carroll (m) at lewis@mail.com
8 ezhik@spark:production/Java-MIREA-prac-2 $
```

```
public static void main(String[] args) {
    System.out.print("Введите координаты мяча: ");
    double x = IN.nextDouble();
    double y = IN.nextDouble();
}
```

Рис.1 AuthorTest.java

```
7 ezhik@spark:production/Java-MIREA-prac-2 $ java ru.mirea.BallTest
6 Введите координаты мяча: 4 5
5 Мяч сдвинулся на 4.5 единицы по оси X, 3.23 единицы по оси Y
4
3 Теперь его координаты таковы:
2 Координата по оси X: 8.5
1 Координата по оси Y: 8.23
8 ezhik@spark:production/Java-MIREA-prac-2 $
```

```
public static void main(String[] args) {
    System.out.print("Введите координаты мяча: ");
    double x = IN.nextDouble();
    double y = IN.nextDouble();
}
```

Рис.2 BallTest.java

ВЫВОД

В ходе выполнения работы изучил UML-диаграммы классов.