|  |  |
| --- | --- |
|  |  |
| МИНОБРНАУКИ РОССИИ |  |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |  |

**Институт информационных технологий**

КАФЕДРА ИНСТРУМЕТНАЛЬНОГО И ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ (ИППО)

ПРАКТИЧЕСКАЯ РАБОТА № 2

по дисциплине «Объектно-ориентированное программирование»

«Проектирование классов: UML-диаграммы»

ВАРИАНТ 1

Выполнил студент группы ИНБО-01-18 *Трофимов С. М.*

Принял Батанов А.О.

Практическая работа выполнена «\_\_\_»\_\_\_\_\_\_\_2019г.

«Зачтено» «\_\_\_»\_\_\_\_\_\_\_2019г.

Москва 2019

**Цель работы**

Работа с UML-диаграммами классов.

**Теоретическое введение**

Графически представляем класс как трехкомпонентный “контейнер-бокс” с отсеками для инкапсуляции данных и операций (методами составляющими интерфейсную часть класса)

Имя (или сущность ) : определяет класс.

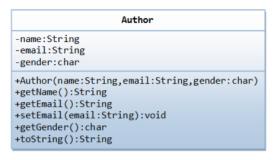
Переменные (или атрибуты, состояние, поля данных класса): содержит статические атрибуты класса, или описывают свойства класса (сущности предметной области).

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

**Выполнение практической работы**

*Задание:*

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



*Решение:*

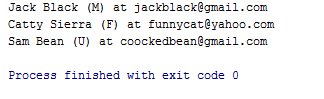
Файл Author.java

**package** com.gradess.company;  
  
**public class** Author {  
  
 **private** String **name**;  
 **private** String **email**;  
  
 **private char gender**;  
  
 **public** Author(String name, String email, **char** gender) {  
 **this**.**name** = name;  
 **this**.**email** = email;  
  
 **if** (gender != **'M'** && gender != **'F'**) {  
 **this**.**gender** = **'U'**;  
 } **else** {  
 **this**.**gender** = gender;  
 }  
 }  
  
 **public** String getName() {  
 **return name**;  
 }  
  
 **public** String getEmail() {  
 **return email**;  
 }  
  
 **public void** setEmail(String email) {  
 **this**.**email** = email;  
 }  
  
 **public char** getGender() {  
 **return gender**;  
 }  
  
 @Override  
 **public** String toString() {  
 **return name** + **" ("** + **gender** + **") at "** + **email**;  
 }  
}

Файл TestAuthor.java

**package** com.gradess.company;  
  
**public class** TestAuthor {  
  
 **public static void** main(String[] args) {  
 Author maleAuthor = **new** Author(**"Jack Black"**, **"jackblack@gmail.com"**, **'M'**);  
 Author femaleAuthor = **new** Author(**"Catty Sierra"**, **"funnycat@yahoo.com"**, **'F'**);  
 Author undefinedAuthor = **new** Author(**"Sam Bean"**, **"coockedbean@gmail.com"**, **'U'**);  
  
 System.***out***.println(maleAuthor);  
 System.***out***.println(femaleAuthor);  
 System.***out***.println(undefinedAuthor);  
 }  
}

**Результат выполнения программы**



**Выводы по работе**

Я научился работать с UML-диаграммами классов.

**Используемая литература:**

1. К. Сьерра, Б. Бэйтс. Изучаем Java, 2-е издание.
2. С. Макконнелл. Совершенный код, 3-е издание.