## Учреждение образования

## Белорусский государственный технологический университет

Кафедра полиграфического оборудования и

системы обработки информации

**ЛАБОРАТОРНАЯ РАБОТА 1**

по дисциплине «Стандартизация и сертификация

информационных систем и технологий»

Тема

Изучение соглашения по оформлению кода - Java Code Conventions

Выполнил студент:

Коржова В. С.

4 курс 4 группа

Проверил

кандидат технических наук

Сулим П.Е.

Минск 2024

***Цель* *работы*:** Написать Java-code по правилам Java Code Conventions.

***Описание работы:*** Необходимо ознакомится с Java Code Conventions, привести примеры Java-кода с пояснениями по оформлению.

**1. Обеспечение доступа к экземпляру и переменным класса**

Не следует делать какие-либо переменные экземпляра или класса об-щедоступными без уважительной причины. Часто встречаются случаи, когда поля класса не должны устанавливаться или считываться напрямую – их чтение и запись должны происходить лишь при вызове методов.

Одним из примеров уместного использования public полей может быть случай, когда класс описывает лишь структуру данных, без какого-либо поведения, тогда можно сделать переменные экземпляра класса public.

**2. Константы**

Численные константы (литералы) не должны кодироваться напрямую, за исключением -1, 0 и 1, которые могут использоваться в циклах для управления счетчиком.

**3. Примеры присваивания значение переменным**

Избегайте присваивания значения некоторым переменным в одном вы-ражении. Это усложняет чтение.

|  |
| --- |
| fooBar.fChar = barFoo.lchar = 'c'; // AVOID! |

Не используйте оператор присваивания в местах, где он может быть легко спутан с оператором сравнения.

|  |
| --- |
| If (c++ = d++) { // AVOID!  …  }  if ((c++ = d++) != 0) { //OK  ...  } |

Не используйте вложенные присваивания, пытаясь ускорить скорость выполнения программы. Это работа компилятора, и, кроме того, на самом деле редко помогает.

|  |
| --- |
| d = (a = b + c) + r; // AVOID!  a = b + c; //OK  d = a + r; //OK |

**4. Круглые скобки**

Как правило, хорошей практикой является использование круглой скобки в выражениях, содержащих различные операторы, чтобы избегать проблем с приоритетом операторов. Даже если приоритет оператора вам ка-жется очевидным, то для других это может быть не так - вы не должны пред-полагать, что другие программисты знают приоритет, как и вы.

|  |
| --- |
| if (a == b && c == d) // AVOID!  if ((a == b) && (c == d)) //OK |

**5. Примеры кода**

В листинге 1 представлен класс Box.

|  |
| --- |
| /\*  \* Box  \* 1.10  \* Copyright notice  \*/ package by.anna;  import java.time.LocalDate;  */\*\*  \* Class description goes here.  \*  \** ***@author*** *Anna Kostyukova  \** ***@version*** *1.10 01 Feb 2022  \*/* public class Box {  private int width, height;   public Box(int width, int height) {  this.width = width;  this.height = height;  }   public int getWidth() {  return width;  }   public void setWidth(int width) {  this.width = width;  }   public int getHeight() {  return height;  }   public void setHeight(int height) {  this.height = height;  }   @Override  public String toString() {  return "Box{" +  "width=" + width +  ", height=" + height +  ", date=" + LocalDate.*now*() +  '}';  } } |

Листинг 1 – Класс Box

В листинге 2 представлен класс Main.

|  |
| --- |
| package by.anna;  public class Main {  public static void main(String[] args) {  final int WIDTH = 4;  final int HEIGHT = 5;    Box myBox = new Box(WIDTH, HEIGHT);  System.out.println(myBox);  }  } |

Листинг 2 – Класс Main

**Выводы:** В ходе выполнения данной лабораторной работы были изучены ключевые принципы написания кода на Java, а также были созданы кодовые примеры, соответствующие всем правилам, представленным в Java Code Conventions.