Институт информационных технологий и управления

в технических системах

Кафедра информационных технологий и компьютерных систем

ОТЧЕТ

по лабораторной работе № 2

«**ОБРАБОТКА ДАННЫХ ПРОСТЫХ ТИПОВ. РАБОТА**

**С ПАНЕЛЬЮ КОДА BLUEJ. ФОРМАТИРОВАННЫЙ ВЫВОД**»

по дисциплине «Программирование. Базовые процедуры обработки информации»

Выполнил студент группы ИВТ/б-21-2-о

Заводов М.А.

Проверила доцент Владимирова Е.С.

Севастополь

2020

**1. ЦЕЛЬ РАБОТЫ**

Ознакомиться с простыми типами данных Java, научиться объявлять переменные и литералы этих типов и выполнять операции над ними, научиться применять оператор присваивания для данных простых типов, научиться применять метод System.out.printf() – метод форматированного вывода − для вывода на экран значений различных типов, научиться применять окно кода в BlueJ.

2. **ПОСТАНОВКА ЗАДАЧИ**

1) Ввести заданные операторы в окно кода BlueJ и проанализировать полученные результаты.

2) Разработать программу, в которой используется метод System.out.printf() для вывода в окно терминала данных, предусмотренных вариантом задания.

Данные варианта 1 приведены в таблице 2.1.

Таблица 2.1 – Вариант задания

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Но-мер варианта | Строка str (String) | Целое  a (int) | Целое  b (short) | Целое  с (byte) | Действи-тельное  d (double) | Действи-тельное  f (float) |
| 8 | Фамилия\_Группа: | С8 | С10,Ш9,НН | С16 | Ш10, Т4,  ОН, | Ш8, Т2,  НН,ВЗ |

**3. ВЫПОЛНЕНИЕ ЗАДАНИЯ 1 В ОКНЕ КОДА**

1. Ввел целые константы в окно кода. Выяснил, что число стандартно имеет тип инт. См. рис 3.1
2. Ввел выражения из пункта 2, выяснил, что результат операций сложения, умножения, деления и получения остатка от деления является целое число (Если константы являются целыми числами) См. рис 3.1
3. Ввел выражения из пункта 3 и выяснил, что результатом операций равенства, сравнения и неравенства является логический тип Boolean См. рис 3.1
4. Ввел дробные числа. Выяснил, что изначально дробное число имеет тип double. Если добавить к концу числа ‘f’, то число будет типа float. См. рис 3.1
5. Ввел операции над дробными числами. Выяснил, что результатом операций сложения, умножения, деления и получения остатка от деления является дробное число, если выражение имеет хотя бы одну дробную переменную. Стандартный тип будет double, но если числа будут типа данных float, то в результате будет число типа float, а не double. См. рис 3.1
6. Ввел логические выражения над дробными числами. Выяснил, что целое число равно дробному числу, если дробное число представляет собой целое число с дробной частью, которая равна 0 (1 == 1.0 (true)). Выяснил, что double == float всегда выводит false См. рис 3.2
7. Ввел в панель кода некоторые части кода. Выяснил, что при объявлении переменной без значени, ей подставляется автоматическое значение равное 0.0. При присвоении типу float дробного числа, оно обязательно должно быть с указанием типа float. См. рис 3.2
8. Присвоил переменной из пункта 7 значение 3.5. Объявил переменные типа float и double. Выяснил, что возможно типу double присвоить значения типа float. Это возможно, потому что тип double более точный чем float, а значит потери данных не будет. См. рис 3.3
9. Ввел операции над переменными и получил результат
10. Ввел константы разных типов и попробовал присвоить им значение, в результате получил ошибку. Константы можно использовать только для чтения. См. рис 3.3

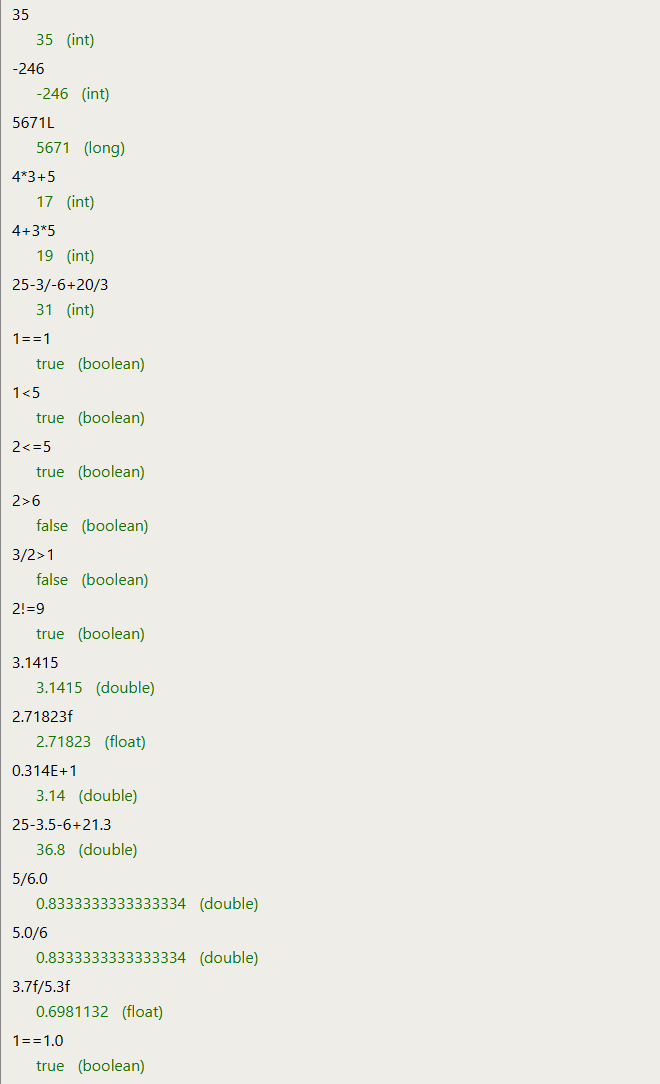


Рисунок 3.1 – Результат работы в панели кода 1

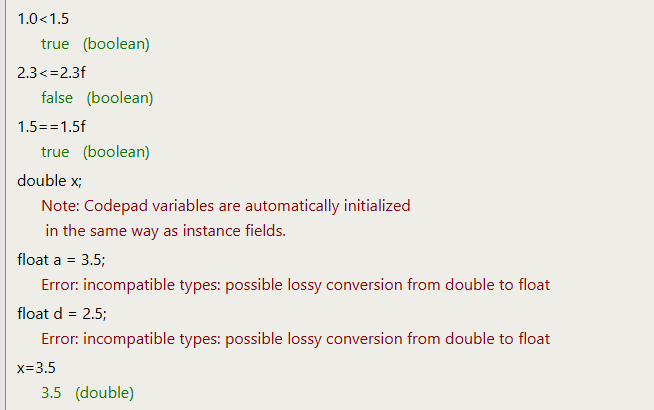


Рисунок 3.2 – Результат работы в панели кода 2



Рисунок 3.3 – Результат работы в панели кода 3

**4. ВЫПОЛНЕНИЕ ЗАДАНИЯ 2**

**4.1. Анализ задачи**

Для выполнения поставленной задачи нужно определить в программе переменные заданных типов (таблица 2.1). Для вывода значений переменных в окно терминала можно использовать вызов метода форматированного вывода System.out.printf(). Форматы вывода будут заданы первым аргументом метода (символьная строка), а имена выводимых в этих форматах переменных будут перечислены далее в списке аргументов через запятую. Число переменных в списке аргументов должно быть равно числу форматов, а типы переменных должны соответствовать типу форматов. После вывода значений переменных в заданном виде нужно обеспечить переход на следующую строку.

**4.2. Тестовый пример и результаты его обработки вручную**

Для проверки работы заданной программы, зададим случайные значения переменных.

**4.3. Текст программы**

**import Labs.Lab;  
import Labs.Lab1;  
import Labs.Lab2;  
  
public class Program {  
 public static void main(String[] args) {  
 Lab lab = new Lab2(); *//инициализация Лабы интерфейса Лаб (выполнил* ...)  
 System.out.println(lab.getIntro()); *// Вывод интро*  
 lab.runLab(); *//Запуск лабы*  
 System.out.println(lab.getOutro()); *//Вывод аутро (проверил ...)*  
 }  
}**

**package Labs;  
import java.util.Date; *//Импорт класса Date*  
public abstract class Lab {  
 String Intro = "Программу выполнил:"; *//Определение основных полей класса.* String Name = "Заводов Максим Александрович"; *//Выписано сюда для удобства изменения* String Code = "Шифр группы: ИВТ/б-21-2-о";  
 String Variant = "Вариант: 7";  
 String Date = "Дата: " + new Date(); //Создание объекта Date, который возвращает текущую дату и перевод в строку  
 String Outro = "Проверила: \nВладимирова Елена Сергеевна";  
  
 public void runLab() {  
 System.out.println("Лаба не выбрана");  
 }  
  
 public String getIntro() { *//Метод вывода интро. Возвращает забитые поля из родительского интерфейса* return Intro + "\n" + Name + "\n" + Code + "\n" + Date + "\n" + Variant;  
 }  
  
 public String getOutro(){ *//Метод вывода аутро. Возвращает забитое поле из родительского интерфейса* return Outro;  
 }  
}**

**package Labs;  
public class Lab2 extends Lab {  
 @Override  
 public void runLab(){  
 String \_string = "Заводов\_ИВТ/б-21-2-о";  
 int \_int = (int)(Math.random()\*1000000000);  
 short \_short = (short)(Math.random()\*10000);  
 byte \_byte = (byte)(Math.random()\*10);  
 double \_double = Math.random()\*100;  
 float \_float = (float)Math.random()\*1000;  
   
 System.out.printf("string = %s , int = %o , short = %09d , byte = %x , double = %10.4f , float %+08.2f \n", \_string, \_int, \_short, \_byte, \_double, \_float); *//Вывод форматированных переменных*  
 }  
}**

**4.4. Сведения об отладке программы и проверке ее работоспособности**

Результат работы программы на тестовом примере, приведенном в разделе 4.2, изображен на рисунке 4.1.

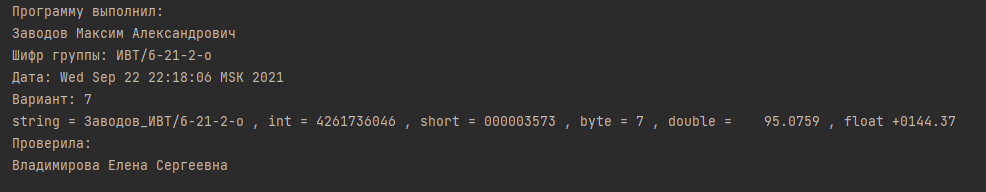
****

Рисунок 4.1 – Результат работы программы

На тестовом наборе входных данных программа выдала ожидаемый результат, что позволяет сделать вывод о ее работоспособности.

**ВЫВОД**

В ходе выполнения лабораторной работы были проведены исследования простых типов данных языка Java в окне кода BlueJ, получены навыки в объявлении переменных и литералов этих типов, выполнении операций над ними, применении оператора присваивания для данных простых типов. Разработана и протестирована программа, использующая System.out.printf() – метод форматированного вывода − для вывода на экран значений различных типов в заданных форматах.