Министерство образования Республики Беларусь Белорусский Национальный Технический Университет

Факультет информационных технологий и робототехники

Кафедра «Программное обеспечение вычислительной техники и автоматизированных систем»

# Отчёт

по лабораторной работе №01

по дисциплине ***«Объектно-ориентированное программирование»***

тема: «Базовый синтаксис языка Java.

Примитивные типы данных в Java.

Переменные и литералы»

Название бригады: «**Best progger**»

Исполнитель: студенты группы 10701118

Воробей Иван Александрович

Преподаватель: Иванченко Виктор Викторович

2018-2019 учебный год

## ЛАБОРАТОРНАЯ РАБОТА №03

**Базовый синтаксис языка Java.**

**Примитивные типы данных в Java**.

Переменные и литералы

**Цель работы**

Ознакомиться с базовым синтаксисом языка Java; приобрести навыки объяв-

ления, инициализации и использования переменных и литерал при програм-

мировании вычислительных алгоритмов; закрепить всё вышеописанное на

примере разработки простейших Java-приложений.

**Требования**

1) При разработке кода можно использовать любую интегрированную среду

разработки. Однако, при запуске программы рекомендуется вручную задей-

ствовать основные компоненты Java (компилятор – javac, утилиту для за-

пуска JVM – java).

2) При написании кода считать, что пользователь вводит всегда корректные дан-

ные.

3) Если логически не подразумевается или в задании иного не указано, то вход-

ными и выходными данными являются вещественные числа (числа с плаваю-

щей запятой).

4) При разработке программ придерживайтесь соглашений по написанию кода

на JAVA (Java Code-Convention).

**Общее задание**

Для закрепления написания простейших программ с использованием языка

программирования Java попробуйте создать простенькие программы-кон-

верторы для различных шкал температур (из градусов Цельсия в градусы Фа-

ренгейта или Кельвина и наоборот) или для различных валют (к примеру, из

бел. руб. в евро или наоборот). Можно использовать любую предметную об-

ласть для создания однотипных приложений (к примеру, конвертор значений

углов из градусы в радианы и наоборот).

**Основное задание**

1) Масса динозавра задаётся в граммах. Разработайте программу, которая вы-

числяет, сколько это килограммов, центнеров и т.д.

2) Дан общий размер файла в байтах (размер задаётся в виде целого числа). Раз-

работайте программу, которая вычисляет, сколько это килобайтов, мегабайтов

и т.д.

3) Значение расстояния между двумя городами задаётся в сантиметрах. Разра-

ботайте программу, которая вычисляет, сколько это километров и метров.

4) Попробуйте разработать программу, которая меняет местами содержимое

двух переменных a и b, не используя для этого дополнительные переменные

(это самый первый и простой алгоритм в программировании).

5) Разработать программу вычисления того, сколько стоит один кг шоколадных

конфет и 1 один кг желатинок, а также во сколько раз шоколадные конфеты

дороже (дешевле) желатинок, если известно, что X кг шоколадных конфет

стоит A бел. руб., а Y кг желатинок стоит B бел. руб.

**Результаты выполнения основного задания:**

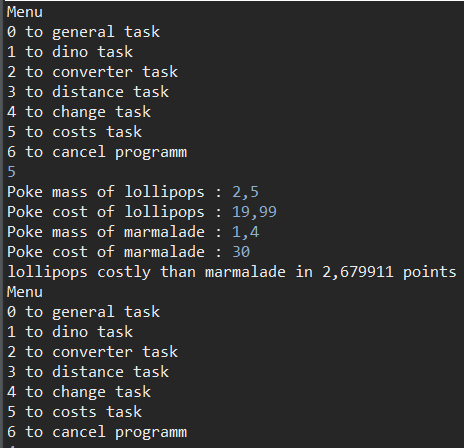


Рисунок 1 – Результат компиляции и запуска программы на языке программирования Java

**Что мы узнали нового в процессе выполнения лабораторной работы:**

1. Какие есть примитивные типы данных в java. Узнал об обертках типов как ими пользываться и зачем они нужны.
2. Узнал каке есть типы операций, над какими типами они возможны, над какими нет.
3. Узнал о приведении типов в языке java.

*ПРИЛОЖЕНИЕ А*

#### Листинг исходного кода Lab02 на языке программирования Java

package therd\_lab;

import java.util.Scanner;

public class therd {

static void menu() {

int choice;

Scanner in = new Scanner(System.***in***);

lable: while (true) {

System.***out***.printf("Menu\n0 to general task\n1 to dino task\n" + "2 to converter task\n"

+ "3 to distance task\n4 to change task\n" + "5 to costs task\n6 to cancel programm\n");

choice = in.nextInt();

switch (choice) {

case 0:

*task0*();

break;

case 1:

*task1*();

break;

case 2:

*task2*();

break;

case 3:

*task3*();

break;

case 4:

*task4*();

break;

case 5:

*task5*();

break;

case 6:

break lable;

}

}

}

static void task0() {

double radian, degree;

degree = 78.32;

radian = degree \* Math.***PI*** / 180;

System.***out***.printf("%f degrees = %f radians\n", degree, radian);

radian = 2.65;

degree = radian \* 180 / Math.***PI***;

System.***out***.printf("%f radians = %f degrees\n", radian, degree);

}

static void task1() {

double grams;

Scanner in = new Scanner(System.***in***);

System.***out***.printf("Enter mass in grams : ");

grams = in.nextDouble();

System.***out***.printf("%f gramms = %f kilos\n%f grams = %f kintals\n%f grams = %f ton\n", grams, grams / 1000,

grams, grams / 10000, grams, grams / 1000000);

}

static void task2() {

int amount\_byte;

Scanner in = new Scanner(System.***in***);

System.***out***.printf("Enter amount of bytes : ");

amount\_byte = in.nextInt();

System.***out***.printf("%d bytes = %d Kb\n%d bytes = %d Mb\n%d bytes = %f Gb\n%d bytes = %f Tb\n", amount\_byte,

amount\_byte / 1000, amount\_byte, amount\_byte / 1000000, amount\_byte,

amount\_byte / Math.*pow*(10, 9), amount\_byte, amount\_byte / Math.*pow*(10, 12));

}

static void task3() {

double centimetrs;

Scanner in = new Scanner(System.***in***);

System.***out***.printf("Enter distance in cetimetrs : ");

centimetrs = in.nextDouble();

System.***out***.printf("%f cantimetrs = %f metrs\n%f centimetrs = %f Km\n", centimetrs, centimetrs / 100, centimetrs,

centimetrs / 100000);

}

static void task4() {

double a = 8, b = 3;

System.***out***.printf("a = %f\nb = %f\n", a, b);

a = a + b;

b = a - b;

a = a - b;

System.***out***.printf("After changeing:\na = %f\nb = %f\n", a, b);

}

static void task5() {

double x, a, y, b, lollipops\_cost, marmalade\_cost, attitude;

Scanner in = new Scanner(System.***in***);

System.***out***.printf("Poke mass of lollipops : ");

x = in.nextDouble();

System.***out***.printf("Poke cost of lollipops : ");

a = in.nextDouble();

System.***out***.printf("Poke mass of marmalade : ");

y = in.nextDouble();

System.***out***.printf("Poke cost of marmalade : ");

b = in.nextDouble();

lollipops\_cost = x / a;

marmalade\_cost = y / b;

if (lollipops\_cost > marmalade\_cost) {

System.***out***.printf("lollipops costly than marmalade in %f points\n", lollipops\_cost / marmalade\_cost);

} else {

System.***out***.printf("lollipops tatty than marmalade in %f points\n", marmalade\_cost / lollipops\_cost);

}

}

public static void main(String[] args) {

*menu*();

}

} }

}

#### 

**Контрольные вопросы**

1. Какие пользовательские типы данных можно использовать в Java?

Class,Intrface,Enum.

2. Сколько классов и с какими модификаторами доступа можно описать в одном

исходном файле в Java?

Any namber fo classes, but only one can be public.

3. Как связано имя исходного файла с именами классов, которые в нём описаны?

The file should contain class with the same name.

4. Что необходимо добавить в класс, чтобы он был запускаемым?

Public static void main(String[] args){}

5. Каков синтаксис стартового метода main(...)?

Public static void main(String[] args){}

6. Что такое литерал (literal) и зачем он нужен?

запись в [исходном коде](https://ru.wikipedia.org/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4) [компьютерной программы](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0), представляющая собой фиксированное значение; Литералами также называют представление значения некоторого типа данных; Литералы представляют собой константы, включаемые непосредственно в текст программы. В отличие от прочих элементов языка ([констант](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%81%D1%82%D0%B0%D0%BD%D1%82%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), [переменных](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D0%B0%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)))[[1]](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%82%D0%B5%D1%80%D0%B0%D0%BB_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)#cite_note-pcmag.ru.encyclopedia.23741-1), литералы не могут быть изменены в тексте программы.

7. Что такое переменная (variable) и зачем она нужна?

a **variable** is a symbol or letter, such as "x" or "y," that represents a value. The data type, if not defined explicitly, is determined based on the initial value given to the **variable**.

8. Что такое время жизни и область действия переменной?

**Life Time** - Life time of any variable is the time for which the particular variable outlives in memory during running of the program.

**Scope** - The scope of any variable is actually a subset of life time. A variable may be in the memory but may not be accessible though. So, the area of our program where we can actually access our entity (variable in this case) is the scope of that variable.

9. Что такое локальная переменная? Что такое глобальная переменная?

Область видимости локальной переменной ограничена методом или циклом, а глобальная переменная видна везде и объявляется в глобальной области.

10. К какому типу языку относиться Java с точки зрения типизации?

Static, strong typing

11. Как в Java объявить локальную переменную?

You can declare a local variable in method, or in loop declaration. Generally in scope, which you need.

12. Что гласит правило объявления идентификаторов в Java?

Должны начинаться с буквы, подчеркивания или знака денежной еденицы. Не могут содержать собаку.

13. Что нужно помнить при выборе имени переменной или имени класса?

Класс начинается с большой буквы и является существительным. Переменные начинаются с маленькой буквы.

14. Какие системы счисления поддерживаются в Java при работе с числовыми

данными?

Decimal, binary, hexadecimal, octal.

15. Как JMV работает с вещественными данными?

Мантиса, экспонента и разделительная еденица.

16. Опишите типы литералов (целочисленные, вещественные, булевские, символьные и строковые), доступных в языке Java. Какими способами каждый из нихможно отобразить?

Целочисленные литтералы по умолчанию имеют тип инт, вещественные по умолчанию тип даубл. Дитерады с “L” на конце типа long , литераллы с “f” на конце типа флоат. Литералы могут быть представлены в разных системах счисления. Булевские литералы это true and false. Символьные могут быть представлени номером в таблице, либо в одинарных скобочком аски кодом либо сам символ в одинарных скобках. Строковые литералы записываются в quotation marks.

17. Какой тип по умолчанию имеют целочисленные и вещественные литералы?

Int, double.

18. Зачем нужны спецификаторы типов и сколько их?

Что бы объявить переменную необходимого типа.

19. Какое новшество было добавлено с JDK 7.0 для удобства чтения чисел? Как

правильно его нужно использовать?

20. Зачем в Java существуют Wrapper-классы (классы-обвёртки над соответствую-

щими примитивными типами данных)? Опишите соответствия между прими-

тивными типами данных и классами обвёртками.

Wrapper-classes expend opportunities for primitive types. It just simple classes. When we create a object for example Integer, we get object with it’s special methods. But primitives take up less space. Byte,Short,Integer,Long,Float,Double,Character,Boolean.