# Лабораторна робота 1. Знайомство з середовищем Eclipce та мовою програмування ‘java’

## Мета роботи

В процесі виконання лабораторної роботи студенти мають отримати такі знання та навички:

* Отримати первинні навички роботи з середовищем Eclipse.
* Отримати первинні уявлення про клас та об’єкт.
* Ознайомитися зі структурою runtime бібліотеки Java.
* Ознайомитися з деякими класами мови Java.
* Отримати уявлення про програму на мові Java.
* Створити простіші програми, використовуючи класи Java.

## Завдання на лабораторну роботу

Під час виконання лабораторній роботі виконати такі завдання:

* Створення і виконання програми «Hello world».
* Дослідити класи як: System, PrintStream, StringBuilder, Scanner, String, Integer, Math, Arrays.

## Результати виконання роботи

### Результати створення і виконання програми «Hello World!».

У лістингу 1.1 представлено код програми.

* 1. Код програми

package HelloWorld;

public class HelloWorld {

public static void main(String[] args) {

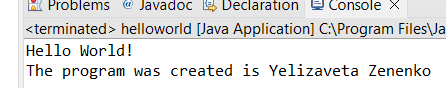
System.*out*.println("Hello World!");

System.*out*.println("The program was created is Yelizaveta Zenenko");

}

}

На рисунку 1.1 наведено результат виконання програми.



* 1. Результат виконання програми

### Дослідження класів System та PrintStream

У класі PrintStream є метод printf() і в цьому завданні нам потрібно дослідити, чи можливо виводити данні на консоль так як у С за допомогою цього методу. Для порівняння створимо дві програми, одну мовою С другу мовою Java.

* 1. Реалізація на мові С

#include <QtCore/QCoreApplication>

#include <stdio.h>

int main(int argc, char \*argv[])

{

QCoreApplication a(argc, argv);

printf("Hello %s!", "World");

int num = 10;

printf("\nMy number %d%", num);

return a.exec();

}

### 

* 1. Результат виконання
  2. Реалізація на мові Java

package print\_Stream;

public class Print\_Stream {

public static void main(String[] args) {

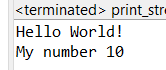
System.*out*.printf("Hello %s!", "World");

int num = 10;

System.*out*.printf("\nMy number %d", num);

}

}



* 1. Результат виконання

Після проведеного порівняння можна зробити висновок, що метод printf() у мові Java використовується так само як і у мові С. Для дослідження класів представимо у приклад такі методи, як print(), println(), printf().

* 1. Дослідження методів print(),println(), printf()

package print\_Stream;

public class Research {

public static void main(String[] args) {

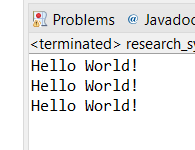
System.*out*.println("Hello World!");

System.*out*.printf("Hello %s!", "World");

System.*out*.print("\nHello World!");

}

}



* 1. Результат

Метод print() використовується для вивіду переданої інформації, в якості аргументів може приймати числа, строки та інші об’єкти.

Метод printf() використовується для форматованого вивіду. Форматує переданий текст використовуючи спеціальні аргументи.

Метод println() використовується для вивіду переданої інформації та переводу рядка.

### Дослідження класів StringBuilder та Scanner

Код програми наведено у лістингу 1.5.

* 1. Код програми

package string\_builder\_and\_Scanner;

import java.util.Scanner;

public class String\_builder\_and\_scanner {

public static void main(String[] args) {

StringBuilder sb = new StringBuilder("Enter information: ");

sb.append("Name/number of bad grades/average/\n");

sb.append("Example: Cat/2/3/");

System.*out*.println(sb);

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

sc.useDelimiter("/");

String name = sc.next();

int n = sc.nextInt();

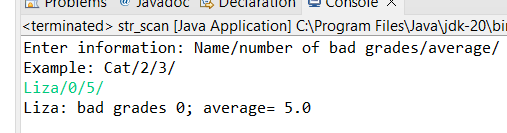
float b = sc.nextFloat();

sc.close();

System.*out*.println(name+ ": bad grades " + n + "; average= "+ b);

}

}



* 1. Результат

Для дослідження класу Scanner видеремо такі методи як nextInt() та nextLine(). nextInt() зчитує введене число, а nextLine() зчитує введену строку.

* 1. Дослідження nextInt() та nextLine()

package string\_builder\_and\_Scanner;

import java.util.Scanner;

public class Research\_scanner {

public static void main(String[] args) {

System.*out*.println("Enter number: ");//nextInt

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

int number = sc.nextInt();

System.*out*.println("Number " + number);

Scanner scan = new Scanner(System.*in*);//nextLine

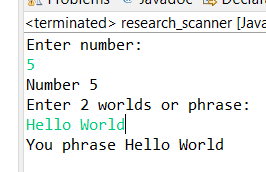
System.*out*.println("Enter 2 worlds or phrase: ");

String phrase = scan.nextLine();

System.*out*.println("You phrase " + phrase);

}

}



* 1. Результат

Для дослідження класу StringBuilder будемо використовувати такі методи як builder.append() для додавання символа, builder.deleteCharAt() для видалення символа, builder.reverse() для вивіду строки у зворотньому напрямку.

* 1. Дослідження класу StringBuilder

package string\_builder\_and\_Scanner;

public class Research\_string\_builder {

public static void main(String[] args) {

StringBuilder builder = new StringBuilder("Hello World");

builder.append("!");//add symbol !

String result = builder.toString();

System.*out*.println(result);

builder.deleteCharAt(11);//delete symbol

String result1 = builder.toString();

System.*out*.println(result1);

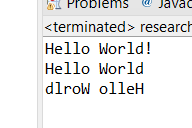
builder.reverse();//reverse

String result2 = builder.toString();

System.*out*.println(result2);

}

}



* 1. Результат

### Дослідження класів String та Integer

* 1. Код програми

package string\_integer;

import java.util.Scanner;

public class String\_integer {

public static void main(String[] args) {

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

System.*out*.println("Enter line the number: ");

String num = sc.nextLine();

String[] sArr = num.split(" ");

int[] numbers = new int[sArr.length];

int sum = 0;

int min = Integer.*MAX\_VALUE*;

int max = Integer.*MIN\_VALUE*;

for (int i = 0; i < sArr.length; i++)

{

numbers[i] = Integer.*parseInt*(sArr[i]);

sum += numbers[i];

if (numbers[i] < min)

{

min = numbers[i];

}

if (numbers[i] > max)

{

max = numbers[i];

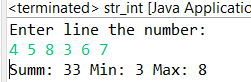
}

}

System.*out*.println("Summ: " + sum + " Min: " + min + " Max: " + max);

}

}



* 1. Результат

Для дослідження класу String використаємо метод equals() для порівняння рядків та метод equalsIgnoreCase() який теж використовується для порівняння рядків тільки без врахування регістру.

* 1. Дослідження класу String

package string\_integer;

public class Research\_string {

public static void main(String[] args) {

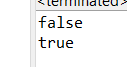
String name = "Liza";

System.*out*.println("liza".equals(name));

System.*out*.println("liza".equalsIgnoreCase(name));

}

}



* 1. Результат

Для дослідження класу Integer використаємо метод Integer.min(), Integer.max() для знаходження мінімального та максимального значення та метод Integer.parseInt()для конвертації рядка у число.

* 1. Дослідження класу String

package string\_integer;

public class Research\_int {

public static void main(String[] args) {

int x = 2;

int y = 5;

System.*out*.println(Integer.*max*(x,y));//max, min

System.*out*.println(Integer.*min*(x,y));

String str = "12";//convert string ro int

int i = Integer.*parseInt*(str);

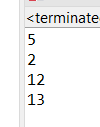
System.*out*.println(i);

int a = i + 1;

System.*out*.println(a);

}

}



* 1. Результат

### Дослідження класів Math та Arrays

Код програми наведено в лістингу 1.11, а результат виконання на рисунку 1.8

* 1. Код програми

package math\_Arrays;

import java.util.Arrays;

import java.util.Scanner;

public class Math\_Arrays {

public static void main(String[] args) {

System.*out*.println("The program was created by Zenenko Yelizaveta");

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

System.*out*.println("Enter length array: ");

int size = sc.nextInt();

if(size > 5) {//if more than 5 then plus 10

size += 10;

}

sc.close();

double[] arr = new double [size];

for(int i = 0; i < arr.length; i++) {

arr[i] = Math.*random*()\*100;

}

Arrays.*sort*(arr);

System.*out*.println("Numbers array");

for (double d : arr) {

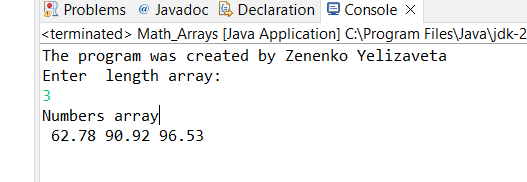
System.*out*.printf("%6.2f", d);

}

System.*out*.println();

}

}



* 1. Результат

Для дослідження класу Math використаємо метод pow() для піднесення до степеня та методи sqrt(), cbrt() для піднесення числа до квадратного та кубічного коренів.

* 1. Дослідження класу String

package math\_Arrays;

public class Research {

public static void main(String[] args) {

double a = Math.*pow*(3, 2);

System.*out*.println(a);

double b = Math.*cbrt*(a);

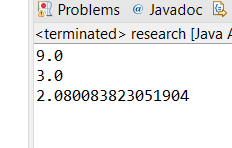
double c = Math.*sqrt*(a);

System.*out*.println(c);

System.*out*.println(b);

}

}



* 1. Результат

Для дослідження класу Arrays використаємо метод equals() для порівняння масивів та метод copyof() в цей метод ми передаємо масиві довжину нового масиву, в який копіюємо дані

* 1. Дослідження класу String

package math\_Arrays;

import java.util.Arrays;

public class Research\_array {

public static void main(String[] args) {

int [] arr1 = {0, 1, 2, 3};

int [] arr2 = {0, 1, 2, 3};

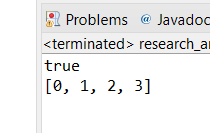
System.*out*.println(Arrays.*equals*(arr1, arr2));//true/false

int [] numbersCopy = Arrays.*copyOf*(arr1, arr1.length);//copy

System.*out*.println(Arrays.*toString*(numbersCopy));

}

}



* 1. Результат

## Висновки

За результатами виконання лабораторної роботи можна зробити наступні висновки.

1. Ми познайомилися з інегрованим середовищем розробки Eclipse та мовою програмування Java.
2. Навчились створювати пакет та Java клас.
3. Ознайомились з такими класами як System, PrintStream, StringBuilder, Scanner, String, Integer, Math, Arrays та дослідили їх деякі методи.

Список джерел посилання

1. Основи програмуванняна Java. Методичні вказівки до лабораторного практикуму та самостійної роботи з дисципліни «Об’єктно орієнтоване програмування» для студентів спеціальності 123 – “Комп’ютерна інженерія”. /Укл.: Бивойно П.Г., Бивойно Т.П. – Чернігів: ЧНТУ, 2020