***Звіт до лабораторної роботи №1***

***Тема*:** *Знайомство з середовищем та мовою програмування Java.*

***Мета:***

* *Сформувати уявлення про Java програми.*
* *Набути навиків у складанні, компіляції та виконанні найпростіших програм на мові Java.*
* *Розвинути навики у створенні програм в середовищі Eclipse.*

***Теоретичні відомості***

Початковий текст програми в Java розміщується в файлі з ім'ям створюваного класу і з розширенням .java.

Створимо клас MyFirstProgram, отже, ім'я файлу з початковим текстом буде MyFirstProgram.java. Імена програм мають бути такими, щоб ім'я відповідало призначенню і по імені можна було зрозуміти призначення програми (класу, методу, даних).

Проста програма, яка виводить привітання, виглядає таким чином:

/\* Моя перша програма MyFirstProgram.class \*/

**class** MyFirstProgram {

**public** **static** **void** main(String args[]) {

System.*out*.println("Привіт! Я студент групи ПІ-13-01(02), а це моя перша програма");

}

}

Перший рядок оточений символами /\* та \*/ - це коментар.

Другий рядок оголошує новий клас з ім'ям MyFirstProgram. Повне визначення класу мітиться фігурними дужками.

Третій рядок відкриває метод main(). Всі Java-додатки починаються з цього методу. Ключове слово public - специфікація доступу. Він має бути саме public, а не private, оскільки на початку програми викликається зовнішнім методом. Ключове слово static дозволяє викликати метод main() без обов'язкового створення конкретного екземпляра класу. Ключове слово void повідомляє компілятор, що функція main() не повертає значень.

У даного методу є один параметр - args, який є масивом екземплярів рядкового класу String. Змінна args приймає в себе будь-які параметри командного рядка. В наведеній вище програмі ця інформація ігнорується. Складні програми можуть мати багато класів, але тільки один з них повинен володіти методом main(), з якого починається виконання програми.

Створення аплетів (програм для Інтернет) не передбачає використання методу main(), оскільки Web-браузер застосовує інші засоби для їх запуску.

Опис методу main() мітиться внутрішніми фігурними дужками.

Четвертий рядок. System - це клас, що представляє доступ до системи, out - це вихідний потік, println() - вбудований метод, який виводить на екран текст, заданий всередині як параметр.

Для компіляції цієї програми в JDK слід виконати в командному рядку команду:

javac MyFirstProgram.java

В процесі компіляції створюється файл з тим же ім'ям і з розширенням .class, тобто MyFirstProgram.class.

Для виконання отриманої програми потрібно задати в командному рядку:

java MyFirstProgram

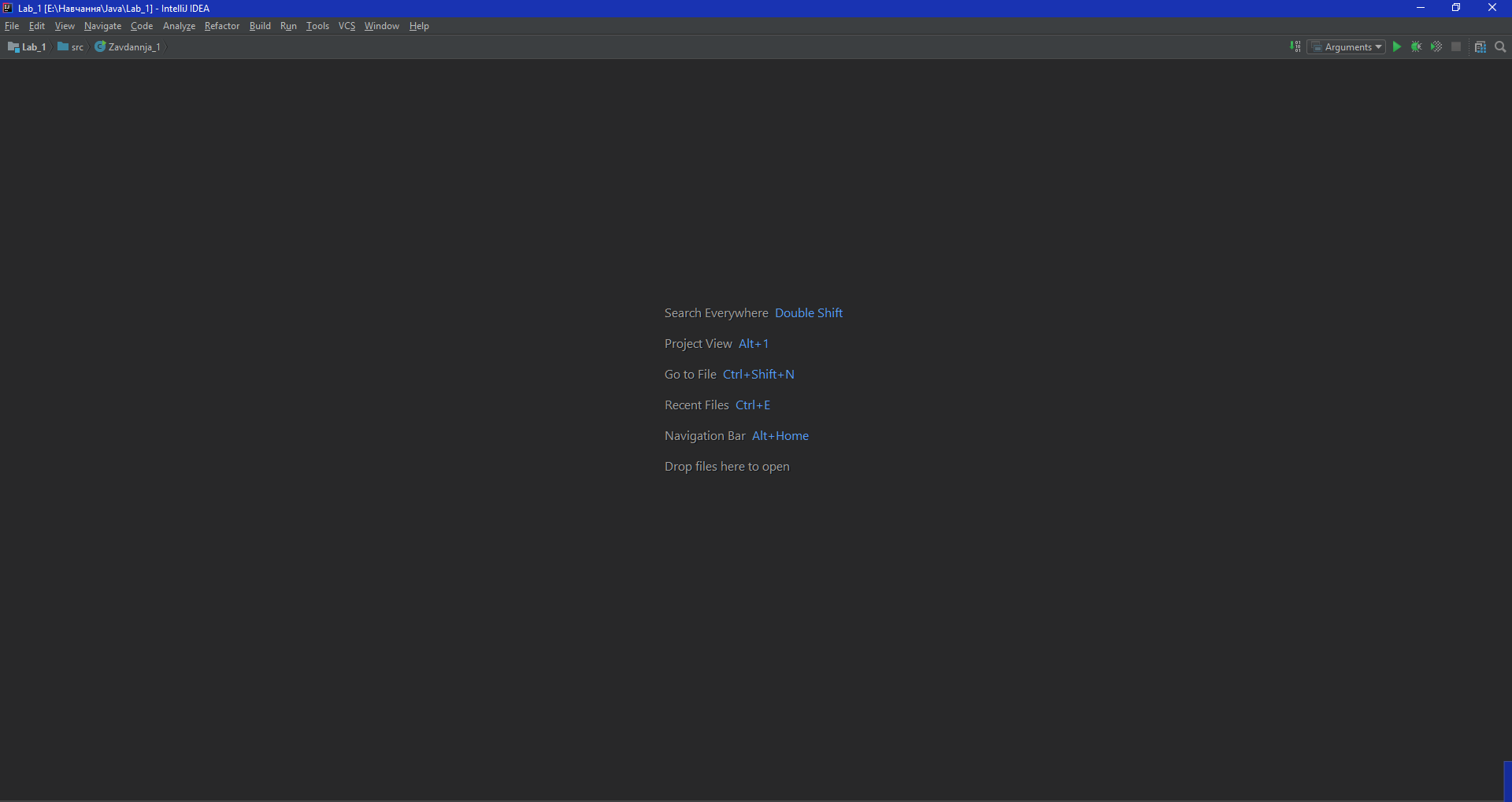
Середовище розробки Eclipse є безкоштовною програмою, яку можна скачати за адресою <http://eclipse.org>

Корисні пункти меню:

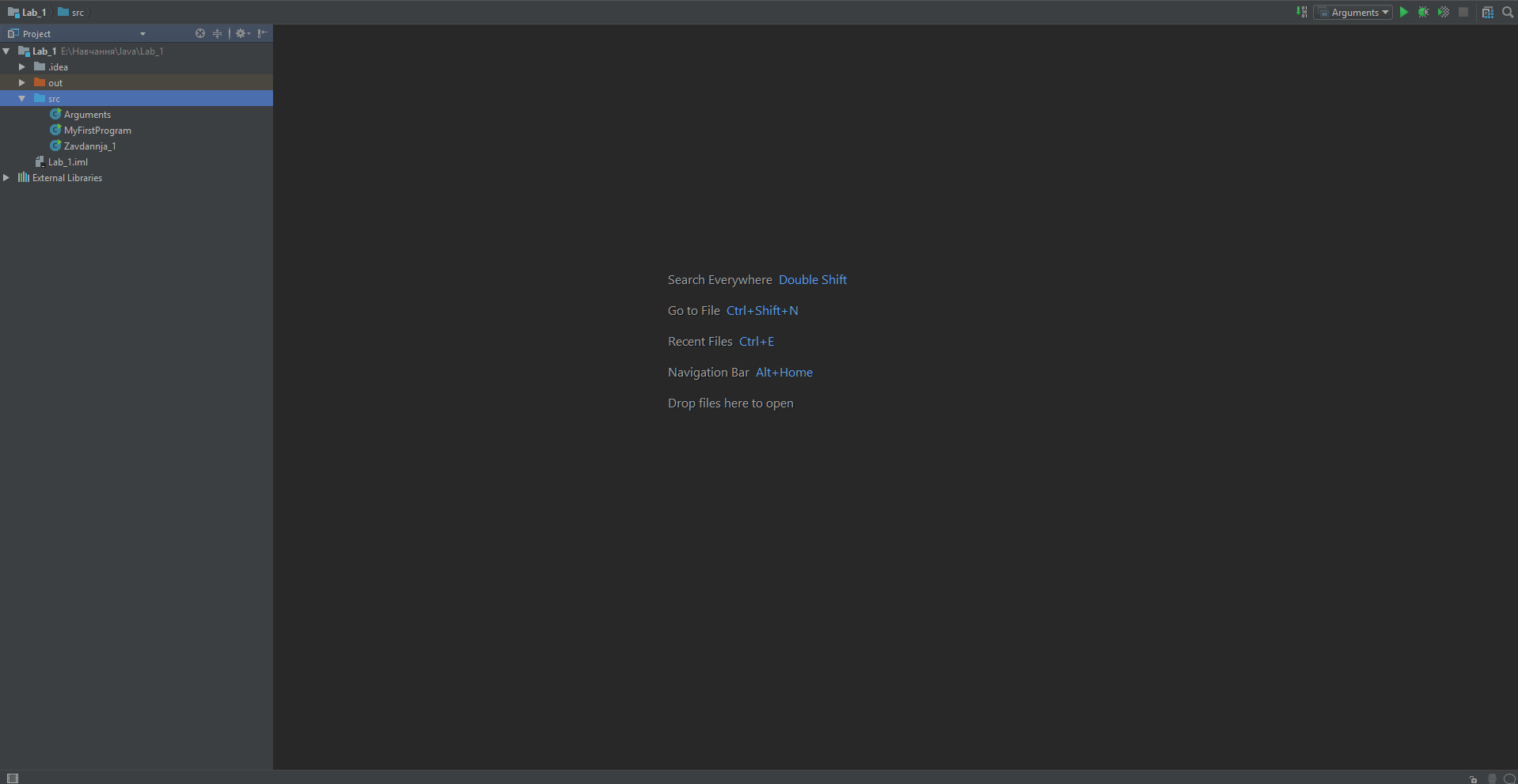
1. Source -> Format – відформатувати код відповідно до правил.
2. Source -> Organize Imports – видалити непотрібні типи та пакети, що імпортуються; відсортувати по порядку.
3. Refactor -> Move – перемістити клас/метод/поле в інше місце.
4. Refactor -> Rename – переіменувати клас/метод/поле/змінну.
5. Window->Preferences – відкрити вікно конфігурації середовища.
6. В дереві проекту вибрати проект, в контекстному меню вибрати Properties для перегляду та конфігурації проекту.
7. Run->Run – запустити програму на виконання.
8. Run -> Debug – запустити програму на виконання в режимі відладки.
9. Run -> Run Configurations... – редагувати конфігурації запуску програм.

***Хід роботи***

1. Запущено середовище програмування Eclipse. Закрито закладку Welcome щоб перейти до середовища розробки Eclipse.

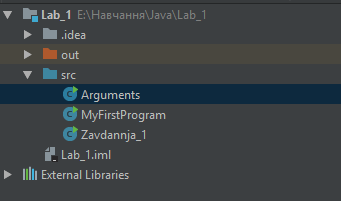


1. Середовище розбито на окремі частини: ліва, права та нижня. Вони легко конфігуруються: можна добавити нові панелі, скрити існуючі, перемістити тощо.

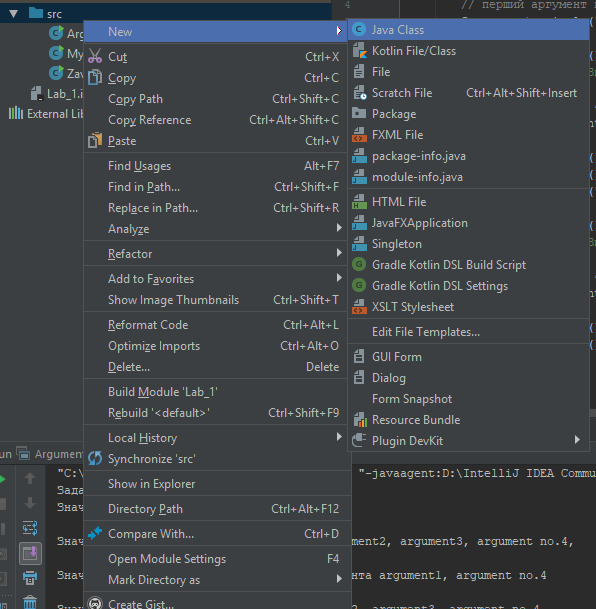


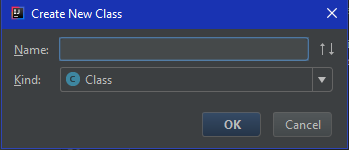
Є такі панелі:

1. Package explorer – дерево проектів та пакетів в межах проектів.
2. Problems – проблеми з програмним кодом.
3. Javadoc – документація на англійській мові для різних методів та класів.
4. Створено новий Java проект з іменем ZAVDANNJA\_1.

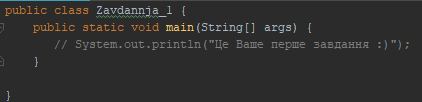


1. Добавимо новий клас під іменем LR\_1 до проекту.

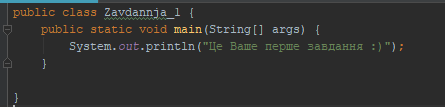




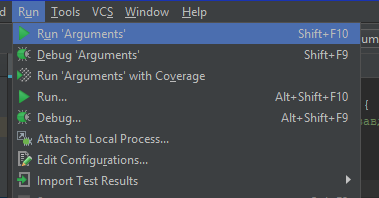
1. Виконано перейменування назви проекту із ZAVDANNJA\_1 на LR\_1 та перейменування назви класу із LR\_1 на ZAVDANNJA\_1 за допомогою команди з головного меню Refactor. Опишено у звіті, які саме дії виконує дана команда.
2. Якщо виконані Вами дії правильні, то в результаті Ви одержите вікно програми, яке має вигляд зображений нижче.



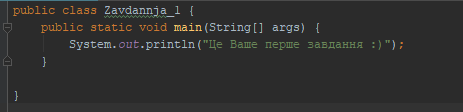
1. Дана програма нічого не виконує. Для того, щоб вона виводила простий текст на екран набрано рядок програмного коду, який виводить повідомлення «Це ваше перше завдання» :



1. Оскільки на панелі Problems знизу немає жодних повідомлень, значень код не містить помилок, і можна буде запускати його на виконання. Код було настроєно і виконано за допомогою команди Run -> Run Configurations.



В результаті виконання внизу добавиться ще одна панелька Console – консоль програми, де буде виведений рядок: **Це Ваше перше завдання :)**.





1. Описання коду:

**package** DEFAULT; - визначення «пакету» в якому знаходиться клас

**public** **class** ZAVDANNJA\_1 { -- оголошення класу

**public** **static** **void** main(String[] args) { -- оголошення методу main() і масиву args невизначеного розміру

System.***out***.println("Це Ваше перше завдання :)"); -- виведення на екран повідомлення «Це Ваше перше завдання :)»

} – закриття методу

} – закриття класу

1. Створено новий клас Arguments у вже створеному проекті в середовищі Eclipse.
2. Добавлено метод **public static void main(String[]args)** з пустим тілом.

**public** **class** Arguments {

// Метод main є точкою входу в програму.

**public** **static** **void** main(String[] args) {

}

}

1. Добавлено ще одну конфігурацію запуску в діалоговому вікні Run, але цього разу для виконання коду із класу Arguments, а не Zavdannja\_1.
2. Виконано програму з використанням добавленої конфігурацію. Конфігурація добавлена успішно, якщо не виникло помилок, а на консоль нічого не було виведено.
3. Змінено код щоб він виводив кількість аргументів, та значення 1 аргументу:

**public** **class** Arguments {

**public** **static** **void** main(String[] args) {

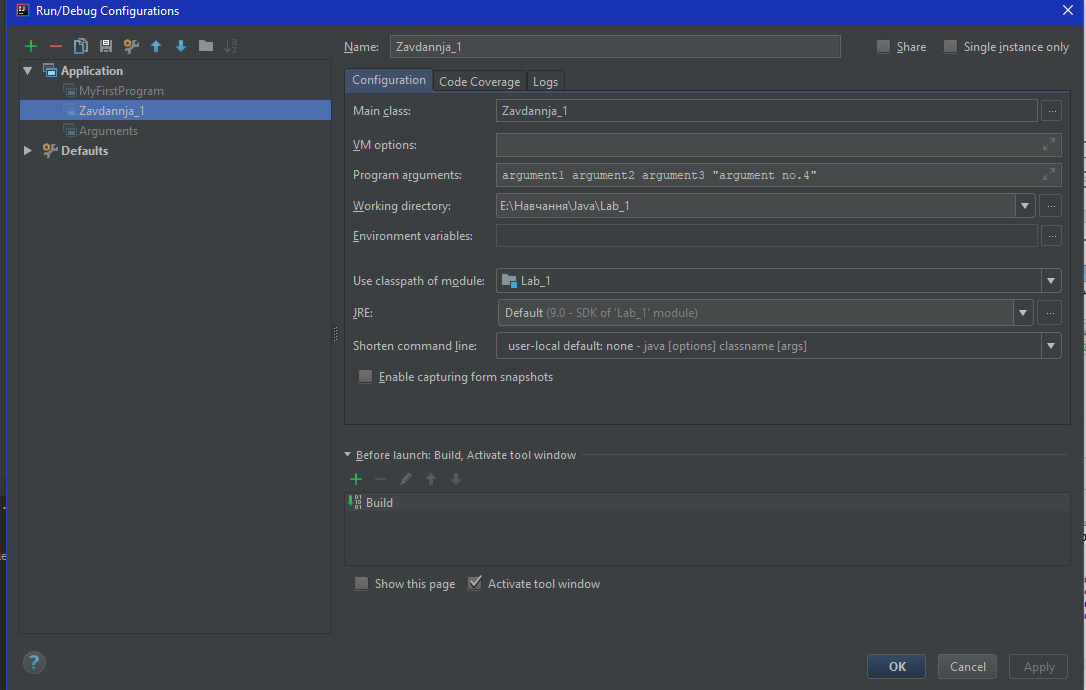
System.***out***.println("Задано " + args.length + " аргуметнів");

System.***out***.println("Значення 1го аргумента " + args[0]);

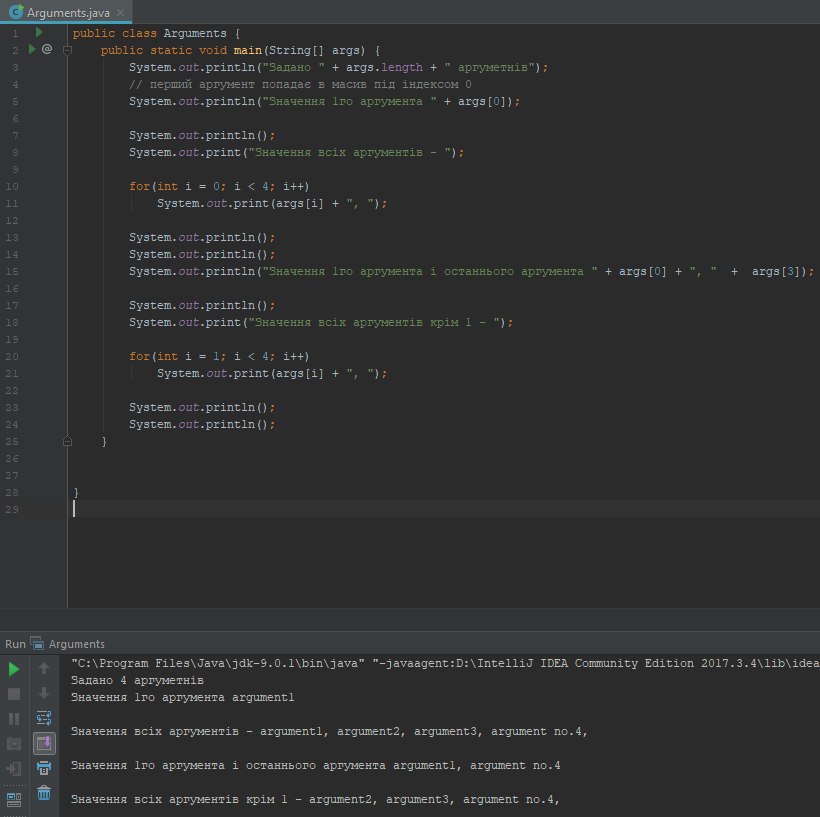
}

}

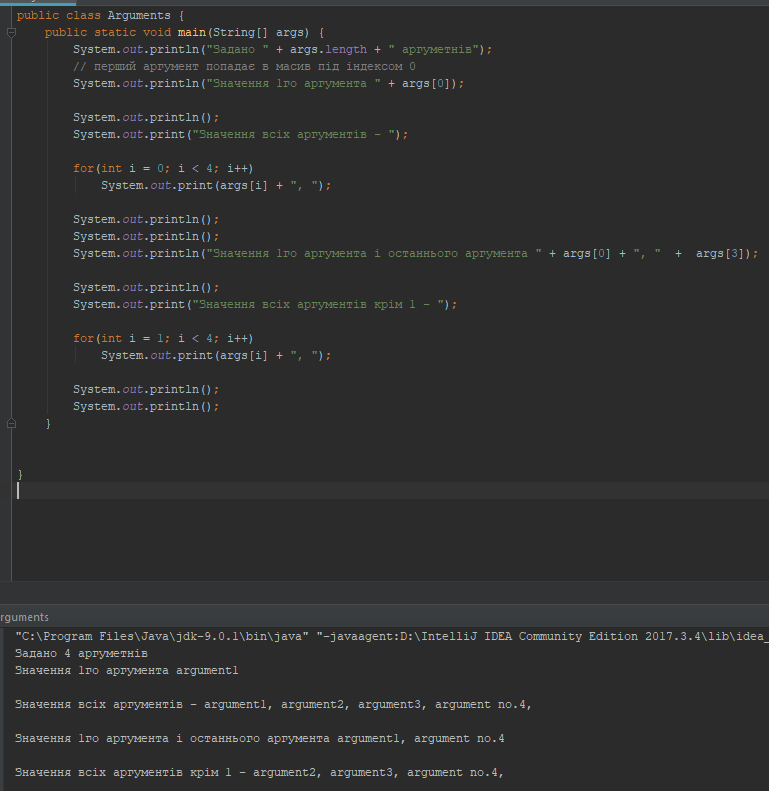
1. Для того, щоб задати параметри для виконання програми відредаговано діалогове вікно Run Configurations конфігурацію для запуску класу Arguments.



1. Виконано програму.



1. Змінено програму таким чином, щоб вона виводила:
2. Всі параметри.
3. Перший та останній параметр.
4. Всі крім першого параметра.



***Контрольні запитання:***

1. *Як створити новий проект в Eclipse?*

*Новий Java проект в Eclipse створюється за таким шляхом File->New->Project-> Java Project*

1. *Як добавити новий клас до проекту в Eclipse?*

*Новий клас в Eclipse створюється за таким шляхом File->New->Class*

1. *Що необхідно добавити до класу X, для того, щоб можна було виконувати його?*

*Щоб клас Х можливо було виконати у нього потрібно добавити будь-який метод (можна пустий).*

1. *Як дізнатися скільки аргументів задано програмі?*

*Дізнатись кількість аргументів можна за допомогою команди length() її структура така: <Назва масиву>.length().*

1. *Як задати аргументи в середовищі Eclipse?*

*Задати аргументи у середовищі Eclipse можна за таким шляхом Run->Run Configurations->Arguments->В полі «Program Arguments» через «Пробел» задати аргументи для програми, якщо аргумент містить пропуск його необхідно написати в лапках.*

**Висновок:** на даній лабораторній роботі було сформовано уявлення про Java програми, набуто навиків у складанні, компіляції та виконанні найпростіших програм на мові Java, а також розвинуто навики у створенні програм в середовищі Eclipse.