

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра «Електронних обчислювальних машин»



Звіт
з лабораторної роботи № 1
з дисципліни: «Кросплатформенні засоби програмування»
на тему: «Основи розробки програм мовою Java»

Виконав:

студент групи КІ-306

Ширий Б. І.

Прийняв:

доцент кафедри ЕОМ

Іванов Ю. С.

МЕТОДИЧНІ ВІДОМОСТІ РОБОТИ

МЕТА

Оволодіти основами розробки програм мовою Java.

ЗАВДАННЯ

№1

Написати, скомпілювати, налагодити та запустити програму на мові Java, яка виводитиме на екран прізвище, ім'я та групу автора програми.

№2

Скласти звіт про виконану роботу з приведенням тексту програми та результату її виконання.

№3

Дати відповідь на контрольні запитання.

ПОРЯДОК ВИКОНАННЯ

№1. JDK ДЛЯ JAVA SE 6 UPDATE 35.

Завантажити та встановити JDK для Java SE 6 Update 35 з сайту компанії Oracle: <http://www.oracle.com>.

№2. ECLIPSE IDE FOR JAVA DEVELOPERS.

Після успішного встановлення JDK для Java SE 6 Update 35 необхідно завантажити Eclipse IDE for Java Developers з сайту <http://www.eclipse.org>.

№3. РОЗАРХІВАЦІЯ ЗАВАНТАЖЕНОГО АРХІВУ.

Після успішного завантаження необхідно розархівувати завантажений архів туди, де ви хочете розташувати Eclipse IDE. Якщо все зроблено коректно, то після запуску на виконання файлу Eclipse.exe з розархівованого каталогу Eclipse і вказання місця розташування проектів Java має з'явитися вікно, яке зображено на рис. 1.1.

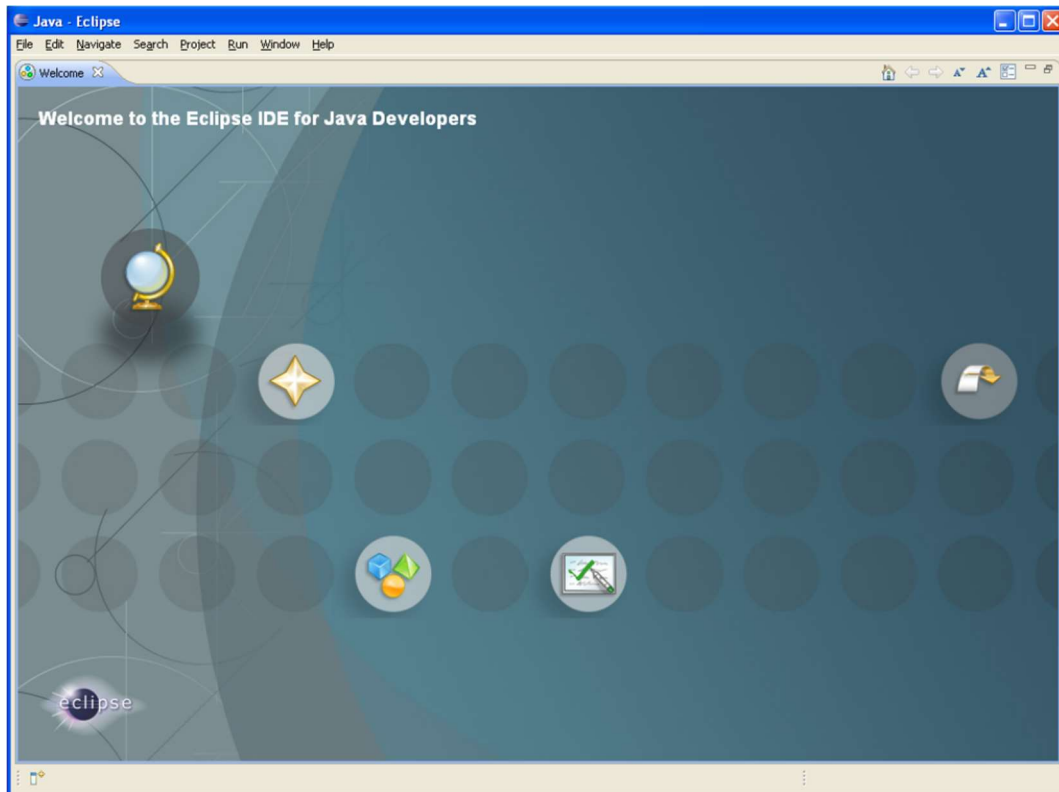


Рисунок 1.1. Головно вікно Eclipse IDE for Java Developers.

№4. СТВОРЕННЯ НОВОГО ПРОЕКТУ.

Почати створення нового проекту. Для цього слід викликати підпункт меню File->New->Project... У вікні, що відкриється слід вибрати Java Project (рис. 1.2) та натиснути кнопку "Next>".

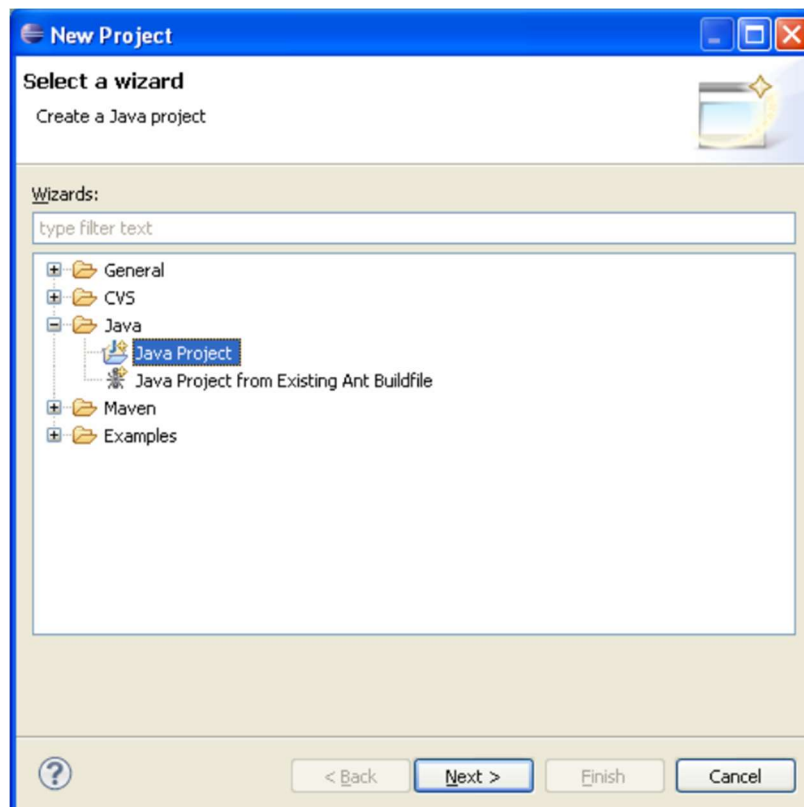


Рисунок 1.2. Діалогове вікно вибору типу проекту в Eclipse IDE.

№5. НАЛАШТУВАННЯ НОВОГО ПРОЕКТУ.

У вікні, що відкрилося, необхідно вказати назву нового проекту "HelloJava", місце розташування проекту, версію JRE на яке орієнтована програма (JavaSE-1.6), топологію проекту (Project layout) та натиснути кнопку "Next>" (рис. 1.3).

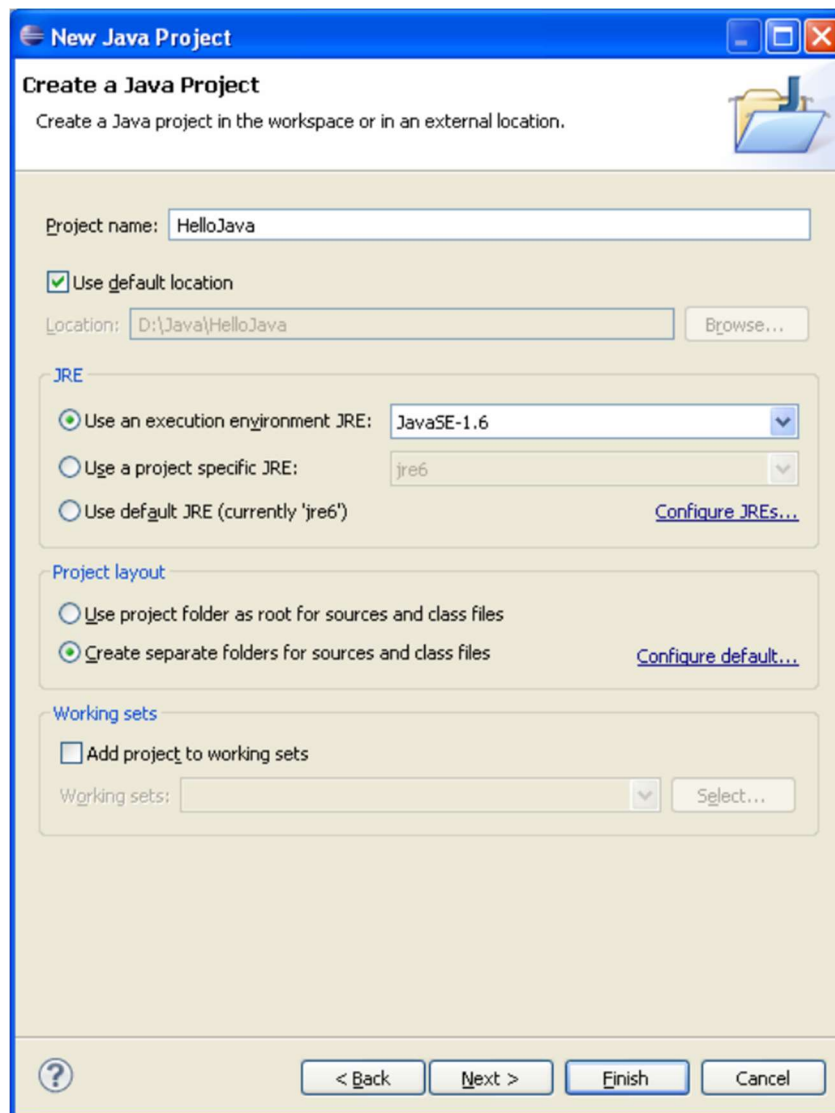


Рисунок 1.3. Діалогове вікно створення нового проекту в Eclipse IDE.

№6. НАВІГАЦІЯ ПО ECLIPSE IDE.

У вікні, що відкрилося, натиснути кнопку "Finish" (рис. 1.4). Після закривання вкладки Welcome має відкритися вікно, що зображене на рис. 1.5.

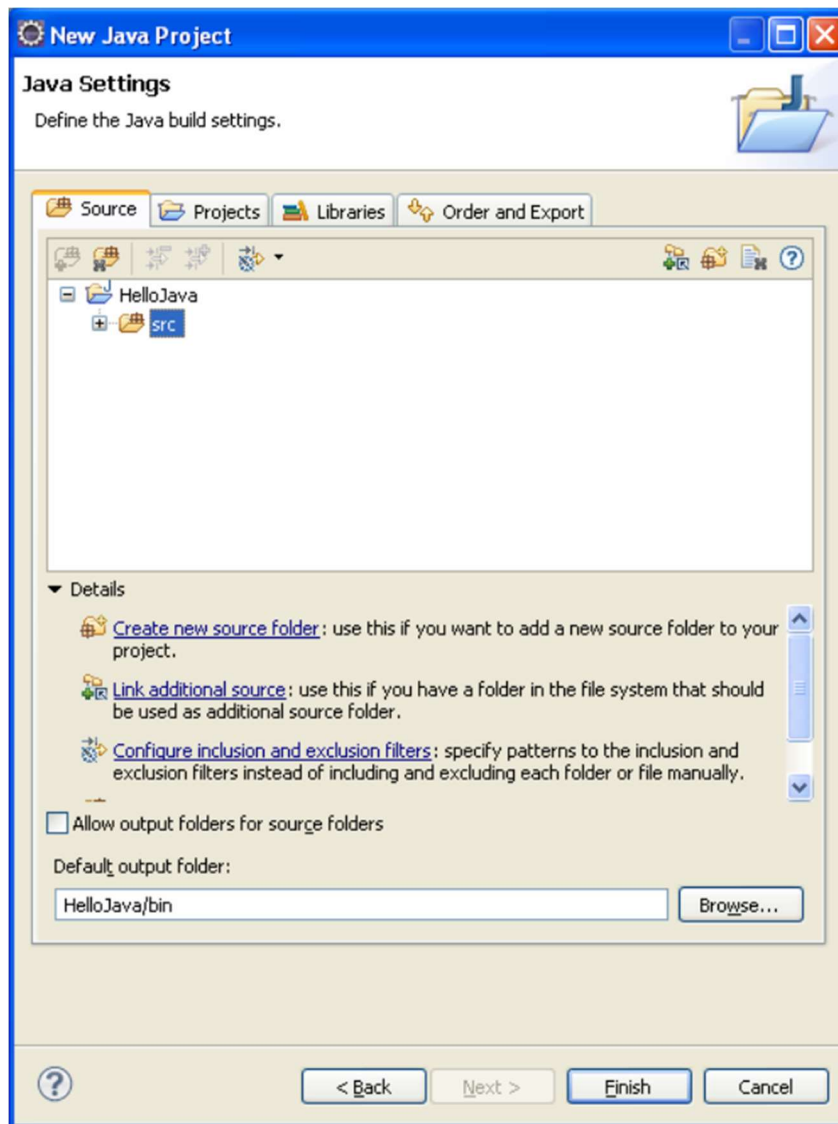


Рисунок 1.4. Діалогове вікно налаштування властивостей побудови.

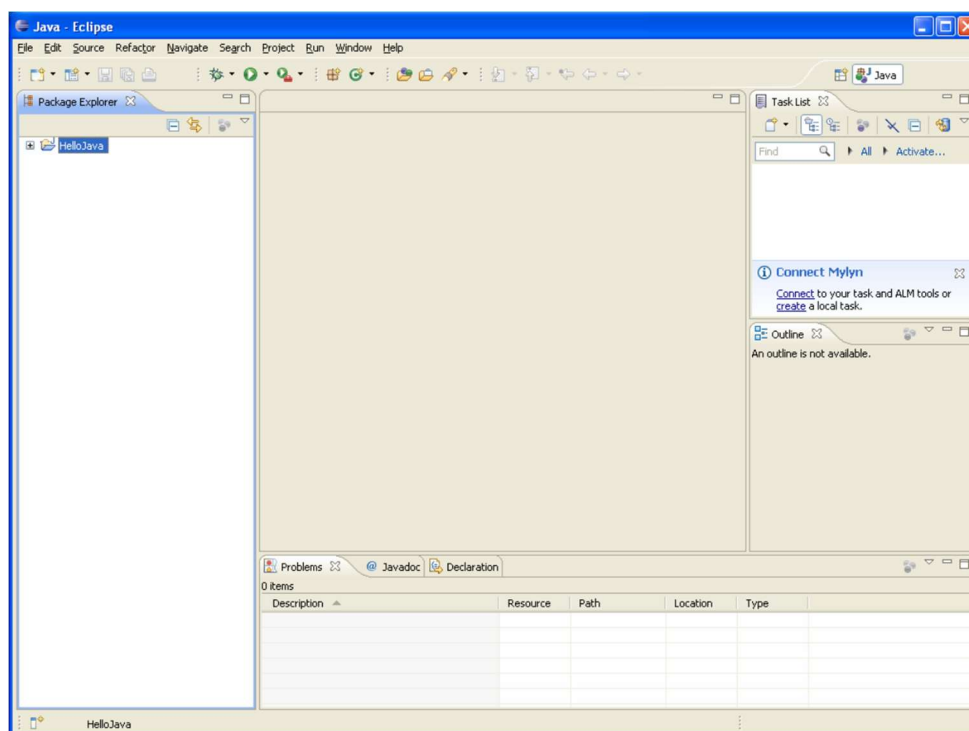


Рисунок 1.5. Серовище Eclipse IDE з відкритим проектом Java.

№7. СТВОРЕННЯМ КАТАЛОГУ.

Викликати підпункт меню File->New->File. У вікні, що відкрилося слід задати каталог, де розташовуватиметься новий файл з кодом "HelloJava/src" та назву файлу "Hello.java" та натиснути кнопку "Finish" (рис. 1.6).

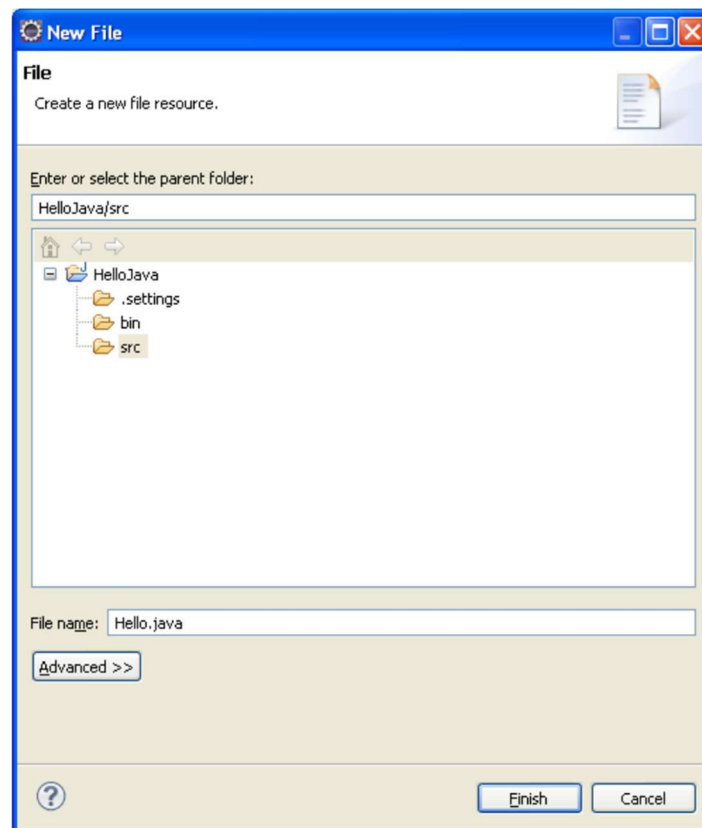


Рисунок 1.6. Створення нового файлу.

№8. НАПИСАННЯ ПРОГРАМИ.

У створеному файлі написати програму згідно завдання. Зверніть увагу, що програма має обов'язково містити public клас, назва якого співпадає з назвою файлу.

№9. ТЕСТУВАННЯ ПРОГРАМИ.

Запустити програму на виконання. Для цього слід вибрати підпункт меню Run->Run. В результаті виконання тестової програми, код якої наведено нижче, в консолі відобразиться текст "Hello, Java" (рис. 1.7).

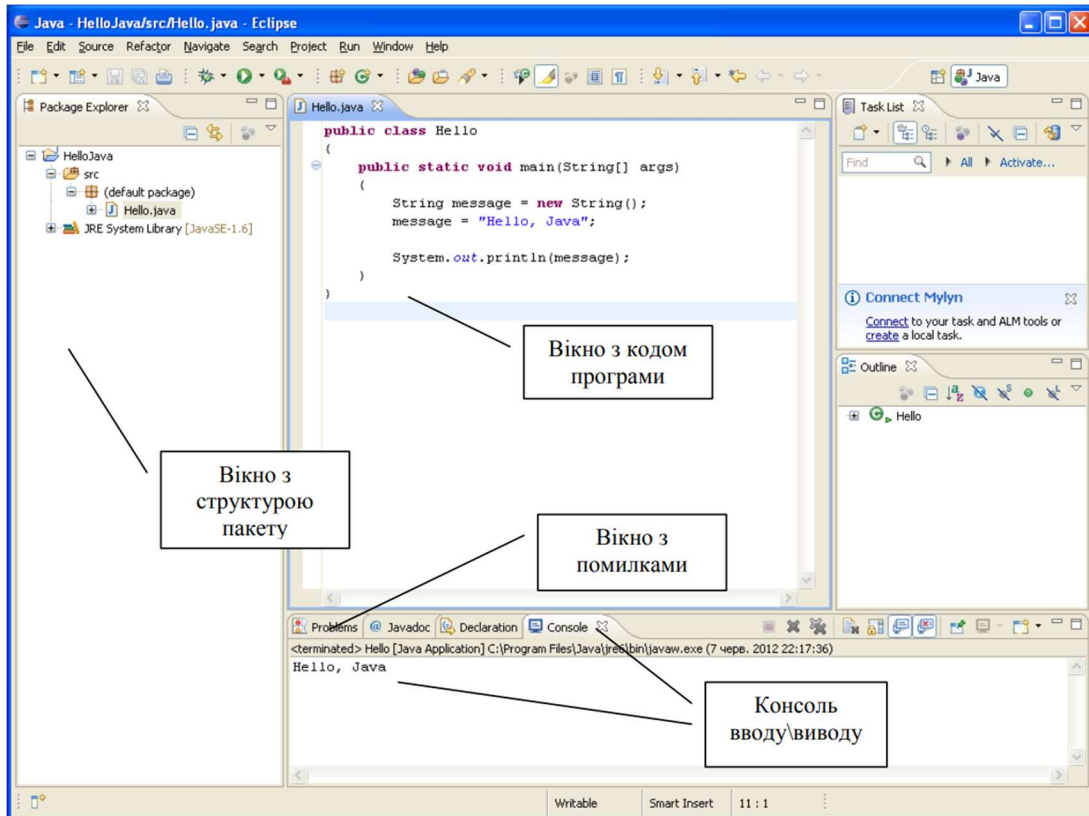


Рисунок 1.7. Результат виконання тестової програми.

№10. ОЗНАЙОМЛЕННЯ З НАЛАГОДЖУВАЧЕМ.

Встановити точки переривання, запустити налагоджувач (Run->Debug) та покроково дослідити процес виконання програми (рис. 1.8).

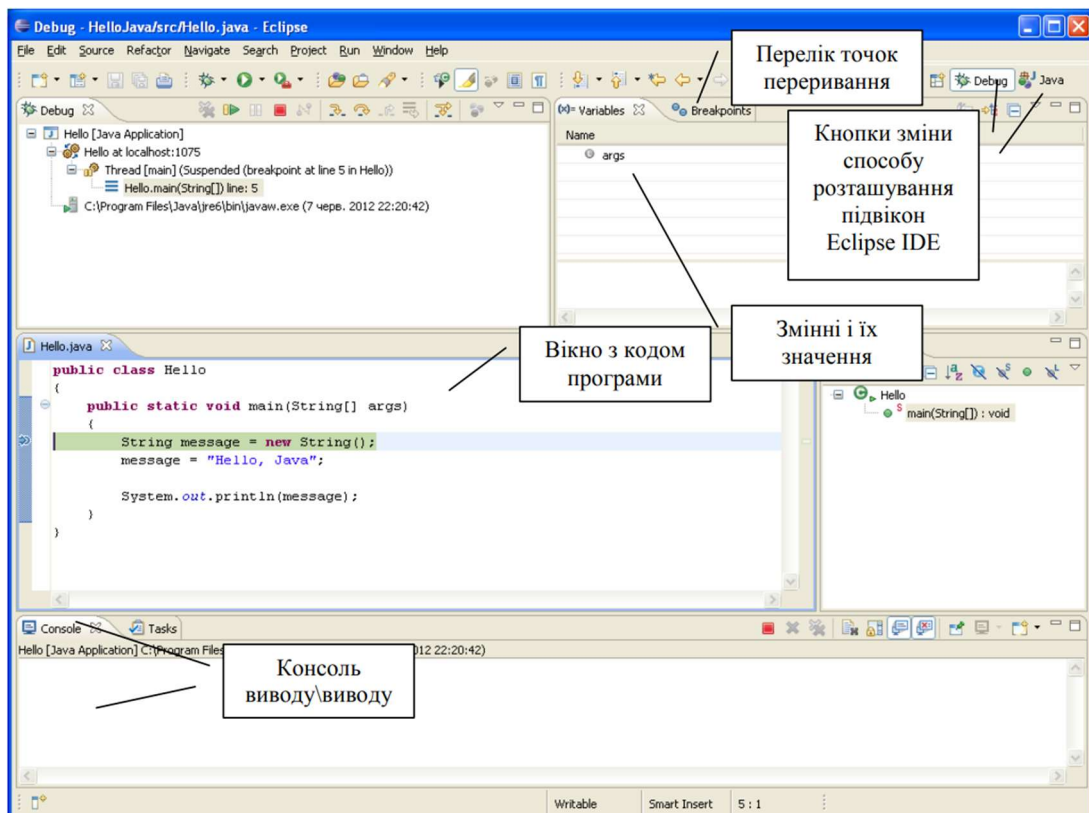


Рисунок 1.8. Процес налагодження програми.

ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ

ПОРЯДОК ВИКОНАННЯ

№1. JDK ДЛЯ JAVA SE 6 UPDATE 35.

Завантажив та встановив JDK для Java SE 17.0.8 (рис 2.1 та 2.2) з сайту компанії Oracle: <http://www.oracle.com>, оскільки Java SE 6 Update 35 не підтримує Windows 11.



Рисунок 2.1. Початок встановлення JDK для Java SE 17.0.8.



Рисунок 2.2. Успішне встановлення JDK для Java SE 17.0.8.

№2. ECLIPSE IDE FOR JAVA DEVELOPERS.

Після успішного встановлення JDK для Java SE 17.0.8 завантажив Eclipse IDE for Java Developers (рис. 2.3 та 2.4) з сайту <http://www.eclipse.org>.

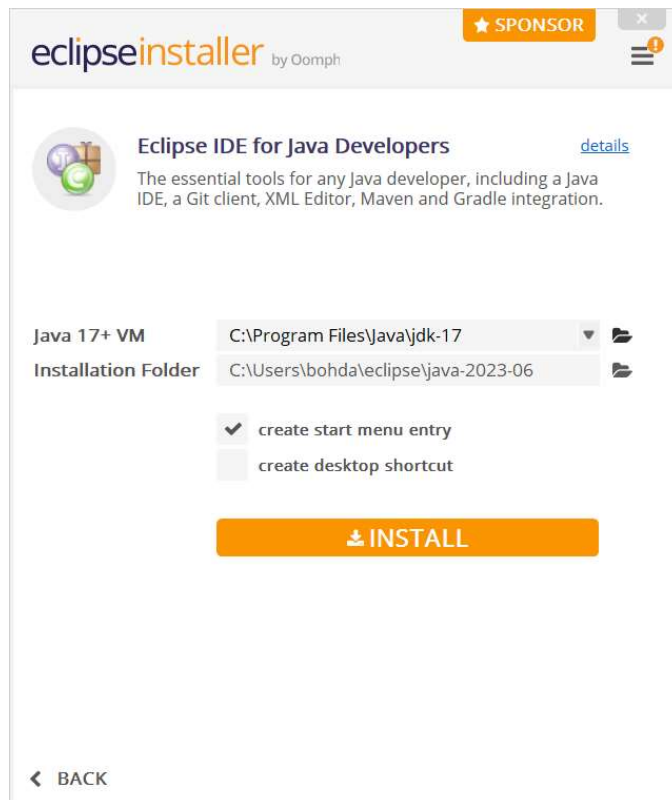


Рисунок 2.3. Початок встановлення *Eclipse IDE for Java Developers*.

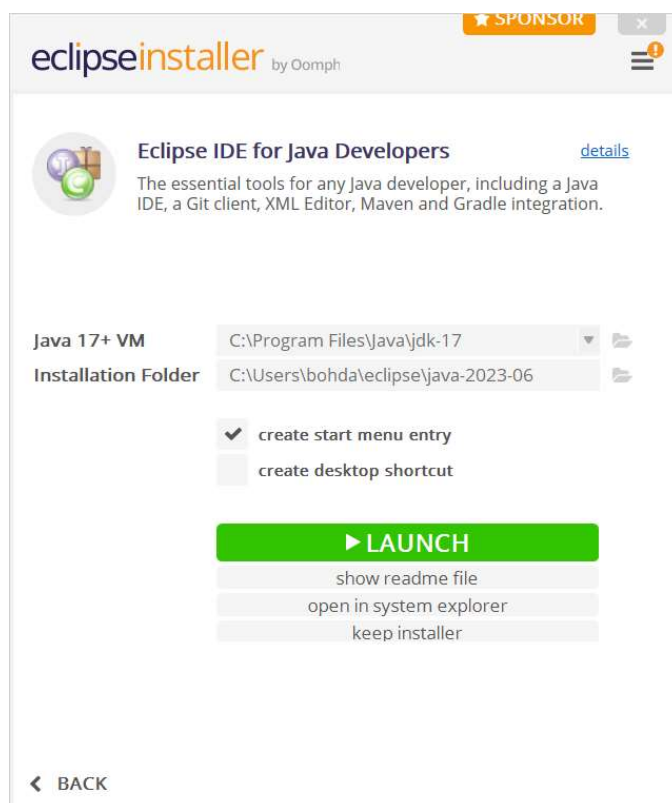


Рисунок 2.4. Початок встановлення *Eclipse IDE for Java Developers*.

№4. СТВОРЕННЯ НОВОГО ПРОЕКТУ.

Почав створення нового проекту. Для цього викликав підпункт меню File->New->Project... У вікні, що відкриється слід вибрати Java Project (рис. 1.2) та натиснути кнопку "Next>".

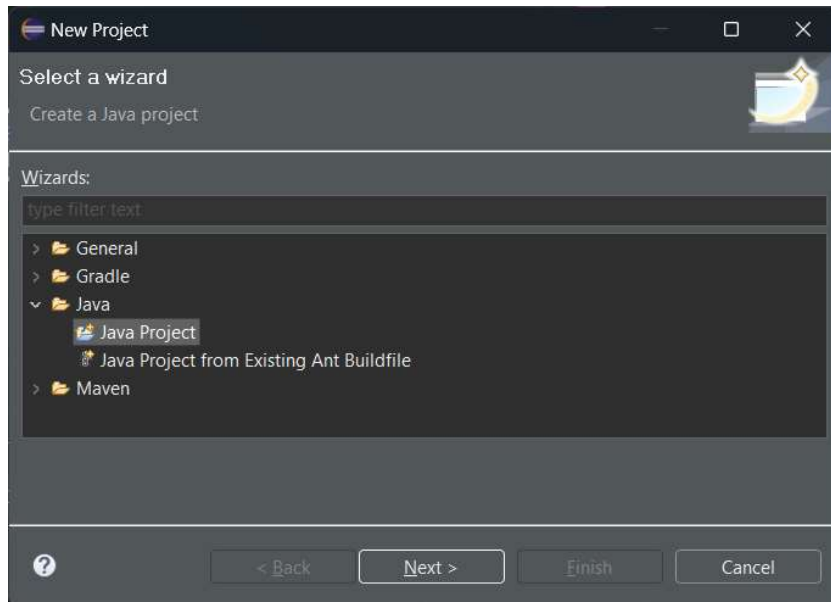


Рисунок 2.5. Діалогове вікно вибору типу проекту в Eclipse IDE.

№5. НАЛАШТУВАННЯ НОВОГО ПРОЕКТУ.

У вікні, що відкрилося, вказав назву нового проекту "CPPT_labN1_Shyryi_HelloJava", місце розташування проекту, версію JRE на яке орієнтована програма (JavaSE-17), топологію проекту (Project layout) та натиснув кнопку "Next>" (рис. 2.6).

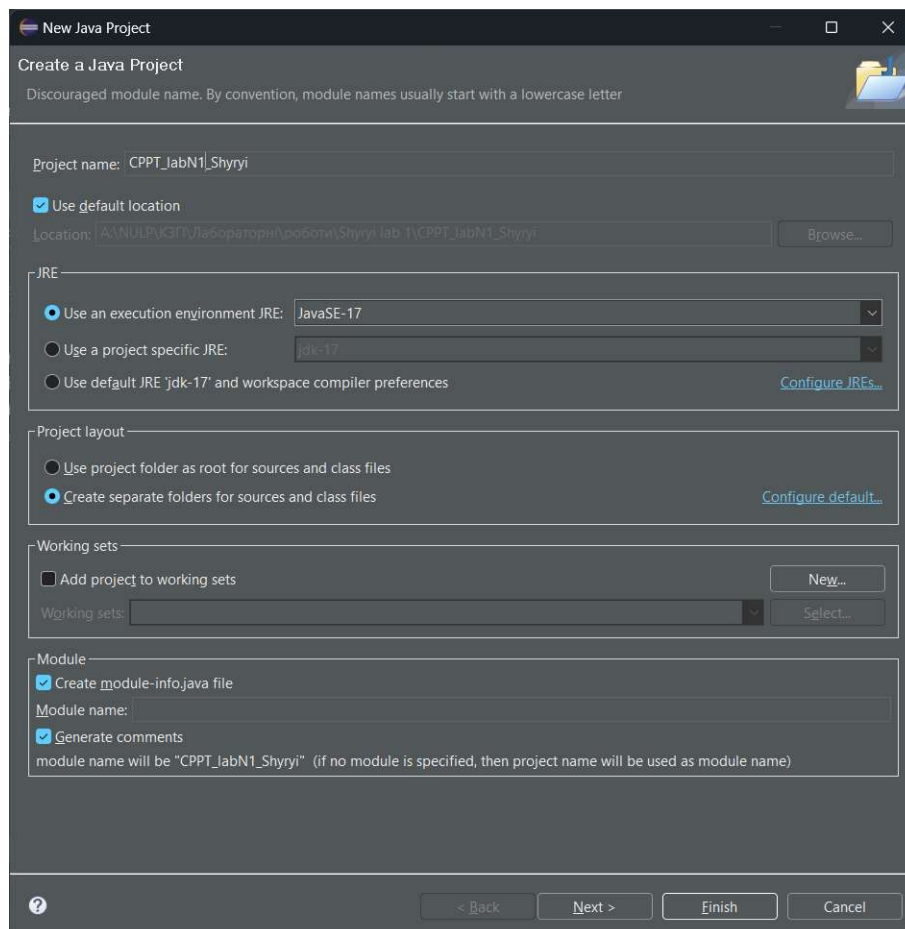


Рисунок 2.6. Діалогове вікно створення нового проекту в Eclipse IDE.

№6. НАВІГАЦІЯ ПО ECLIPSE IDE.

Після закривання вкладки Welcome відкрилося вікно проекту, що зображене на рис. 2.7.

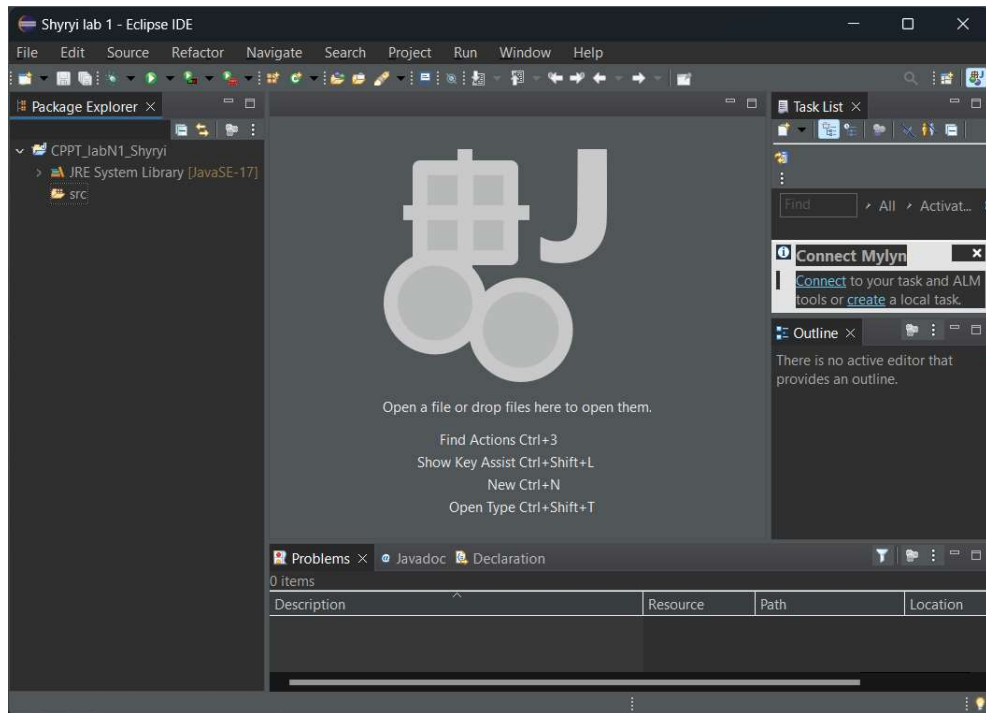


Рисунок 2.7. Середовище Eclipse IDE з відкритим проектом Java.

№7. СТВОРЕННЯМ КАТАЛОГУ.

Викликав підпункт меню File->New->File. У вікні, що відкрилося задав каталог, де розташовуватиметься новий файл з кодом "CPT_labN1_Shyryi_HelloJava/src" та назву файлу "NamesGroup.java" та натиснути кнопку "Finish" (рис. 2.8).

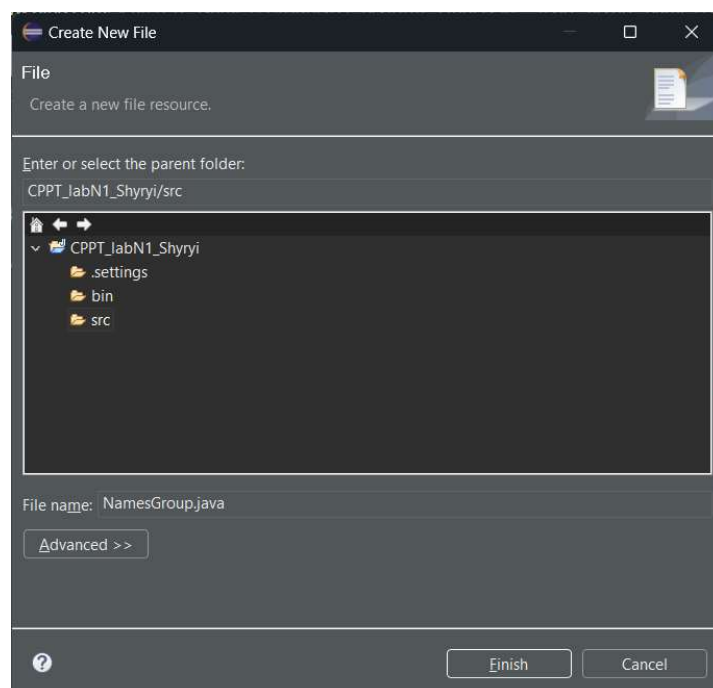


Рисунок 2.8. Створення нового файлу.

№8. НАПИСАННЯ ПРОГРАМИ.

У створеному файлі написав програму згідно завдання, що наведена у лістингу 2.1.

Лістинг 2.1. Код програми.

```
public class NamesGroup
{
    public static void main(String[] args)
    {
        System.out.println("Shyryi Bohdan, CI-306");
    }
}
```

№9. ТЕСТУВАННЯ ПРОГРАМИ.

Запустив програму на виконання. В результаті виконання програми в консолі відобразиться текст "Shyryi Bohdan, CI-306" (рис. 2.9).

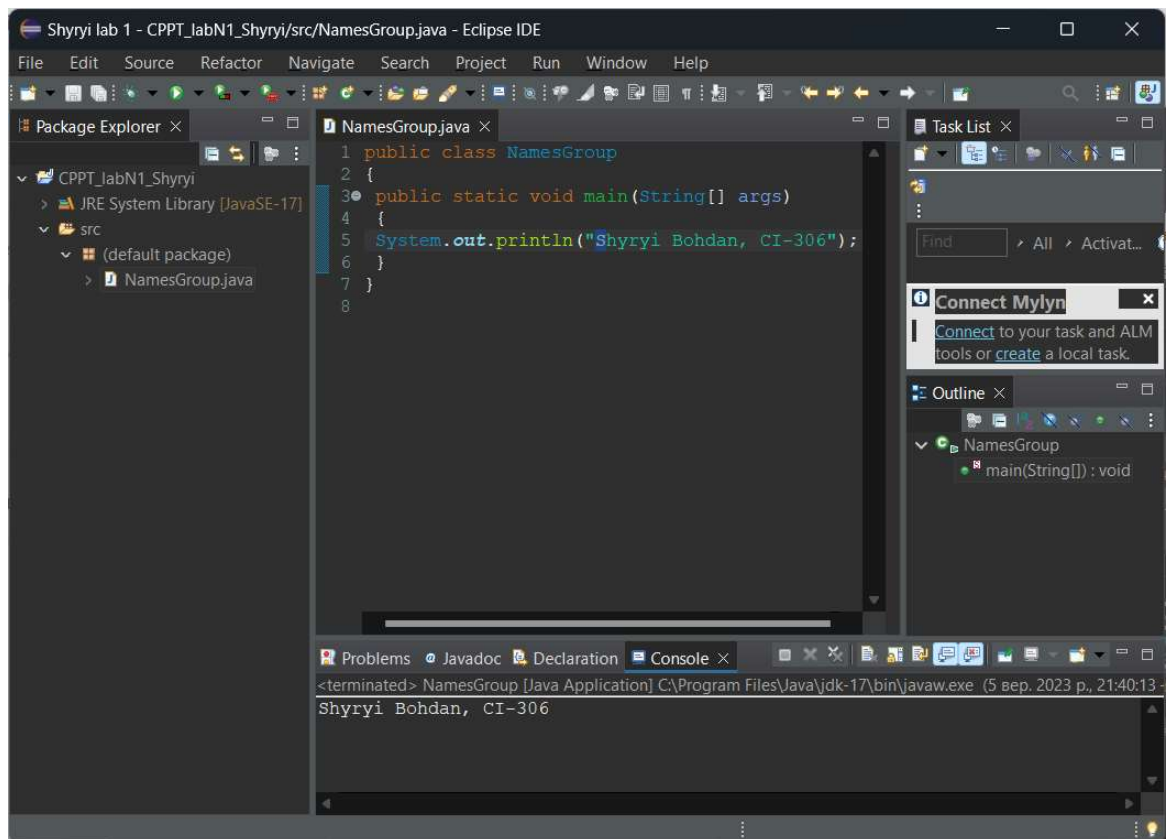


Рисунок 2.9. Результат виконання тестової програми.

№10. ОЗНАЙОМЛЕННЯ З НАЛАГОДЖУВАЧЕМ.

Встановив точки переривання, запустив налагоджувач (Run->Debug) та покроково дослідив процес виконання програми (рис. 2.10).

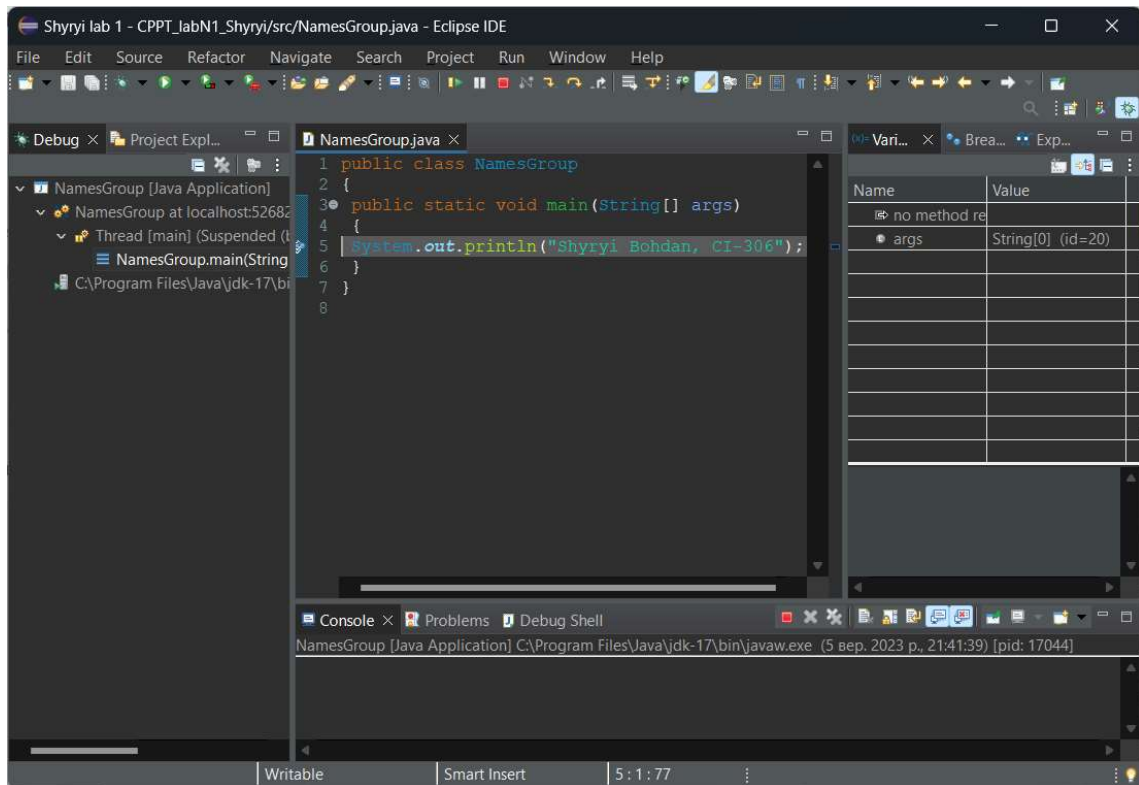


Рисунок 2.10. Процес налагодження програми.

РЕЗУЛЬТАТИ ВИКОНАННЯ

У лістингу 2.2 навів повний код програми, що написав протягом виконання лабораторної роботи.

Лістинг 2.2. Повний код програми.

```
public class NamesGroup
{
    public static void main(String[] args)
    {
        System.out.println("Shyryi Bohdan, CI-306");
    }
}
```

Нижче, навів вивід програми:

Shyryi Bohdan, CI-306

КОНТРОЛЬНІ ПИТАННЯ

№1. ЩО ТАКЕ JRE?

JRE (Java Runtime Environment) - це середовище виконання програм Java. Воно надає віртуальну машину Java (JVM), яка виконує Java-програми, а також стандартні бібліотеки класів, необхідні для виконання цих програм.

№2. ЯКА СТРУКТУРА JRE?

JRE включає в себе два основних компонента: стандартні бібліотеки класів Java і віртуальну машину Java (JVM). JVM, у свою чергу, складається з блоку перевірки безпечності коду, інтерпретатора або JIT (Just-in-time) компілятора і блоку керування пам'яттю.

№3. ЩО TAKE JVM?

JVM (Java Virtual Machine) - це віртуальна машина Java, яка виконує байт-код Java-програм. Вона відповідає за запуск та виконання програм, перетворюючи байт-код у машинний код або інтерпретуючи його.

№4. ЩО TAKE JIT КОМПІЛЯТОР?

JIT (Just-in-time) компілятор - це компонент JVM, який компілює байт-код програми в машинний код в процесі її виконання. Це дозволяє підвищити швидкість виконання програми, оскільки машинний код виконується швидше за інтерпретацію байт-коду.

№5. ЩО TAKE JDK?

JDK (Java Development Kit) - це набір інструментів для розробки програм на мові Java. Він включає в себе компілятор `javac`, бібліотеки, документацію та інші розробницькі інструменти, необхідні для написання та компіляції програм.

№6. В ЧОМУ ПОЛЯГАЄ ВІДМІННІСТЬ МІЖ JRE І JDK?

Відмінність між JRE і JDK полягає в їх призначенні і складі. JRE призначений для виконання Java-програм і включає в себе лише середовище виконання та стандартні бібліотеки. З іншого боку, JDK призначений для розробки Java-програм і включає в себе компілятор `javac` та інші розробницькі інструменти, дозволяючи створювати та компілювати програми.

№7. ЯКІ ІНТЕГРОВАНІ СЕРЕДОВИЩА РОЗРОБКИ ПРОГРАМ МОВОЮ JAVA ВИ ЗНАЄТЕ?

Існує кілька інтегрованих середовищ розробки для мови Java, таких як NetBeans IDE, Eclipse IDE, IntelliJ IDEA, JDeveloper та BlueJ. Ці середовища надають зручний інтерфейс для написання, компіляції та відлагодження програм Java.

№8. НАВЕДІТЬ ОСНОВНІ ЕТАПИ НАПИСАННЯ ТА ЗАПУСКУ ПРОГРАМИ НАПИСАНОЇ МОВОЮ JAVA.

- Написання програми у файлі з розширенням *.java, включаючи використання текстового редактора або інтегрованого середовища розробки (IDE).
- Компіляція програми за допомогою компілятора javac, що перетворює *.java файл у *.class файл, який містить байт-код.
- Виконання програми за допомогою JRE, вказуючи назву класу, що містить метод main: java filename.

ВИСНОВОК

Протягом цієї лабораторної роботи, я ознайомився з новим для себе Eclipse IDE для програмування Java, оскільки раніше для цих потреб я використовував IntelliJ IDEA. Загалом, я створив свій перший проект, та написав простеньку програму з виводом стрічки: «Shyryi Bohdan, CI-306», яку вивів без створення змінної, а зразу помістивши її у `System.out.println`. На жаль, оскільки я користуюся Windows 11, то Java SE 6 Update 35 вже не сумісна з моєю версією Windows, тому натомість я встановив Java SE 17.0.8, оскільки вона є найстабільнішою версією на даний момент та має довго тривалу підтримку від компанії.