

Міністерство освіти і науки України  
Національний університет „Львівська політехніка”



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

Виконав: ст.групи КІ-36  
Борисов І. С.

Прийняв, та перевірів :  
Іванов Ю. С.

**Львів 2022**

**Тема:** основи розробки мовою Java.

**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.

### Хід роботи:

1. Написав програму на мові програмування Java, яка виводить на екран прізвище, ім'я та групу.

```
public class Main {  
    public static void main(String[] args) {  
        String message;  
        message = "Borysov Ihor KI-36";  
        System.out.println(message);  
    }  
}
```

2. Скомпілював та запустив програму.

```
Borysov Ihor KI-36  
  
Process finished with exit code 0
```

### Контрольні запитання

#### 1. Що таке JRE?

JRE (Java Runtime Environment) – це мінімальна реалізація віртуальної машини, необхідна виконання Java додатків, без компілятора та інших засобів розробки. Складається з віртуальної машини та бібліотек Java класів.

#### 2. Яка структура JRE?

Структурно, це середовище складається двох елементів: стандартних бібліотек класів Java та віртуальної машини Java, яка у термінології Java називається JVM (Java Virtual Machine).

### 3. Що таке JVM?

JVM (Java Virtual Machine) – це віртуальна машина Java – основна частина виконуючої системи Java, так званої Java Runtime Environment (JRE). Віртуальна машина Java виконує байт-код Java, попередньо створений з вихідного тексту Java програми компілятором Java (javac). JVM забезпечує платформи-незалежний спосіб виконання коду. Програмісти можуть писати код, не замислюючись, як і де він буде виконуватися.

### 4. Що таке JIT компілятор?

Це компілятор, який компілює код під час виконання програми, а не достроково. Звідси і назва JIT (Just-In-Time)

### 5. Що таке JDK?

JDK (Java Development Kit) – це комплект розробника додатків на мові Java, що включає компілятор, стандартні бібліотеки класів Java, приклади, документацію, різні утиліти і виконавчу систему JRE.

### 6. В чому полягає відмінність між JRE і JDK?

JRE не має компілятора Java, тому не може компілювати програми. JDK має компілятор javac і відповідає за компіляцію програм. JRE можна встановити як самостійну програму. JDK - це окремий інсталятор, який постачається в комплекті з JRE. JDK містить такі інструменти, як Javadoc та архіватор, які використовуються для розробки програм Java.

### 7. Які інтегровані середовища розробки програм мовою Java ви знаєте?

**NetBeans IDE** - безкоштовне інтегроване середовище розробки для всіх платформ Java - Java ME, Java SE і Java EE. Пропагувалося Sun Microsystems, як базовий засіб для розробки ПЗ на мові Java і інших мовах (C, C++, PHP, Fortran тощо).

**Eclipse IDE** – одне з найпопулярніших безкоштовних інтегрованих середовищ розробки для Java SE, Java EE і Java ME. Пропагується IBM, як базовий засіб для розробки ПЗ на мові Java і інших мовах (C, C++, Ruby, Fortran тощо).

**IntelliJ IDEA** - інтегроване середовище розробки для платформ Java SE, Java EE і Java ME. Розробник - компанія JetBrains. Розповсюджується в двох версіях: безкоштовної (Community Edition) та комерційної пропрієтарної (Ultimate Edition).

**JDeveloper** - інтегроване середовище розробки для платформ Java SE, Java EE і Java ME. Розробник - компанія Oracle.

**BlueJ** - інтегроване середовище розробки програм на мові Java, створене в основному для використання в навчанні, але також підходить для розробки невеликих програм.

## **8. Наведіть основні етапи написання та запуску програми написаної мовою Java.**

Переданий у JRE байт-код програми поступає у блок перевірки безпечності коду програми, який здійснює три головні перевірки:

- перевірка чи переходи відбуваються в допустимі місця програми;
- перевірка чи дані є ініціалізовані і посилання є типізовані;
- жорстка перевірка можливості доступу до приватних секцій класів і пакетів.

Перші дві перевірки переважно відбуваються при завантаженні програми в JRE, а остання – в процесі виконання програми, коли одні класи звертаються до полів інших класів. Після успішного завершення перевірки байт-коду JVM емулює (виконує) набір інструкцій, написаних для JVM, інтерпретуючи їх або використовуючи JIT компілятор. Сучасні JVM використовують JIT компілятор, який одночасно компілює частини байткоду, що мають аналогічну функціональність. Завдяки цьому зменшується кількість часу, що необхідна для компіляції порівняно з використанням інтерпретатора

**Висновок:** на даній лабораторній роботі я оволодів основами розробки програм мовою Java.