

## ЛАБОРАТОРНА РОБОТА 1

Тема: Робота з JDK і IDE IntelliJ IDEA.

Мета: ознайомитися з основами технології Java. Навчитися користуватися засобами JDK та IDE для створення java-програм.

### **Теоретичні питання**

1. Призначення і особливості мови Java.
2. Технологія Java і її складові. Робота з JDK без IDE.
3. Призначення і особливості IDE IntelliJ IDEA.

### **Контрольні питання**

1. Опишіть призначення складників технології Java: JVM, JRE, JDK, Java IDE.
2. Які переваги дає виконання Java-програм на JVM і чому?
3. Чи можна запустити java-програму, не маючи IDE і JDK? Як це зробити?
4. Чи можна компілювати java-програму, не маючи IDE? Як це зробити?
5. Опишіть основні особливості IDE IntelliJ IDEA та її переваги порівняно з іншими IDE.

### **Індивідуальні завдання**

#### **Загальне завдання**

1. Встановіть JDK (версію не нижче 1.8). Задайте шлях до компілятора в PATH. У звіті опишіть порядок своїх дій і надайте скріншоти для кожного етапу.

2. Скомпілюйте і виконайте просту програму з командного рядка, користуючись тільки засобами JDK. У звіті опишіть порядок своїх дій і наведіть скріншоти для кожного етапу.

```

import java.util.Scanner; // Підключаємо клас з бібліотеки

public class Joke { // Основний клас програми

    public static void main(String [] args) { // Головний метод

        Scanner scanner = new Scanner(System.in); // Створюємо об'єкт для читання
        System.out.println("Enter your first name"); // Виводимо текст
        String name = scanner.next(); // Вводимо ім'я
        String reverse = ""; // Змінна для збереження результату
        for (int i = name.length() - 2; i >= 0; i--) { // Міняємо порядок букв
            reverse += name.substring(i, i + 1);
        }
        // Робимо першу букву заголовною
        reverse = name.substring(name.length() - 1, name.length())
            .toUpperCase() + reverse.toLowerCase();
        System.out.println ("Then my name is" + reverse); // Виводимо результат
    }
}

```

3. Скомпілюйте і виконайте ту ж програму в IDE. У звіті опишіть порядок своїх дій зі створення та налаштування проекту і надайте скріншоти для кожного етапу.

4. Протестуйте на прикладі тієї ж програми роботу засобів доповнення та автогенерації коду, підказок, а також засобів налагодження (debugging) IDE. У звіті перерахуйте можливості, якими ви навчилися користуватися, і надайте їх скріншоти.

### ***Додаток А. Основні гарячі клавіші IDE***

IntelliJ IDEA має великий набір гарячих клавіш, які дозволяють виконувати велику частину роботи, не відриваючи рук від клавіатури, і в такий спосіб економити час розробника. Ті ж поєднання клавіш працюють і в інших продуктах JetBrains, як-от PyCharm і CLion. Нижче наведено перелік найбільш уживаних гарячих клавіш.

## ***Редагування***

Shift + Del (Ctrl + Y)	Видалити рядок (відмінність у тому, де потім залишиться курсор);
Ctrl + Del	Видалити від поточної позиції до кінця слова;
Ctrl + Backspace	Видалити від поточної позиції до початку слова;
Ctrl + D	Дублювати поточний рядок;
Tab / Shift + Tab	Збільшити / зменшити поточний відступ;
Ctrl + Alt + I	Вирівнювання відступів у коді;
Ctrl + Alt + L	Приведення коду у відповідність із code style;
Ctrl + /	Закоментувати / розкоментувати поточний рядок;
Ctrl + Shift + /	Закоментувати / розкоментувати виділений код;
Ctrl + - / +	Фолдінг: згорнути / розгорнути фрагмент коду.

## ***Вікна, вкладки***

Alt + вліво / вправо	Перейти на іншу вкладку редактора;
Ctrl + F4	Закрити вкладку;
Alt + цифра	Відкриття / закриття вікон Project, Structure, Changes і т. д.;
Ctrl + Tab	Перехід між вкладками та вікнами;
Shift + Esc	Закрити активне вікно;
Ctrl + коліщатко	Зміна розміру шрифту (налаштовується в меню File   Settings).

## ***Підказки та документація***

Ctrl + Q	Документація до елемента, на якому знаходиться курсор / виділення;
Ctrl + Shift + I	Показати реалізацію методу або класу;

Alt + Q	Відобразити ім'я класу або методу, у якому знаходиться курсор;
Ctrl + P	Підказка про очікувані аргументи методу;
Ctrl + F1	Показати опис помилки або попередження під курсором;
Alt + Enter	Показати пропоновані варіанти дій («лампочки»).

### ***Пошук***

Подвійний Shift	Швидкий пошук в усьому проекті (в тексті);
Ctrl + Shift + A	Швидкий пошук у налаштуваннях, діях і т. д.;
Alt + вниз / вгору	Перейти до наступного / попереднього методу;
Ctrl + [ та Ctrl + ]	Перехід на початок і кінець поточної області коду;
Ctrl + F	Пошук у поточному файлі;
Ctrl + Shift + F	Пошук в усіх файлах проекту (перехід – F4);
F3 / Shift + F3	Перейти до наступного / попереднього результату пошуку;
Ctrl + F12	Список методів із переходом до їх оголошення;
Ctrl + H	Ієрархія наслідування поточного класу і перехід за нею;
Ctrl + N	Пошук класу за ім'ям і перехід до нього;
Ctrl + Shift + N	Пошук файла за ім'ям і перехід до нього;
Ctrl + B	Перейти до оголошення змінної, класу або методу;
Ctrl + Alt + B	Перейти до реалізації класу або методу;
Ctrl + Shift + B	Визначити клас об'єкта і перейти до його реалізації;
Shift + Alt + 7	Знайти всі місця, де використовується метод або змінна;
Ctrl + Alt + 7	Аналогічно до попереднього пункту, але результат виводиться у спливаючому вікні.

## *Генерація коду і рефакторинг*

Ctrl + Space	Автодоповнення коду;
Ctrl + Shift + Space	Розумне автодоповнення;
Ctrl + O	Перевизначити метод батьківського класу;
Alt + Insert	Автогенерація коду (відобразити варіанти);
Shift + F6	Перейменування об'єкта під курсором (змінної, класу, методу);
Alt + Delete	Безпечне видалення класу, методу або атрибута.

## *Інше*

Ctrl + ~	Швидке перемикання колірної схеми, code style і т. д.;
Alt + F12	Відкрити / закрити термінал;
F10	Запустити останню запущену програму або тест;
Shift + F10	Запустити відкриту в редакторі програму або тест;
Shift + Alt + F10	Список програм / тестів, які нещодавно запускалися;
Ctrl + F2	Зупинити запущену програму.