

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет “Львівська політехніка”
Інститут комп’ютерних наук та інформаційних технологій

Кафедра САП



Звіт
до лабораторної роботи №2
з дисципліни: “Управління ІТ-проектами”
на тему:
“АВТОМАТИЗАЦІЯ ЗБІРКИ ПРОЕКТУ”

Виконала студентка групи КН-408

КИЧУК О. О.

Прийняв асистент
ШПАКОВИЧ І. М.

ЛВІВ – 2023

Мета роботи: Ознайомитись з системами автоматизованих збірок проектів. Зрозуміти принципи їх роботи. Набути практичних навичок автоматизації збірки проекту на прикладі інструментів Maven та Gradle.

Хід роботи:

Написала елементарний код на Java для використання в Gradle:

```
package lab_2_gradle;
package gradle_lab;

public class App {
    public static void main(String[] args) {
        double number1 = 15; // Перше число
        double number2 = 3;  // Друге число

        double sum = add(number1, number2);
        double difference = subtract(number1, number2);
        double product = multiply(number1, number2);
        double quotient = divide(number1, number2);

        System.out.println("Summary: " + sum);
        System.out.println("Difference: " + difference);
        System.out.println("Multiply: " + product);
        System.out.println("Division: " + quotient);
    }

    public static double add(double a, double b) {
        return a + b;
    }

    public static double subtract(double a, double b) {
        return a - b;
    }

    public static double multiply(double a, double b) {
        return a * b;
    }

    public static double divide(double a, double b) {
        if (b == 0) {
            System.out.println("Error: Division by zero!");
            return Double.NaN; // Повертає "не число" у разі помилки
        }
        return a / b;
    }
}
```

Скачала та встановила Gradle, запустила команду *gradle init* для ініціалізації проекту в обраному місці:

```
o\hak001ia MINGW64 /d/універ/7 семестр/Y11N/lab_2_gradle
$ gradle init

Select type of project to generate:
 1: basic
 2: application
 3: library
 4: Gradle plugin
Enter selection (default: basic) [1..4] 2

Select implementation language:
 1: C++
 2: Groovy
 3: Java
 4: Kotlin
 5: Scala
 6: Swift
Enter selection (default: Java) [1..6] 3

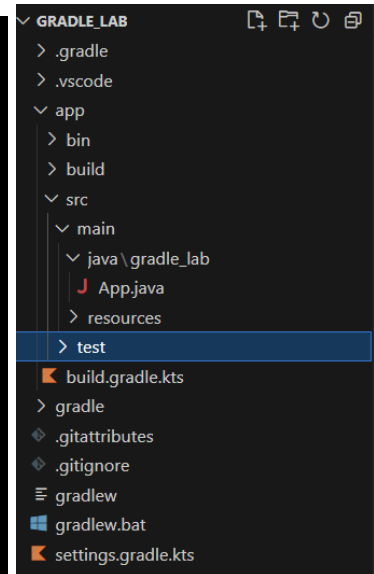
Generate multiple subprojects for application? (default: no) [yes, no]
Select build script DSL:
 1: Kotlin
 2: Groovy
Enter selection (default: Kotlin) [1..2] 1

Select test framework:
 1: JUnit 4
 2: TestNG
 3: Spock
 4: JUnit Jupiter
Enter selection (default: JUnit Jupiter) [1..4] 4

Project name (default: lab_2_gradle):
Source package (default: lab_2_gradle):
Enter target version of Java (min. 7) (default: 11):
Generate build using new APIs and behavior (some features may change in the next minor release)? (default: no) [yes, no]

> Task :init
To learn more about Gradle by exploring our Samples at https://docs.gradle.org/8.4/samples/sample_building_java_applications.html

BUILD SUCCESSFUL in 1m
2 actionable tasks: 2 executed
```



settings.gradle.kts:

```
plugins {
    // Apply the foojay-resolver plugin to allow automatic download of JDKs
    id("org.gradle.toolchains.foojay-resolver-convention") version "0.7.0"
}

rootProject.name = "gradle_lab"
include("app")
```

Gradle читає це і розуміє, який проект буде включений в білд і які плагіни є потрібними.

build.gradle.kts:

```
plugins {
    // Apply the application plugin to add support for building a CLI
    application in Java.
    application
}

repositories {
    // Use Maven Central for resolving dependencies.
    mavenCentral()
}

dependencies {
    // Use JUnit Jupiter for testing.
    testImplementation("org.junit.jupiter:junit-jupiter:5.9.3")
}
```

```

testRuntimeOnly("org.junit.platform:junit-platform-launcher")

// This dependency is used by the application.
implementation("com.google.guava:guava:32.1.1-jre")
}

// Apply a specific Java toolchain to ease working on different environments.
java {
    toolchain {
        languageVersion.set(JavaLanguageVersion.of(11))
    }
}

application {
    // Define the main class for the application.
    mainClass.set("gradle_lab.App")
}

tasks.named<Test>("test") {
    // Use JUnit Platform for unit tests.
    useJUnitPlatform()
}

```

Gradle розуміє, що саме треба і робити для білда, які залежності і плагіни потрібні, визначення головного класу, та запуск тестів з JUnit Platform.

Результат:

```

PS D:\gradle_lab> ./gradlew build

BUILD SUCCESSFUL in 1s
6 actionable tasks: 5 executed, 1 up-to-date
PS D:\gradle_lab> ./gradlew run

> Task :app:run
Summary: 18.0
Difference: 12.0
Multiply: 45.0
Division: 5.0

BUILD SUCCESSFUL in 1s
2 actionable tasks: 1 executed, 1 up-to-date
PS D:\gradle_lab> █

```

Висновки: Ознайомилася з системами автоматизованих збірок проектів. Зрозуміла принципи їх роботи. Набула практичних навичок автоматизації збірки проекту на прикладі інструмента Gradle.