DevOps Final Project

Maven-Based Body Mass Index (BMI) Calculator with GitHub Actions CI/CD

Anush Kar

22BBS0246

Project Repository: https://github.com/anush-kar/bmiCalculator.git

---

Objective

The purpose of this project is to develop a Java-based Body Mass Index (BMI) Calculator using the Maven build automation tool. The project also integrates version control using Git and GitHub, and sets up Continuous Integration (CI) using GitHub Actions. The calculator includes enhancements like BMR (Basal Metabolic Rate) and Maintenance Calories calculation based on user input such as age, sex, and activity level.

Step-by-Step Implementation

Step 1: Setup Maven Project in Eclipse

1. Open Eclipse IDE.
2. Create a new Maven project.
3. Choose Maven Archetype:
    - **Group Id:** org.apache.maven.archetypes
    - **Artifact Id:** maven-archetype-quickstart
4. Fill in the following project identifiers:
    - **Group ID:** com.bmi.calculator
    - **Artifact ID:** bmiCalculator
5. Eclipse will create a project structure with directories:
    - src/main/java/
    - src/test/java/
    - pom.xml (Maven configuration file)

Step 2: Implement Java Class

Navigate to:
src/main/java/com/bmi/calculator/bmiCalculator/

Create a new Java class: **BMICalculator.java**

```
1. package com.bmi.calculator.bmiCalculator;
2. import java.util.Scanner;
3.
4. public class BMICalculator {
5.     public static void main(String[] args) {
6.         Scanner input = new Scanner(System.in);
7.
8.         System.out.print("Enter weight in kilograms: ");
9.         double weight = input.nextDouble();
10.
11.             System.out.print("Enter height in meters: ");
12.             double height = input.nextDouble();
13.
14.             double bmi = weight / (height * height);
15.             System.out.printf("Your BMI is: %.2f\n", bmi);
16.
17.             if (bmi < 18.5)
18.                 System.out.println("You are underweight.");
19.             else if (bmi < 25)
20.                 System.out.println("You have a normal weight.");
21.             else if (bmi < 30)
22.                 System.out.println("You are overweight.");
23.             else
24.                 System.out.println("You are obese.");
25.
26.             input.close();
27.         }
28.     }
```

Step 3: Create Unit Tests

Create test class: **BMICalculatorTest.java**
Location: src/test/java/com/bmi/calculator/bmiCalculator/

```
1.  package com.bmi.calculator.bmiCalculator;
2.
3. import static org.junit.jupiter.api.Assertions.assertEquals;
4. import org.junit.jupiter.api.Test;
5.
6. public class BMICalculatorTest {
7.     @Test
8.     public void testBMI() {
9.         double weight = 70;
10.             double height = 1.75;
11.             double expected = weight / (height * height);
12.             assertEquals(expected, 22.8571428571, 0.01);
13.         }
14.     }
```

Note: Ensure the correct JUnit version is added to pom.xml.

Step 4: Add Dependencies to `pom.xml`

Example dependencies:

BMICalculator.java | bmiCalculator/pom.xml | BMICalculatorTest.java

```
    http://maven.apache.org/xsd/maven-4.0.0.xsd (xsi:schemaLocation with catalog)
 1  <?xml version="1.0" encoding="UTF-8"?>
 2  <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSche
 3    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.
 4    <modelVersion>4.0.0</modelVersion>
 5
 6    <groupId>com.bmi.calculator  </groupId>
 7    <artifactId>bmiCalculator</artifactId>
 8    <version>0.0.1-SNAPSHOT</version>
 9
10    <name>bmiCalculator</name>
11    <!-- FIXME change it to the project's website -->
12    <url>http://www.example.com</url>
13
14    <properties>
15      <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
16      <maven.compiler.release>17</maven.compiler.release>
17    </properties>
18
19    <dependencyManagement>
20      <dependencies>
21        <dependency>
22          <groupId>org.junit</groupId>
23          <artifactId>junit-bom</artifactId>
24          <version>5.11.0</version>
25          <type>pom</type>
26          <scope>import</scope>
27        </dependency>
28      </dependencies>
29    </dependencyManagement>
30
31    <dependencies>
32      <dependency>
33        <groupId>org.junit.jupiter</groupId>
34        <artifactId>junit-jupiter</artifactId>
35        <version>5.10.0</version>
36        <scope>test</scope>
37      </dependency>
38      <!-- Optionally: parameterized tests support -->
```

Step 5: Compile & Install Maven Project

Use Maven commands:

```
[INFO] ------------------< com.bmi.calculator:bmiCalculator >------------------
[INFO] Building bmiCalculator 0.0.1-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- resources:3.3.1:resources (default-resources) @ bmiCalculator ---
[INFO] skip non existing resourceDirectory /home/anush/eclipse-workspace/bmiCalculator/src/main/resources
[INFO]
[INFO] --- compiler:3.13.0:compile (default-compile) @ bmiCalculator ---
[INFO] Recompiling the module because of changed source code.
[INFO] Compiling 1 source file with javac [debug release 17] to target/classes
[INFO]
[INFO] --- resources:3.3.1:testResources (default-testResources) @ bmiCalculator ---
[INFO] skip non existing resourceDirectory /home/anush/eclipse-workspace/bmiCalculator/src/test/resources
[INFO]
[INFO] --- compiler:3.13.0:testCompile (default-testCompile) @ bmiCalculator ---
[INFO] Recompiling the module because of changed dependency.
[INFO] Compiling 1 source file with javac [debug release 17] to target/test-classes
[INFO]
[INFO] --- surefire:3.3.0:test (default-test) @ bmiCalculator ---
[INFO] Using auto detected provider org.apache.maven.surefire.junitplatform.JUnitPlatformProvider
[INFO]
[INFO] -------------------------------------------------------
[INFO]  T E S T S
[INFO] -------------------------------------------------------
[INFO] Running com.bmi.calculator.bmiCalculator.BMICalculatorTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 1.238 s -- in com.bmi.calculator.bmiCalculator.BMICalculatorTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
```

Running `mvn install`

```
[INFO]
[INFO] --- install:3.1.2:install (default-install) @ bmiCalculator ---
[INFO] Installing /home/anush/eclipse-workspace/bmiCalculator/pom.xml to /home/anush/.m2/repository/com/bmi/calculator/bmiCalculator/0.0.1-SNAPSHOT/bmiCalculator-0.0.1-SNAPSHOT.pom
[INFO] Installing /home/anush/eclipse-workspace/bmiCalculator/target/bmiCalculator-0.0.1-SNAPSHOT.jar to /home/anush/.m2/repository/com/bmi/calculator/bmiCalculator/0.0.1-SNAPSHOT/bmiCalculator-0.0.1-SNAPSHOT.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  31.203 s
[INFO] Finished at: 2025-06-30T12:14:25+05:30
[INFO] ------------------------------------------------------------------------
```

```
anush@anush-VirtualBox: ~/eclipse-workspace/bmiCalculator/target
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator/target$
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator/target$ ls
bmiCalculator-0.0.1-SNAPSHOT.jar  generated-test-sources  surefire-reports
classes                           maven-archiver          test-classes
generated-sources                 maven-status
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator/target$ java -cp bmiCal
culator-0.0.1-SNAPSHOT.jar com.bmi.calculator.bmiCalculator.BMICalculator
Enter weight in kilograms: 70
Enter height in meters: 1.8
Your BMI is: 21.60
You have a normal weight.
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator/target$ java -cp bmiCal
culator-0.0.1-SNAPSHOT.jar com.bmi.calculator.bmiCalculator.BMICalculator
Enter weight in kilograms: 90
Enter height in meters: 1.76
Your BMI is: 29.05
You are overweight.
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator/target$ java -cp bmiCal
culator-0.0.1-SNAPSHOT.jar com.bmi.calculator.bmiCalculator.BMICalculator
Enter weight in kilograms: 58
Enter height in meters: 1.82
Your BMI is: 17.51
You are underweight.
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator/target$
```

Step 5

Setting up git in the ubuntu VM

```
Initialized empty Git repository in /home/anush/eclipse-workspace/bmiCalculator/.git/
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git remote add origin https://github.com/anush-kar/bmiCalculator.git
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git add .
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git commit -m "Initial commit - BMI calculator"
Author identity unknown

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'anush@anush-VirtualBox.(none)')
```

```
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git config --global user.email "anushk04@gmail.com"
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git commit -m "Initial commit - BMI calculator"
[master (root-commit) 6422942] Initial commit - BMI calculator
 20 files changed, 281 insertions(+)
 create mode 100644 .classpath
 create mode 100644 .mvn/jvm.config
 create mode 100644 .mvn/maven.config
 create mode 100644 .project
 create mode 100644 .settings/org.eclipse.core.resources.prefs
 create mode 100644 .settings/org.eclipse.jdt.core.prefs
 create mode 100644 .settings/org.eclipse.m2e.core.prefs
 create mode 100644 pom.xml
 create mode 100644 src/main/java/com/bmi/calculator/bmiCalculator/BMICalculator.java
 create mode 100644 src/test/java/com/bmi/calculator/bmiCalculator/BMICalculatorTest.java
 create mode 100644 target/bmiCalculator-0.0.1-SNAPSHOT.jar
 create mode 100644 target/classes/com/bmi/calculator/bmiCalculator/BMICalculator.class
 create mode 100644 target/maven-archiver/pom.properties
 create mode 100644 target/maven-status/maven-compiler-plugin/compile/default-compile/createdFiles.lst
 create mode 100644 target/maven-status/maven-compiler-plugin/compile/default-compile/inputFiles.lst
 create mode 100644 target/maven-status/maven-compiler-plugin/testCompile/default-testCompile/createdFiles.lst
 create mode 100644 target/maven-status/maven-compiler-plugin/testCompile/default-testCompile/inputFiles.lst
 create mode 100644 target/surefire-reports/TEST-com.bmi.calculator.bmiCalculator.BMICalculatorTest.xml
 create mode 100644 target/surefire-reports/com.bmi.calculator.bmiCalculator.BMICalculatorTest.txt
 create mode 100644 target/test-classes/com/bmi/calculator/bmiCalculator/BMICalculatorTest.class
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git branch -M main
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git push -u origin main
```

This will compile the project and generate a `.jar` file in the `/target` folder.

Step 6: Git Setup and GitHub Push

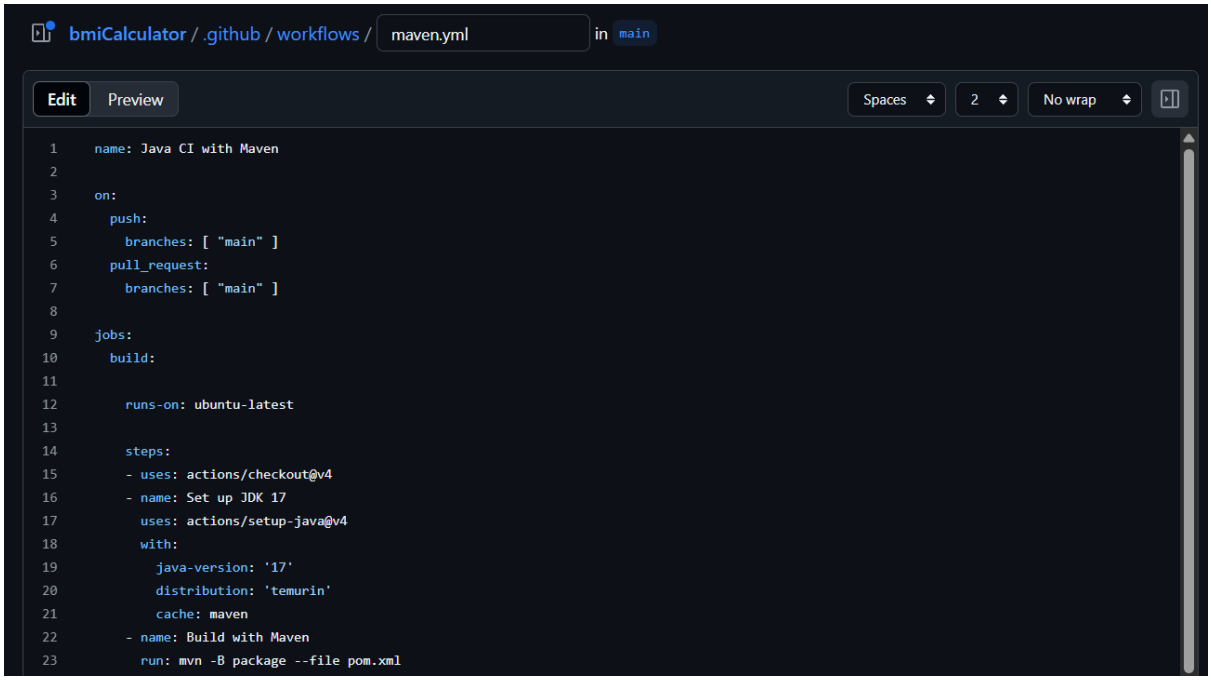
1. Initialize git in your project directory:
2. Add remote origin and push to GitHub:

```
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git push -u origin main
Username for 'https://github.com': anush-kar
Password for 'https://anush-kar@github.com':
Enumerating objects: 55, done.
Counting objects: 100% (55/55), done.
Delta compression using up to 6 threads
Compressing objects: 100% (30/30), done.
Writing objects: 100% (55/55), 10.06 KiB | 44.00 KiB/s, done.
Total 55 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/anush-kar/bmiCalculator.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$
```

Step 7: GitHub Actions CI Workflow

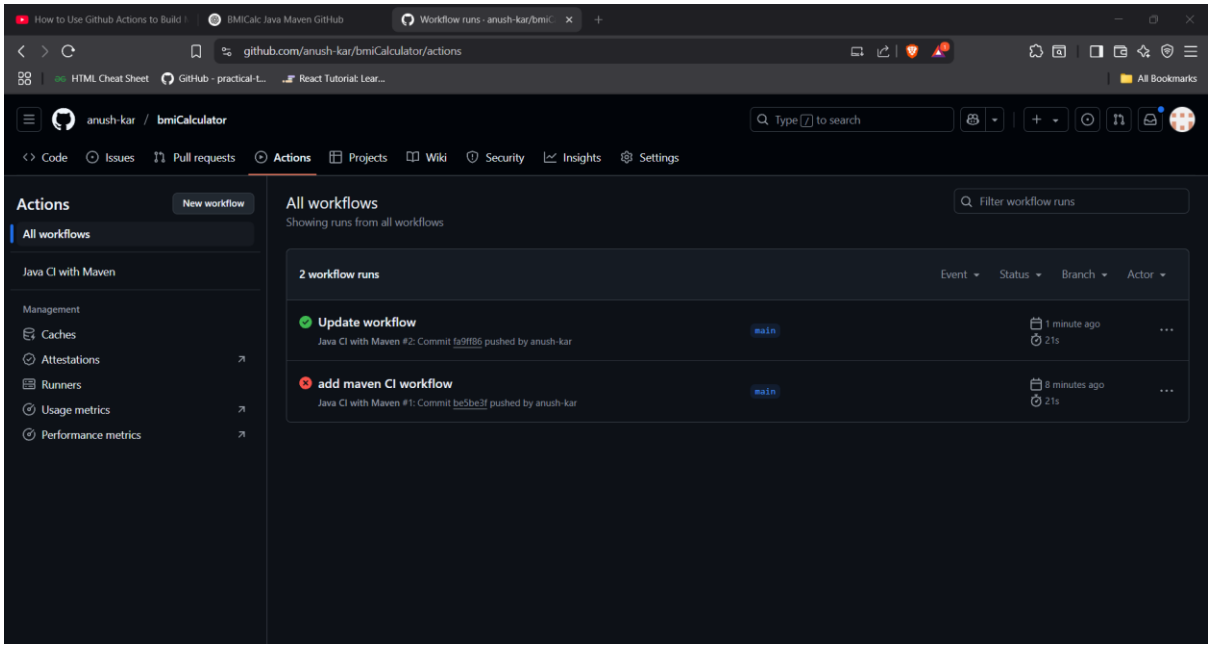
Create GitHub Actions workflow in `.github/workflows/maven.yml`:

Workflow file



This workflow triggers on every push or pull request to the `main` branch, automatically building and testing the project.

Automated build



Step 8: Feature Enhancements

Modified application to include:

- **Inputs**:
    - Weight (kg)
    - Height (m)
    - Age (years)
    - Sex (male/female)
    - Activity Level (Sedentary, Light, Moderate, Active, Very Active)
- **Calculations**:
    - BMI = weight / (height × height)
    - BMR (Basal Metabolic Rate):
        - Males: `BMR = 10×weight + 6.25×height(cm) - 5×age + 5`
        - Females: `BMR = 10×weight + 6.25×height(cm) - 5×age - 161`
    - Maintenance Calories = BMR × Activity Factor
        - Activity Multipliers:
            - Sedentary = 1.2
            - Light = 1.375
            - Moderate = 1.55
            - Active = 1.725
            - Very Active = 1.9

Outputs both BMI category and Maintenance Calorie Needs.

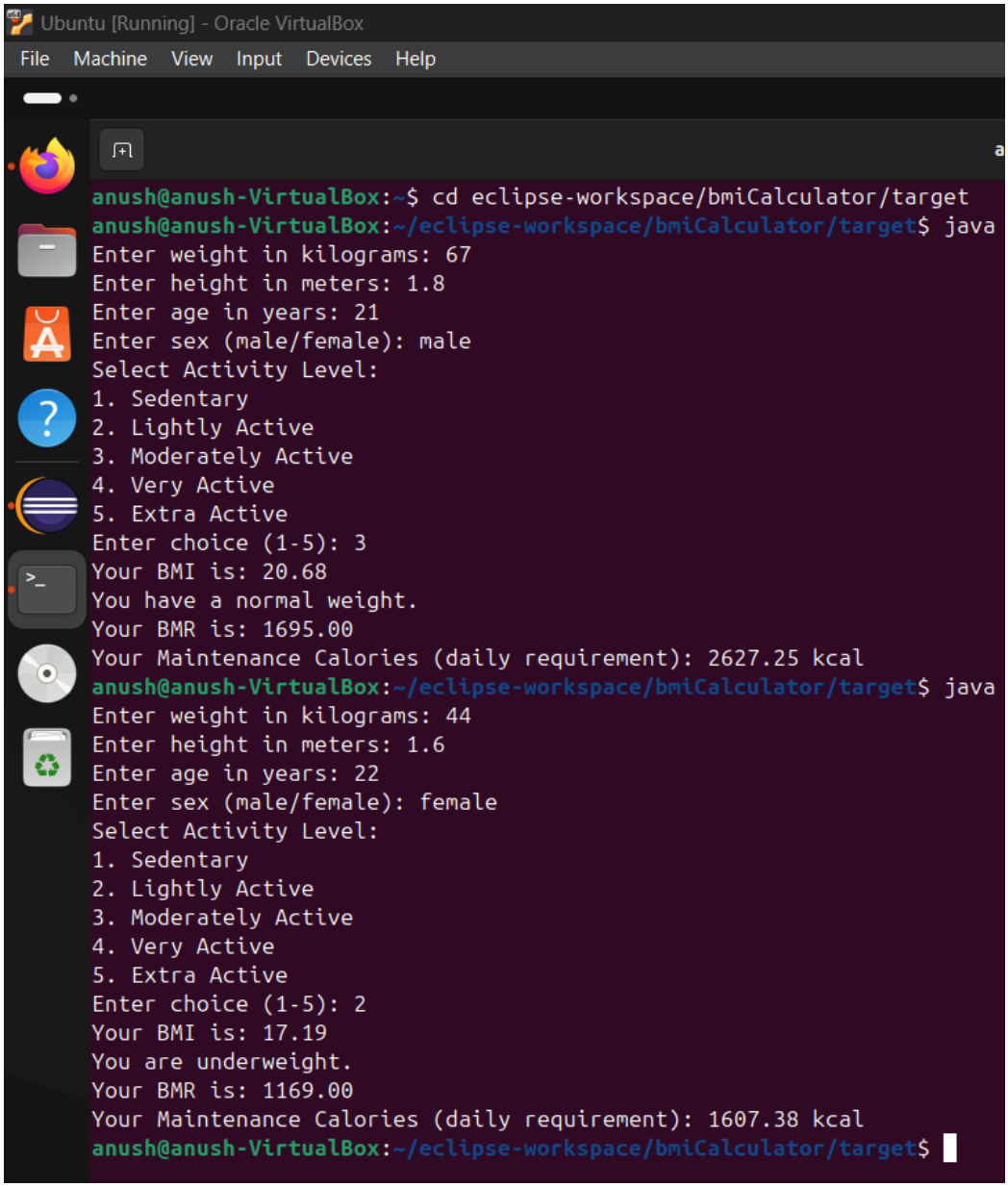
Step 9: Rebuilding and Testing

`mvn clean`

`mvn install`

Test the app locally

`java -cp bmiCalculator-1.0-SNAPSHOT.jar com.bmi.calculator.bmiCalculator.BMICalculator`

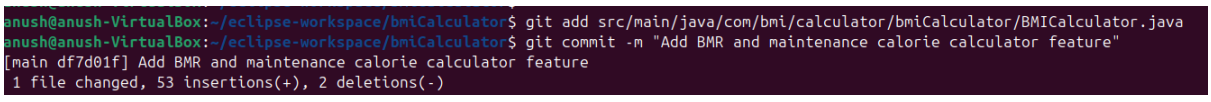


Step 10: Git Workflow Updates

To reflect changes in GitHub Actions:

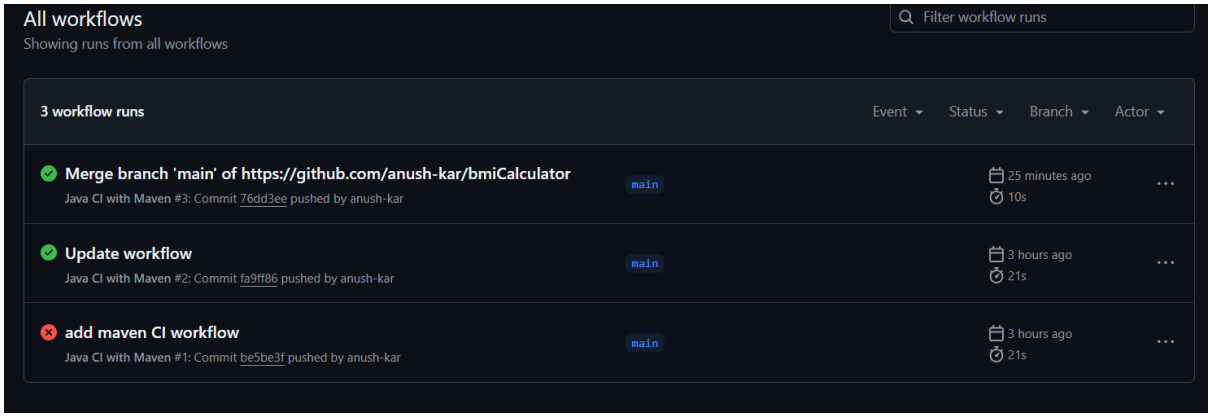
`git pull` to merge the branch with changes made to workflow file on github.

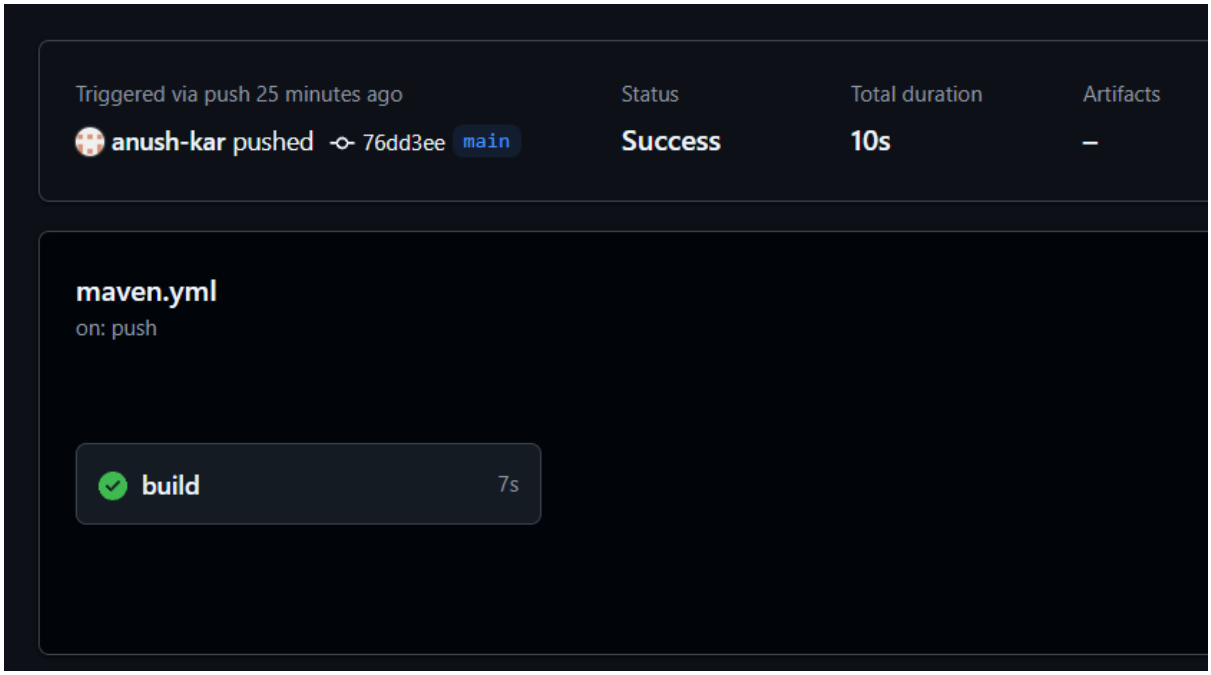
Add, Commit & Push the changes to github

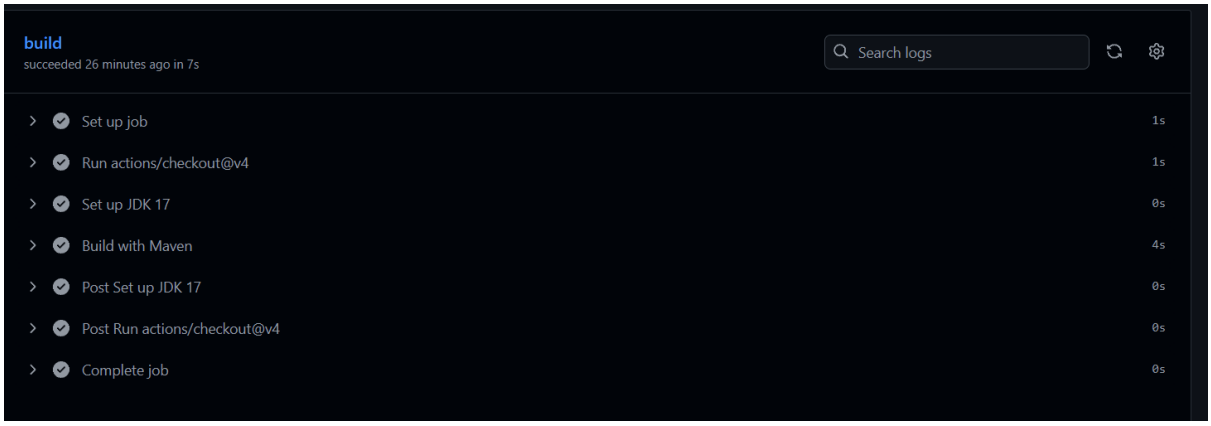


```
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git config pull.rebase false
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git pull
Merge made by the 'ort' strategy.
 .github/workflows/maven.yml | 27 ++++++++++++++++++++++++++
 1 file changed, 27 insertions(+)
 create mode 100644 .github/workflows/maven.yml
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$ git push
Username for 'https://github.com': anush-kar
Password for 'https://anush-kar@github.com':
Enumerating objects: 30, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 6 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (12/12), 1.63 KiB | 833.00 KiB/s, done.
Total 12 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/anush-kar/bmiCalculator.git
   fa9ff86..76dd3ee  main -> main
anush@anush-VirtualBox:~/eclipse-workspace/bmiCalculator$
```

Changes reflected on github actions workflows







Conclusion

This project demonstrates the full lifecycle of a DevOps-integrated Java application:

- Code development with Maven and JUnit
- Version control using Git and GitHub
- Automation via GitHub Actions CI
- Enhancement of application features using user input and logic
- Local and remote testing of builds and workflows