Build app on multiple environments

Github actions provides a method to run your tests with various input. This can be useful if you want to make sure that the code builds on different versions of, e.g., Java or Node. To avoid repeating the same thing in YAML, you can use strategy

The app in this repository is Java-based and it is running on Java 11. The example how we can use it for different versions of Node:

```
jobs:
  build:
    runs-on: ubuntu-latest
    strategy:
      matrix:
        node: [6, 8, 10]
    steps:
      - uses: actions/setup-node@v2
        with:
          node-version: ${{ matrix.node }}
```

Here, job will run three seperate jobs for different versions of Node 6, 8 and 10. There can be similar solution for container based pipelines:

```
jobs:
  job:
    runs-on: ubuntu-latest
    strategy:
      matrix:
        container: ["ubuntu:bionic", "fedora:31", "opensuse/leap:42.3", "centos8"]
    container:
      image: ${{ matrix.container }}
    steps:
      - name: checkout
        uses: actions/checkout@v1
```

In this example pipeline runs-on ubuntu-latest for 4 different containers: "ubuntu:bionic", "fedora:31", "opensuse/leap:42.3", "centos8". This way we can skip repeating the workflows.

Tasks

we would like to build our application on different versios of Java.

Add a new job that builds on various versions

- You can add a new file on .github/workflows/matrix.yml, which will only include Clone-down and Build job.

▶ Build job

```
name: Java CI
on: push
jobs:
  Clone-down:
    name: Clone down repo
    runs-on: ubuntu-latest
    container: gradle:6-jdk11
    steps:
    - uses: actions/checkout@v2
    - name: Upload Repo
      uses: actions/upload-artifact@v2
      with:
        name: code
        path: .
  Build:
    runs-on: ubuntu-latest
    needs: Clone-down
    container: gradle:6-jdk11
    steps:
    - name: Download code
      uses: actions/download-artifact@v2
      with:
        name: code
        path: .
    - name: Build with Gradle
      run: chmod +x ci/build-app.sh && ci/build-app.sh
    - name: Upload Repo
      uses: actions/upload-artifact@v2
      with:
        name: code
        path: .
```

- Edit build job to run for different versions of Java. Add matrix for types of containers as: `["gradle:6-jdk8", "gradle:6-jdk11", "gradle:6-jdk17"]` to your build job.

```
strategy:
  matrix:
    container: ["gradle:6-jdk8", "gradle:6-jdk11", "gradle:6-jdk17"]
```

- Remember to edit container name:

```
    container:
      image: ${{ matrix.container }}
```

▶ Solution

```
name: Java CI
on: push
jobs:
  Clone-down:
    name: Clone down repo
    runs-on: ubuntu-latest
    container: gradle:6-jdk11
    steps:
    - uses: actions/checkout@v2
    - name: Upload Repo
      uses: actions/upload-artifact@v2
      with:
        name: code
        path: .
  Build:
    runs-on: ubuntu-latest
    needs: Clone-down
    strategy:
      matrix:
        container: ["gradle:6-jdk8", "gradle:6-jdk11", "gradle:6-jdk17"]
    container:
      image: ${{ matrix.container }}
    steps:
    - name: Download code
      uses: actions/download-artifact@v2
      with:
        name: code
        path: .
    - name: Build with Gradle
      run: chmod +x ci/build-app.sh && ci/build-app.sh
    - name: Upload Repo
      uses: actions/upload-artifact@v2
      with:
        name: code
        path: .
```

Results

You should have 3 jobs under `Build` job for different version of Java.

Questions:

- Are there versions of Java which do not work?