Storing artifacts

It is possible to store artifacts in Github Actions. The `artifact` is the result of the build, in this case the compiled code.

> This should not be mistaken for proper artifact management, but it is useful for making the artifacts built by the pipeline available.

To deal with artifacts, a `Github Actions Action` can be used, which can be found on Github Marketplace.

To upload artifacts use the following syntax with `actions/upload-artifact@v2` Link to documentation:

```
- name: Upload a Build Artifact
  uses: actions/upload-artifact@v2
  with:
    name: my-artifact
    path: path/to/artifact/
```

As artifacts can be uploaded it can also be downloaded from Github Actions with help of `actions/download-artifact@v2` as:

```
- name: Download a single artifact
  uses: actions/download-artifact@v2
  with:
    name: my-artifact
    path: path/to/download/artifact/
```

Link to documentation

This information will be needed in next exercises.

💡

▶ More information about storing artifacts

Tasks

We want to add a step that builds all of the code, and then makes the compiled code available for other steps to use.

In order to achieve this we will simply save the state of the entire repository after running the build script.

- Add step named `Upload Repo` to the existing job, which will upload an artifact with the name `code`, with the path `.` to use the current directory.

```
- name: Upload Repo
  uses: actions/upload-artifact@v2
```

```
      with:
        name: code
        path: .
```

Solution

If you strugle and need to see the whole ***Solution*** you can extend the section below.

▶ Solution

```
on: push
jobs:
Build:
  runs-on: ubuntu-latest
  container: gradle:6-jdk11
  steps:
    - name: Clone-down
      uses: actions/checkout@v2
    - name: Build application
      run: chmod +x ci/build-app.sh && ci/build-app.sh
    - name: Test
      run: chmod +x ci/unit-test-app.sh && ci/unit-test-app.sh
    - name: Upload Repo
      uses: actions/upload-artifact@v2
      with:
        name: code
        path: .
```

Results

If all works out fine, your newest build should show something like, where you can find your uploaded artifact:

Resources

https://docs.github.com/en/actions/guides/storing-workflow-data-as-artifacts