

Exercise: CI/CD in GitHub Actions

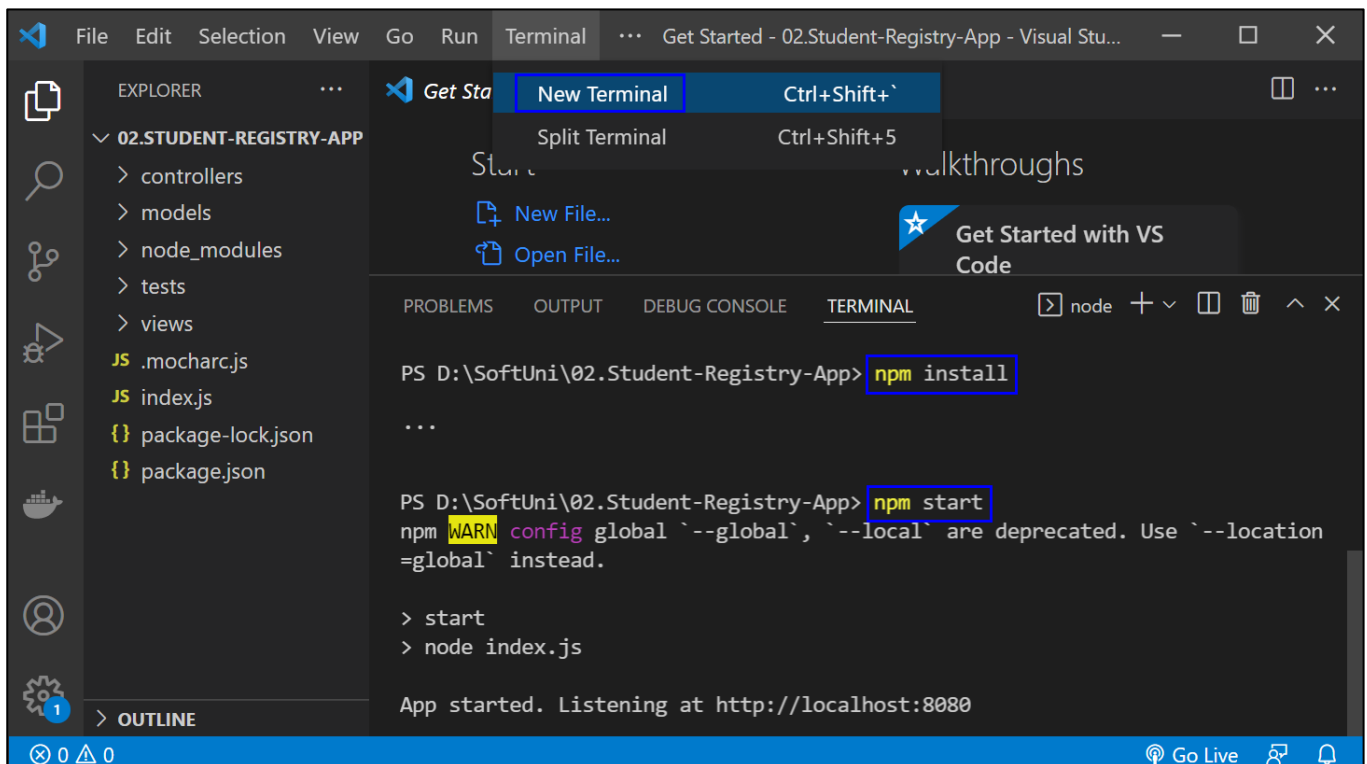
Problems for exercises for the "[Software Engineering and DevOps](#)" course @ SoftUni.

1. CI Workflow – "Student Registry" App

Step 1: Run the App Locally

We have the "Student Registry" Node.js app in the **resources**. Your task is to **create a CI workflow in GitHub Actions** to **start and test the app** on three different Node.js versions:

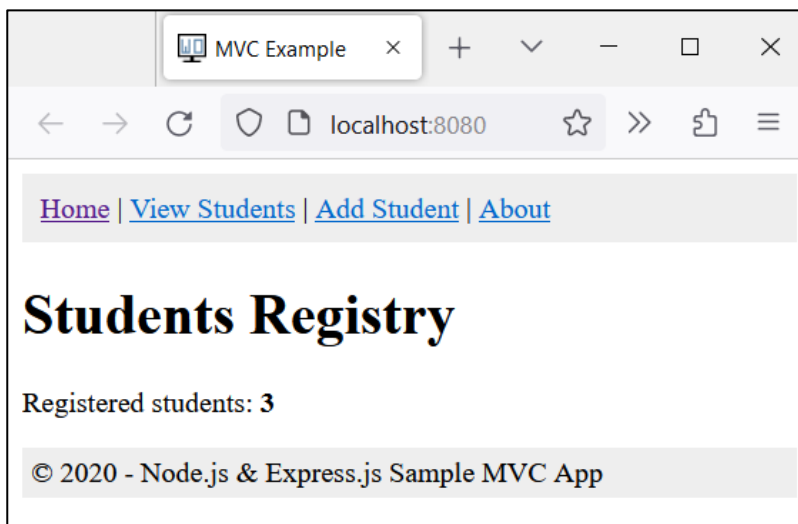
Let's first **start the app locally** in **Visual Studio Code**. To do this, you should **open the project**, open a **new terminal** from [Terminal] → [New Terminal] and **execute the "npm install" and "npm start" commands**:



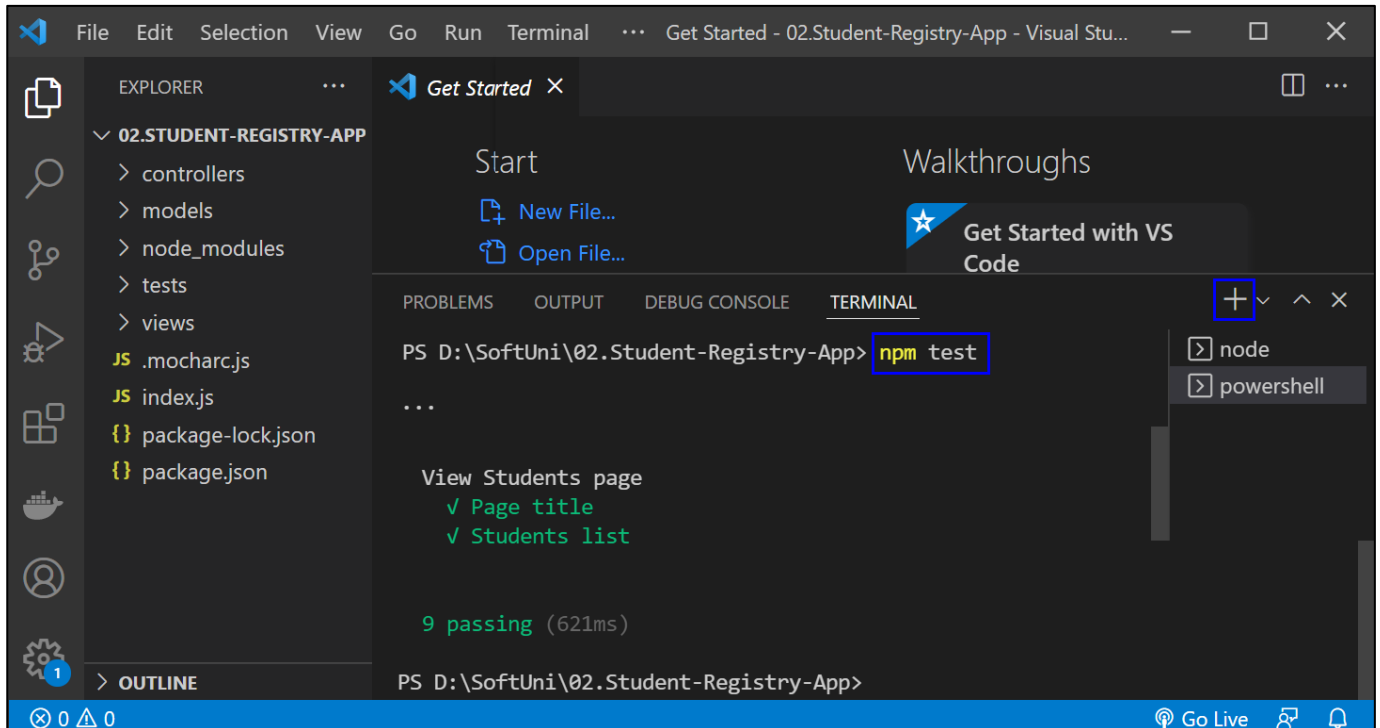
The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays the project structure for '02.STUDENT-REGISTRY-APP', including folders for controllers, models, node_modules, tests, and views, and files for .mocharc.js, index.js, package-lock.json, and package.json. The Terminal panel at the bottom shows the execution of 'npm install' and 'npm start' commands. The output of 'npm start' includes a warning about deprecated flags and confirms the app is listening at http://localhost:8080.

```
PS D:\SoftUni\02.Student-Registry-App> npm install
...
PS D:\SoftUni\02.Student-Registry-App> npm start
npm WARN config global `--global`, `--local` are deprecated. Use `--location`
=global` instead.
> start
> node index.js
App started. Listening at http://localhost:8080
```

The "npm install" command installs app dependencies from the **package.json** file and "npm start" starts the app. You can look at the app on <http://localhost:8080>:



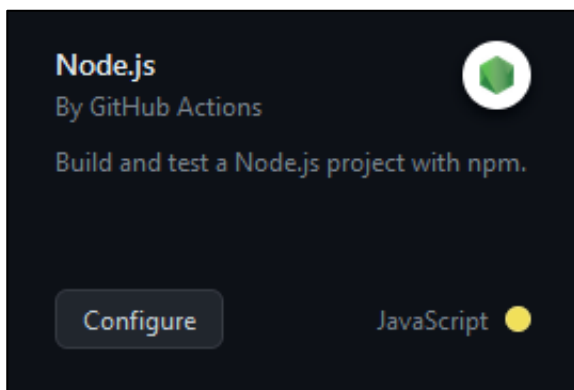
Then, you can **return to Visual Studio Code**, open a **new terminal** with **[+]** and run **"npm test"** to run the **app tests**. They should be **successful**:



NOTE: if the **app was not started**, **tests would fail** because these are integration tests and are executed on the running app.

Step 2: Create and Run Workflow

Now you should **upload the app code to GitHub** and **create a GitHub Actions CI workflow** to **start and test the app**. You can use the **following template**:



Before you commit the **generated YAML workflow file**, you should:

- **Change the YAML file name** to something more meaningful
- **Examine the workflow**, the **job** you have and its **steps**
- **Run the job** on the **last Node.js versions: 18.x**
- **Change the workflow name**
- **Modify workflow job steps**: you should **use the three commands** which we used above to **start and test the app**, not the ones you have in the generated YAML file or **your workflow won't be successful**
- **Add names for each step** in your workflow job

Finally, **run the workflow job** and make sure that **it is successful**:

build (16.x)
succeeded now in 22s

Q Search logs

⚙

> Set up job

> Checkout repo

> Use Node.js 16.x

> Install dependencies

> Start app

> Run tests

> Post Use Node.js 16.x

> Post Checkout repo

> Complete job

2s

1s

2s

9s

5s

1s

0s

0s

0s

2. CD Workflow – "Student Registry" App

Now, let's **create a CD workflow** for the **"Student Registry" Node.js app** to **deploy it to Render.com**.

We will continue working on the file that we created for the **CI** workflow.

To do this, you should **fulfill the following steps**:

- Create a free **Render.com** account
- **Generate an API Token**:
 - Navigate to the **"API Keys"** section in your **Render.com** Account settings;
 - Generate an API token by clicking on **"Create API Key"**;
 - Give it a meaningful name (e.g., **"GitHub Actions Token"**);
 - Click on **"Create Token"** to generate it.
- Add a new **Web Service**:
 - Connect your **GitHub account** to the service;
 - Connect your **GitHub repository** holding the application;
 - Give your **service a unique and meaningful name**;
- Add **Render Service ID as a GitHub Secret**:
 - Go to the **Settings** menu of your web service in **Render.com** and find the **Deploy Hook**;
 - **Copy the value that matches the pattern from the red square**:

Deploy Hook
Your private URL to trigger a deploy for this server. Remember to keep this a secret.

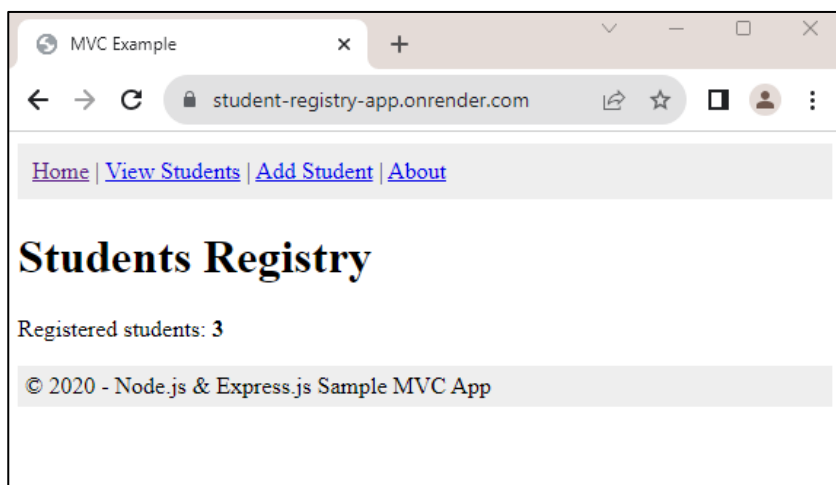
https://api.render.com/deploy/srv-civrkf6nqq148o2bm9j0?key=hIJFE0hhWYE

Ⓞ

Regenerate Hook

- Go to your GitHub repository, click on "**Settings**," then select "**Secrets and variables**" from the left sidebar;
- Click on "**Actions**" and then click on "**New repository secret**" and add a new secret with the following details:
 - Name: **SERVICE_ID**
 - Value: The service id that you copied from Render.com
- Click "**Add secret**" to save it.
- **Add Render.com API Token as a GitHub Secret:**
 - Go to your GitHub repository, click on "**Settings**," then select "**Secrets and variables**" from the left sidebar;
 - Click on "**Actions**" and then click on "**New repository secret**" and add a new secret with the following details:
 - Name: **RENDER_TOKEN**
 - Value: The API token you generated on Render.com
 - Click "**Add secret**" to save it.
- **Create and define the CD workflow:**
 - Set the **job** to be **dependent** of the **test** job from the **CI workflow**
 - In the **YAML** file that we used for the CI workflow, use the **custom** GitHub action [johnbeynon/render-deploy-action@v0.0.8](https://github.com/johnbeynon/render-deploy-action@v0.0.8) to deploy the application to Render;
 - Use the Render service ID and API key, which are stored as secrets in the repository.

GitHub Actions will execute the CD workflow, which involves installing Node.js, installing dependencies, and deploying the app to **Render.com**. The workflow will log in to **Render.com** using the API token you provided as a secret and then deploy your app.



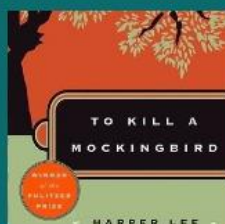
3. CI/CD Workflow – "Library Catalog" App

We have the "**Library Catalog**" app in the **resources**. Your task is to **create a CI/CD workflow** in **GitHub Actions** to **start, test and deploy** the app to Render.com following the steps from the previous tasks.

All Books

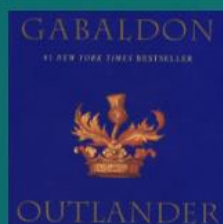
To Kill a Mockingbird

Type: Classic

[Details](#)

Outlander

Type: Other

[Details](#)

A Court of Thorns and Roses

Type: Fiction

[Details](#)

@Library Catalog