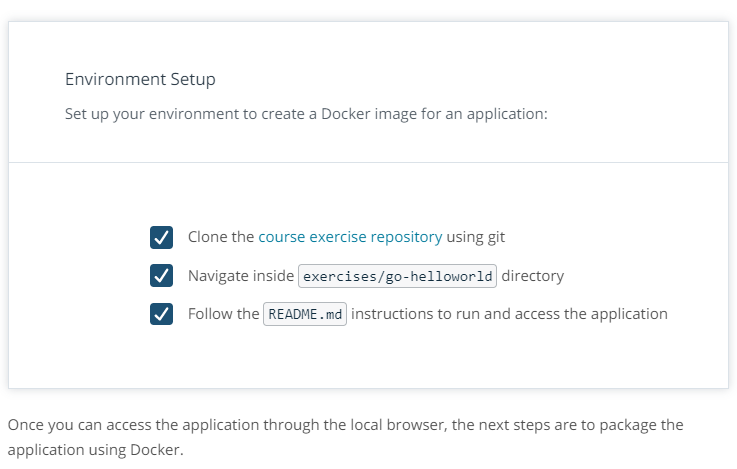
**Exercise: Docker for Application Packaging**

Package a Go web application using Docker capabilities. This exercise will involve the creation of a Docker image and pushing it to a public image registry, such as [DockerHub](https://hub.docker.com/).

Note: You will require a valid DockerHub account.



**Exercise**

Create the Docker image for the Go web application and push it to DockerHub, considering the following requirements:

Dockerfile:

* use the golang:alpine base image
* set the working directory to /go/src/app
* make sure to copy all the files from the current directory to the container working directory (e.g. /go/src/app)
* to build the application, use go build -o helloworld command, where -o helloworld will create the binary of the application with the name helloworld
* the application should be accessible on port 6111
* and lastly, the command to start the container is to invoke the binary created earlier, which is ./helloworld

Docker image:

* should have the name go-helloworld
* should have a valid tag, and a version with a major, minor, and patch included
* should be available in DockerHub, under your username e.g. pixelpotato/go-helloworld

Docker container:

* should be running on your local machine, by referencing the image from the DockerHub with a valid tag e.g. pixelpotato/go-helloworld:v5.12.3

***Note:****You will need to use docker login to login into Docker before pushing images to DockerHub.*