**DevOps Exercise**

You are an advanced lifeform from the planet Kepler-186f, you were on an expedition to earth to explore and understand the primitive lifeform called Terran (humans) but your spaceship crashed somewhere in the continent called by the natives North America.

Your mission, if you choose to accept it, is to find one of these primitive terminals called computers, connect to their largest code repository (we believe it’s called GitHub) on their so called “Internet” build a simple hello-world program which through it you’ll be able to signal us to come and pick you up and take you back to your home planet.

In order to get back home you must complete the following tasks:

1. Fork the following repository <https://github.com/pdurbin/maven-hello-world> if you don’t know what fork is google it, apparently, they have a primitive yet quite an efficient device which could answer most of your questions, it’s called Google!
2. Understand this repository -
   1. Which programming language is this?

**Answer:** java

* 1. What is maven?

**Answer:** Maven is a open-source build tool and packaging manager to build, publish, and deploy several projects at once. The tool provides allows developers to build and document the lifecycle framework. this tool has made the lives of developers easier while developing reports, checks build and testing automation setups.

* 1. How does maven work?

**Answer:** When we execute a maven command, maven looks for the pom.xml file. This file applies all the project data: the project version, the plugins needed for the project, the application's dependencies, and more. maven uses the data in the file to build, test and manage the project in the defined configurations. Each developer can define and create their own settings in the pom.xml file, thus speeding up the work

* 1. What is this pom.xml everyone keeps mentioning?

**Answer:** XML files with all the information regarding project and configuration details. It has the description of the project, information regarding the versioning and configuration management of the project[.](https://www.simplilearn.com/tutorials/maven-tutorial/maven-project-in-eclipse) The XML file is in the project home directory. When we tend to execute a task, Maven searches for the POM in the current directory.

1. In order to build the above repository, use **Azure Pipelines** or **GitHub Actions.**

**Hint: GitHub Actions is easier**

1. Change the code:
   1. Add your name to the "Hello World" message.
   2. Set Jar version to 1.0.0
2. Create a simple pipeline which would do the following actions:
   1. Increase the Patch part of the jar version ( 1.0.0 - > 1.0.1) automatically
   2. Compile the code
   3. Package it into an artifact
   4. Create an artifact item for the build
   5. Create a docker image containing the artifact - use Dockerfile:
      1. Tag the Docker image as the the Jar version automatically.
      2. The Docker image shouldn't run with root
   6. Push the docker image that was created in the previous step to Docker Hub
   7. Download and run the docker image.
3. Once it’s running, we would come and pick you up immediately

Bonus – do as much as possible from the above using Multistage Docker

If you encounter any issues with these primitive technologies, we highly recommend using Google, we believe you’ll be able to get most of the answers from this tool.

Good Luck!