## 10.11 Homework

Clone the course repo if you haven't, using the command ---🡪 ***git clone https://github.com/alexeygrigorev/mlbookcamp-code.git***

Graphical user interface, text

Description automatically generated

Contents of cloned repo are saved in a new folder I created named ***MLZoomCamp-HomeWork\_10*** as shown below.

Graphical user interface, text, application

Description automatically generated

Go to the course-zoomcamp/05-deployment/code folder and execute the following:

***docker build -t churn-model:v001 .***

It results in the image building and gets us the following output –

Text

Description automatically generated

Run it to test that it's working locally:

***docker run -it --rm -p 9696:9696 churn-model:v001***

It results in the following output –

Graphical user interface, text

Description automatically generated

And in another terminal, execute predict-test.py file: ***python predict-test.py***

You should see this:

{'churn': False, 'churn\_probability': 0.3257561103397851}

not sending promo email to xyz-123

Text

Description automatically generated

## Installing kubectl and kind

You need to install:

* kubectl - <https://kubernetes.io/docs/tasks/tools/> (you might already have it - check before installing)
* kind - <https://kind.sigs.k8s.io/docs/user/quick-start/>

## Question 1: Version of kind

What's the version of kind that you have? Use ***kind --version*** to find out.

**Ans. -** kind **version 0.11.1**  

## Creating a cluster

Now let's create a cluster with kind: ***kind create cluster --name kind-1***

Name of my cluster – **kind-1**

Shape

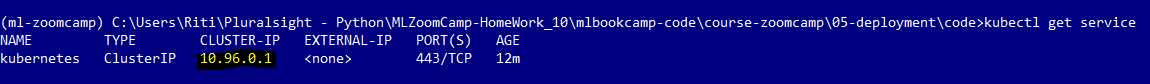
Description automatically generated with medium confidence

## Question 2: Verifying that everything works

Now let's test if everything works. Use kubectl to get the list of running services.

What's CLUSTER-IP of the service that is already running there?

**Ans. - 10.96.0.1**

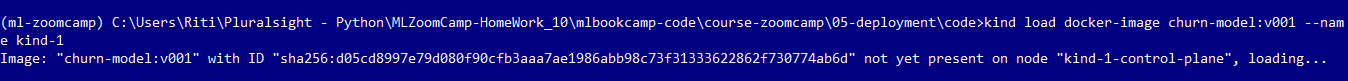


## Question 3: Uploading the image to kind

To be able to use the docker image we previously created (churn-model:v001), we need to register it with kind.

What's the command we need to run for that?

**Ans. - kind load docker-image churn-model:v001 --name kind-1**



Graphical user interface, text, application, email

Description automatically generated

[**https://kind.sigs.k8s.io/docs/user/quick-start/**](https://kind.sigs.k8s.io/docs/user/quick-start/)

## Question 4: Creating a deployment

Now let's create a deployment (e.g. deployment.yaml):

A picture containing graphical user interface

Description automatically generated

Replace <Image> and <Port> with the correct values.

What is the value for <Port>?

**Ans. - 9696**

Text

Description automatically generated

## Question 5: Pod name

Apply this deployment: ***kubectl apply -f deployment***

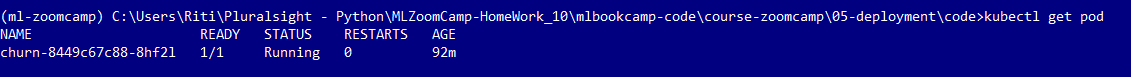
Now get a list of running pods. What's the name of the pod that just started?

**Ans. – churn-8449c67c88-8hf2l**



Graphical user interface

Description automatically generated



## Question 6: Creating a service

Let's create a service for this deployment (service.yaml):

Graphical user interface, application

Description automatically generated

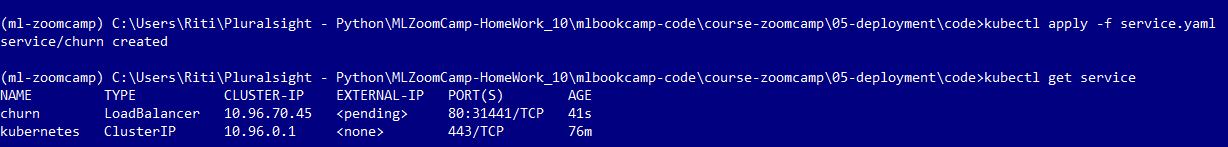
Fill it in. What do we need to write instead of <???>?

Apply this config file.

**Ans.** **-** **churn**

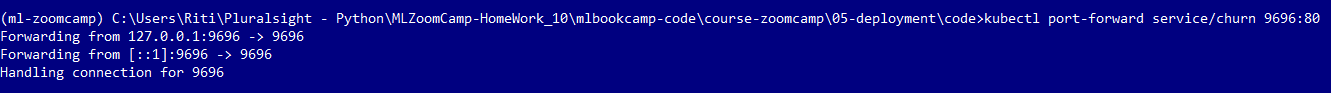
Graphical user interface, text

Description automatically generated with medium confidence



## Testing the service locally

We can do it by forwarding the 9696 port on our computer to the port 80 on the service: ***kubectl port-forward service/churn 9696:80***



Run predict-test.py from session 5 to verify that everything is working.

