# DATA GLACIER SIMPLE APPLICATION (DEPLOYMENT ON CLOUD ):

**Name: Blaise Papa**

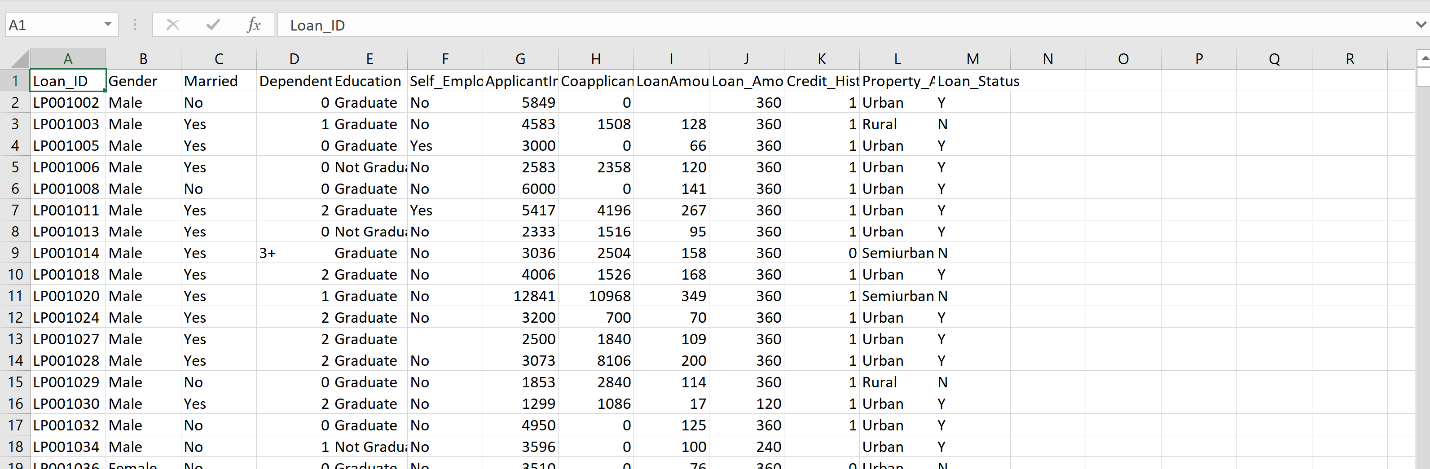
**Batch Code: LISUM01**

**Submission date:11th July 2021**

## GET TOY DATA AND LOAD

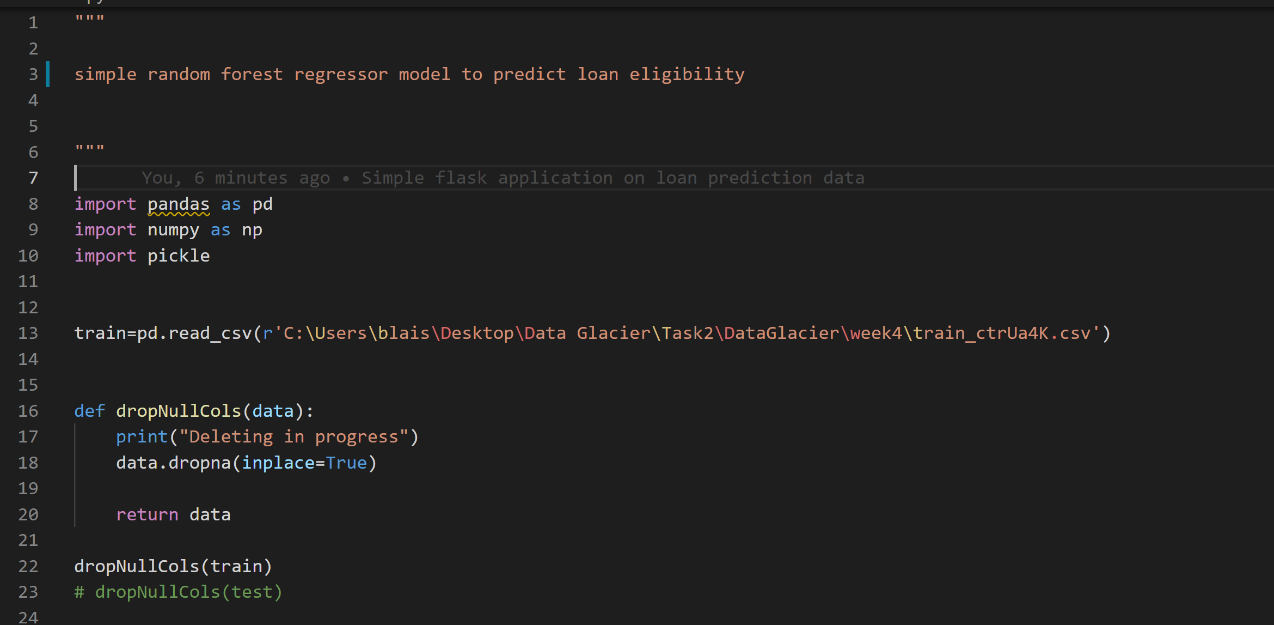
The model was developed around loan prediction data.

The data is collected on customer who apply for loans, the model is based to classify the customers between two classes; those whose loans are accepted and those whose loans are rejected.

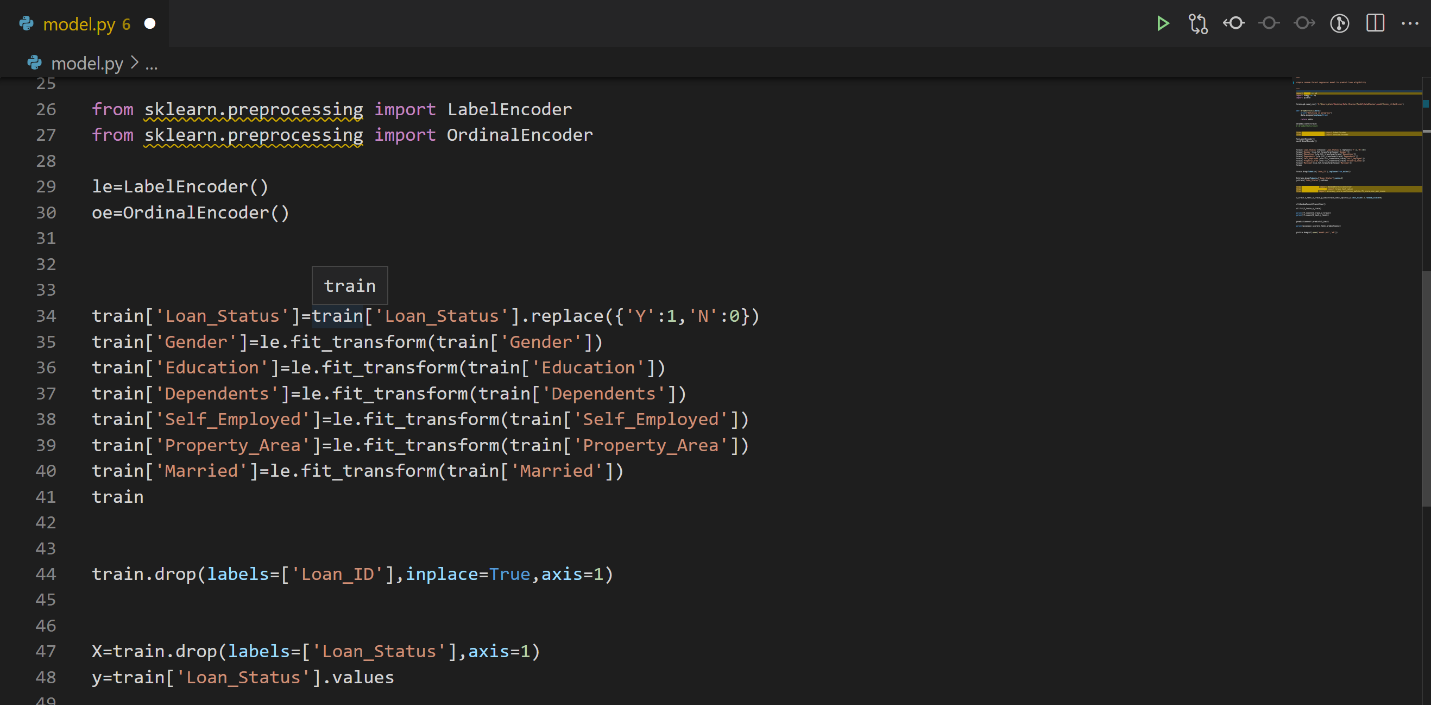


## MODEL BUILDING

The model is built and saved in the model.py file

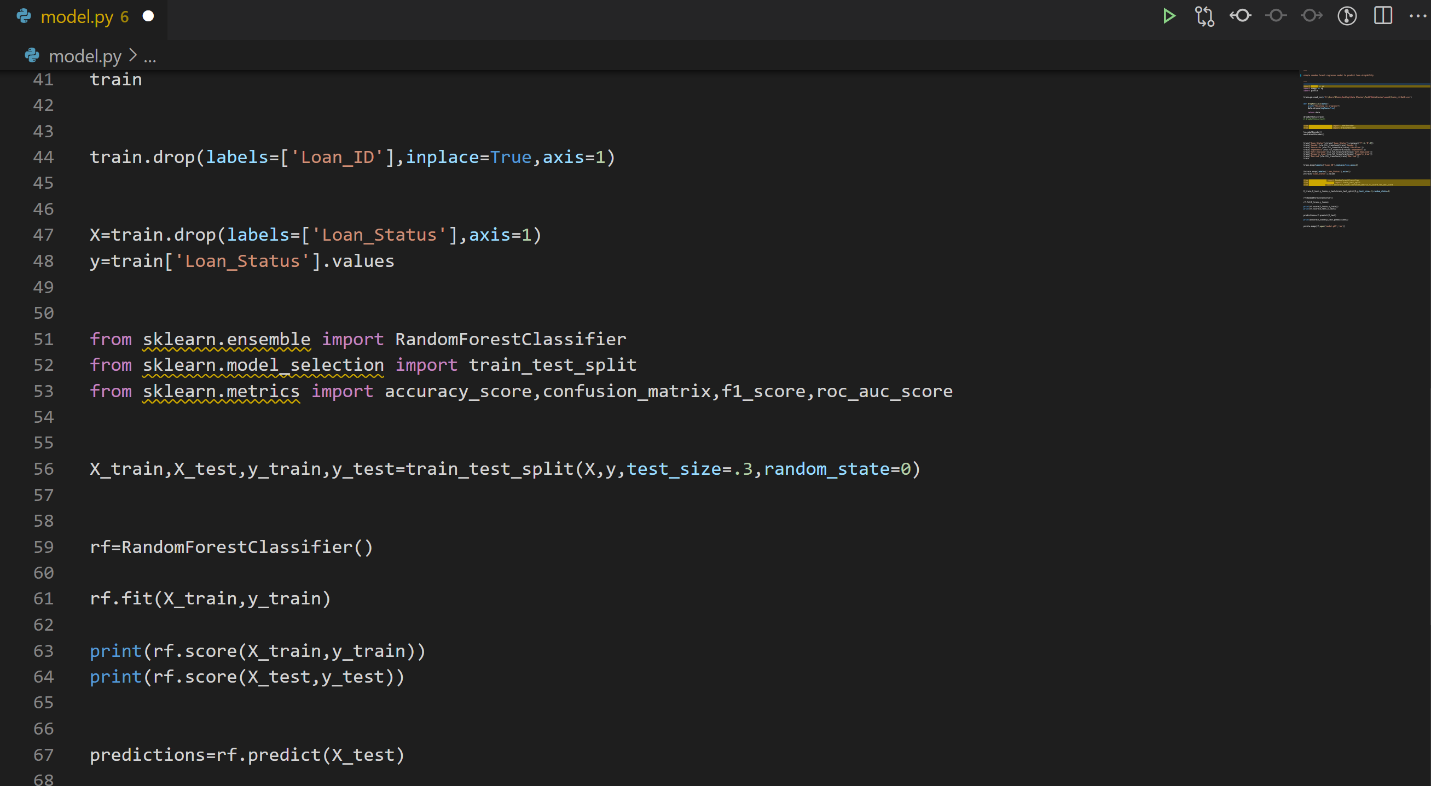
Loading data to python and converting into a data frame we can manipulate.

Simple data preprocessing steps to prepare data for model training. In this case we simply dropped all null column along with the ‘loan\_satus’ column and label encoded the object columns.



## MODEL TRAINING

For this binary classification model, we shall employ the random forest classifier, a well-known ensemble model. We also incorporate accuracy, f1 score and roc score as model evaluation metrics



## MODEL SERIALIZATION.

Once satisfied with out model we serialize it into a pickle file which will enable us to deploy the trained model.



## BUILD FLASK APPLICATION

We create a simple flask application that will serve and deploy our model.

Model deserialization

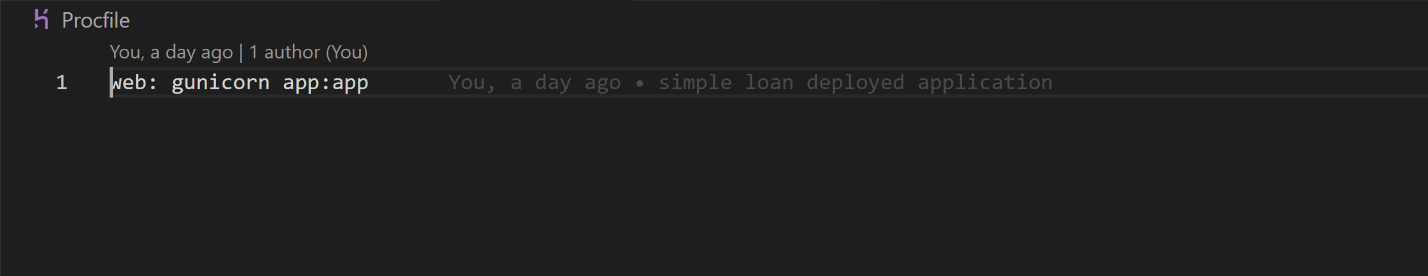
We define our flask application and then deserialize the model we earlier trained.

We create a default router which will be rendered when the default endpoint is called in this case the default endpoint(‘/’) receives data in json format and converts it to a data frame which can be processed by the model.



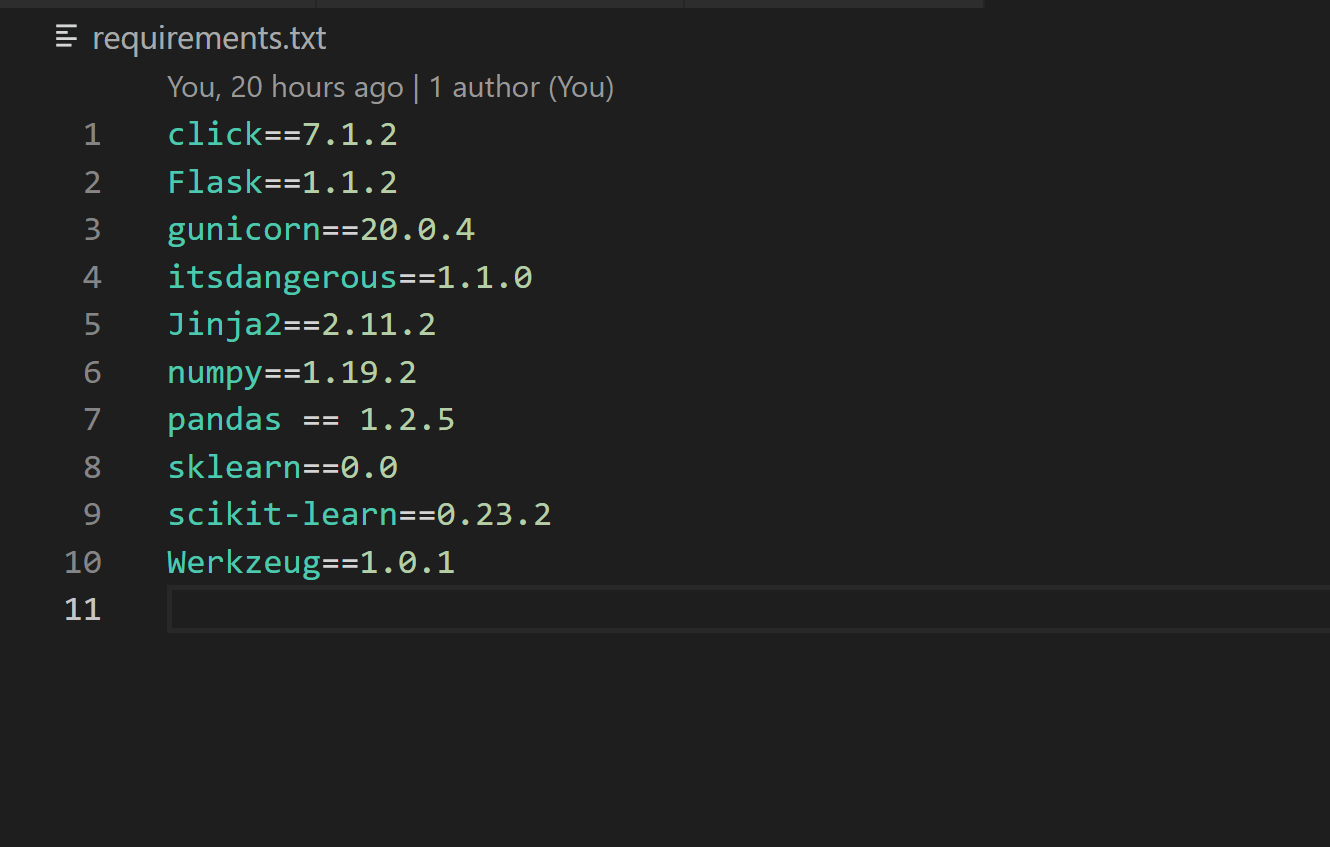
### CREATE A PROCFILE

This specifies the commands that are executed by Heroku app on startup in this case it should run the app.py script that we have created.



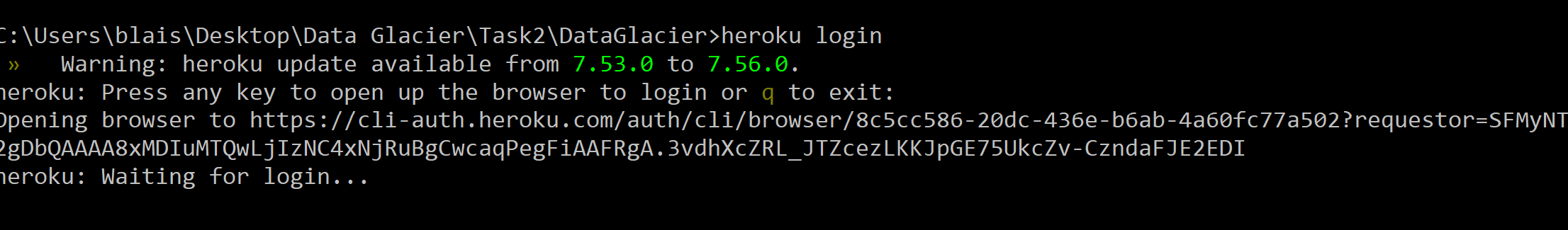
### CREATE A REQUIREMENTS.TXT FILE

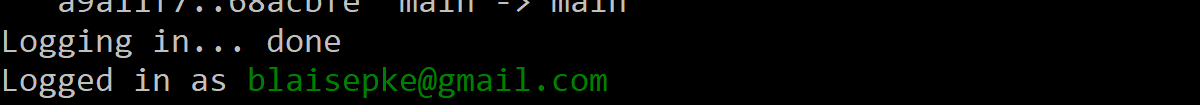
This file should contain all the dependencies/ libraries that are necessary to succefully deploy the application.



## DEPLOYING TO HEROKU

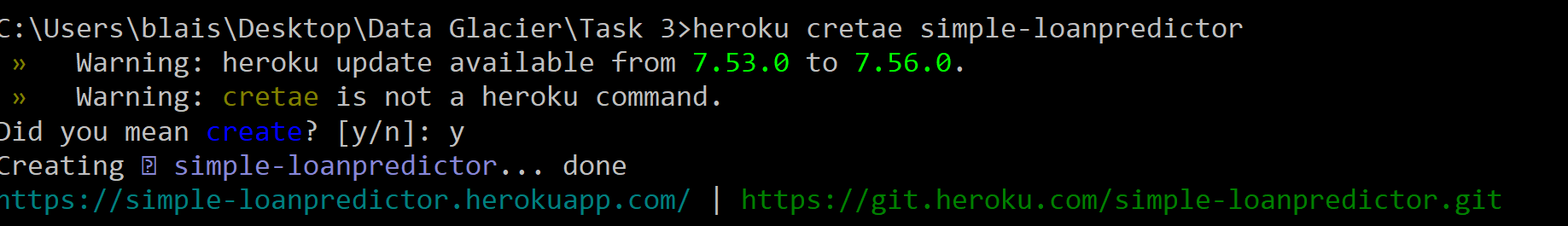
Before deployment we require a Heroku account, since we already have one we simply login to it.





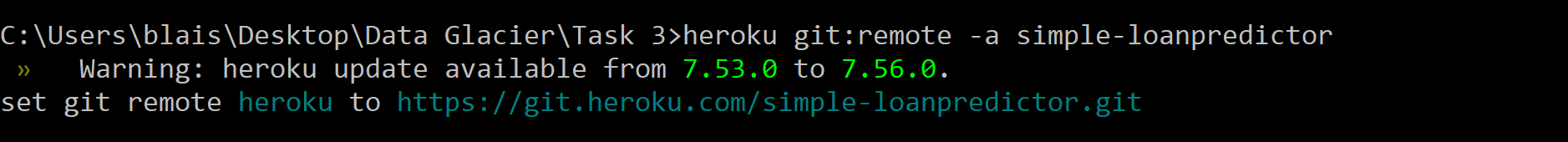
### CREATE AN APPLICATION.

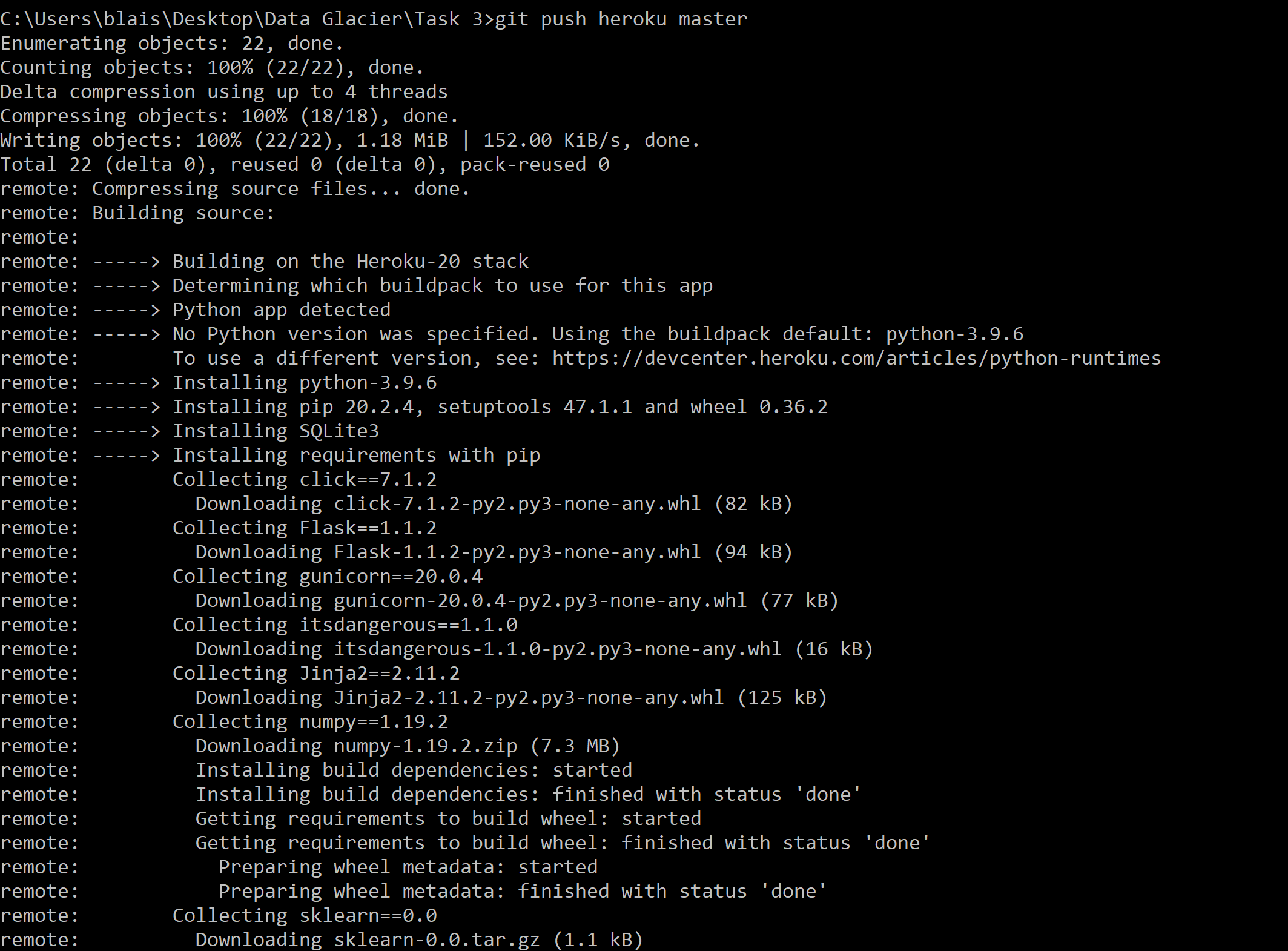
We create a new application which we will deploy to the cloud



### DEPLOY MODEL ON HEROKU

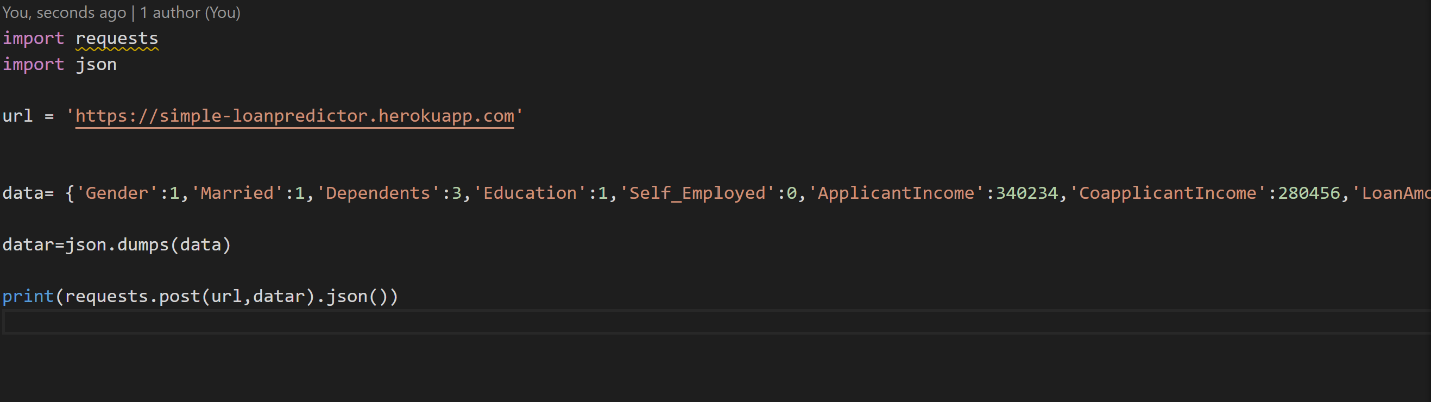
To deploy the application, we first create a remote repo with the app name so we will be able to push to





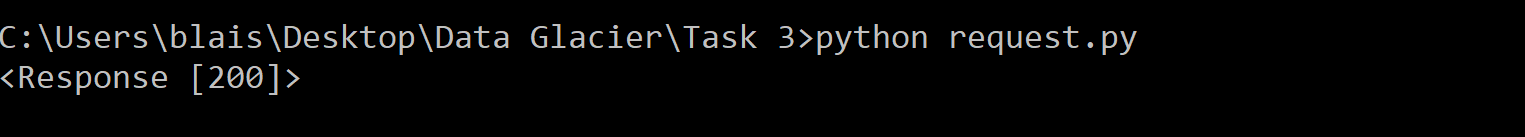
### CREATE A DUMMY API REQUEST

We create a dummy request to interact with the application to test whether it works



### CHECK FOR RESPONSE

We view the server response



The server response looks good so we check for the predictions

