

**Name:** A.P.Raja Lakshmi

**College code:** 9530

**College name:** St.Mother Theresa Engineering College

**Team id:**

**Naan Mudhalvan id:**au953021104037

**Project Name:** Image Recognition with IBM cloud

## **Problem Solution:**

Image recognition with IBM cloud visual recognition

In this part you will continue building your project.

Continue building the image recognition system by integrating IBM Cloud Visual Recognition and AI-generated captions.

Implement the image classification process using the IBM Cloud Visual Recognition API.

Use natural language generation to create captions for the recognized images.

1.IBM Cloud™ Code Engine by deploying an image classification application with pre-defined MobileNet TensorFlow.js model.

In my previous post, “Text Analysis with IBM Cloud Code Engine” you learned how to create an IBM Cloud™ Code Engine project, select the project and deploy Code Engine entities — applications and jobs to the project. You also learned how to bind IBM Cloud services (e.g., IBM Cloud Object Storage and Natural Language Understanding) to your Code Engine entities to analyze your text files uploaded to Cloud Object Storage.

In this post, you will deploy an image classification application, upload images to IBM Cloud Object Storage and then classify the uploaded images using a pre-defined MobileNet Tensorflow.js model without any training. The images are classified with labels from the ImageNet database.

## Clone the repository

On your machine, launch a terminal or command prompt and run the below commands to clone the GitHub repository and then move it to the cloned repo folder

```
Cd image-classification-code-engine
```

## Build the container images

Before building and pushing your container images, plan your image registry:

- 1.If you plan to use a private container registry like IBM Cloud Container Registry, follow the steps here to add access to a private registry.
- 2.If you plan to use public Docker Hub, run the below command to build and push three container images — frontend, backend and backend-job respectively. Replace <DOCKER\_ACCOUNT\_NAME> with your own Docker account name:  
`./deploy.sh <DOCKER_ACCOUNT_NAME>.`
- 3.If you don't wish to build your own container images, you can use the pre-built container images — vidyasagarmsc/\*. For example: `docker pull vidyasagarmsc/frontend`

## Use the container images with the solution tutorial

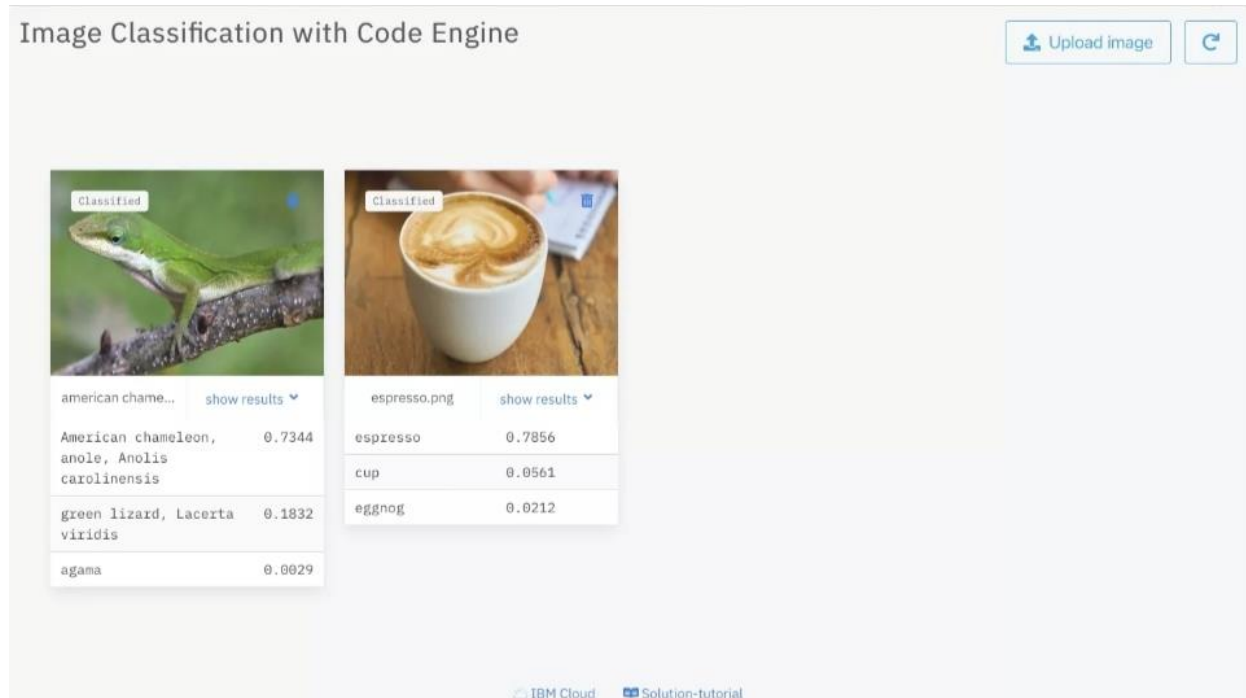
Follow the steps in the solution tutorial and use this code sample to learn about IBM Cloud Code Engine by deploying an image classification application.

Use the container images built from this code sample. Replace `ibmcom/*` with `<ACCOUNT_NAME>/*`.

Instead of uploading a text file, upload an image (.jpeg, .png) to COS. For sample images, check the images folder in this repo.

# The result

You can find the sample images to test the classification under the output folder of the repository:



Git hub link: