

Assignment 3

Artificial Intelligence

Assignment Date	3-10-2022
Student Name	K Soundarya
Register Number	921319104187

1.Create a Bucket in IBM object storage.

2.Upload an 5 images to ibm object storage and make it public. write html code to displaying all the 5 images.

3.Upload a css page to the object storage and use the same page in your HTML code.

4.Design a chatbot using IBM Watson assistant for hospital. Ex: User comes with query to know the branches for that hospital in your city. Submit the web URL of that chat bot as a assignment.

5.Create Watson assistant service with 10 steps and use 3 conditions in it.

Load that script in HTML page.

```
from google.colab import drive
```

```
drive.mount('/content/drive')
```

```
ls
```

```
cd /content/drive/MyDrive/CNN
```

```
pwd
```

```
##unzipping the zip file
```

```
!unzip Flowers-Dataset.zip
```

```
## Image Augmentation
```

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

```
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal  
_flip=True,vertical_flip=True)
```

```
test_datagen=ImageDataGenerator(rescale=1./255)
```

```
x_train=train_datagen.flow_from_directory(r"/content/drive/MyDrive/CNN/flowers",target_size=(64,64),class_mode="categorical",batch_size=24)
```

```
x_test=test_datagen.flow_from_directory(r"/content/drive/MyDrive/CNN/flowers",target_size=(64,64),class_mode="categorical",batch_size=24)
```

```
x_train.class_indices
```

```
## Creating The Model
```

```
from tensorflow.keras.models import Sequential
```

```
from tensorflow.keras.layers import  
Convolution2D,MaxPooling2D,Flatten,Dense
```

```
model=Sequential()
```

```
## Adding The Layers
```

```
##Adding Convolution2D Layer
```

```
model.add(Convolution2D(32,(3,3),activation="relu",strides=(1,1),input_shape  
=(64,64,3)))
```

```
##Adding MaxPooling2D Layer
```

```
model.add(MaxPooling2D(pool_size=(2,2)))
```

```
##Adding Flatten Layer
```

```
model.add(Flatten())
```

```
##Adding Dense-(Hidden Layers)
```

```
model.add(Dense(300,activation="relu"))
```

```
model.add(Dense(300,activation="relu"))
```

```
##Adding Output Layer
```

```
model.add(Dense(5,activation="softmax"))
```

```
##To see the added layers
```

```
model.summary()
```

```
## Compiling The Model
```

```
model.compile(loss="categorical_crossentropy",optimizer="adam",metrics=['accuracy'])
```

```
## Fitting The Model
```

```
len(x_train)
```

```
model.fit(x_train,epochs=10,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test))
```

```
## Saving The Model
```

```
model.save('flowers.h5')
```

```
## Testing The Model
```

```
import numpy as np
```

```
from tensorflow.keras.models import load_model
```

```
from tensorflow.keras.preprocessing import image
```

```
model=load_model('flowers.h5')
```

```
img=image.load_img(r"/content/drive/MyDrive/CNN/flowers/rose/537207677_f96a0507bb.jpg")
```

```
img
```

```
img=image.load_img(r"/content/drive/MyDrive/CNN/flowers/rose/537207677_f96a0507bb.jpg",target_size=(64,64))
```

```
img
```

```
x=image.img_to_array(img)
```

```
x=np.expand_dims(x,axis=0)
```

```
pred=model.predict(x)
```

```
pred = x_test.class_indices
```

```
index=['daisy','dandelion','rose','sunflower','tulip']
```

```
index[np.argmax(pred)]
```