IBM Project Name: Real-Time Communication System Powered by AI for Specially Abled

TEAM ID: PNT2022TMID19449

In[54]:

```
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
```

In [55]:

```
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Dropout
from keras.layers import Flatten
```

In [56]:

```
model=Sequential()
```

In [57]:

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

In [58]:

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

In [59]:

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

In [60]:

```
model.add(Dense(200,activation='relu'))
model.add(Dense(9,activation="softmax"))
```

In [61]:

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

In[62]:

```
len(x_train)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
 in
----> 1 len(x_train)

NameError: name 'x_train' is not defined
```

In [ ]:

```
len(x_test)
```

In [ ]:

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

In [ ]:

```
model.save("aslpng.h5")
```

Testing the model

In [ ]:

```
from keras.models import load_model
import numpy as np
import cv2
```

In [ ]:

```
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np
```

In [ ]:

```
model=load_model('asl_model_84_54.h5')
img=image.load_img(r'E:\Projects\SmartBridge\ModelGen\Dataset\test_set\D\2.
png',
                   target_size=(64,64))
```

In [ ]:

```
model=load_model("aslpng.h5")
img = image.load_img(r"/content/drive/MyDrive/IBM
project/test_set/D/10.png",target_size=(64,64))
img
```

In[]:

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

In [ ]:

```
x.shape
```

In [ ]:

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

In [ ]:

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

In [ ]:

```
pred
```

In [ ]:

```
class_name=["A","B","C","D","E","F","G","H","I"]
pred_id = pred.argmax(axis=1)[0]
pred_id
```

In [ ]:

```
print("the alphabet is ",str(class_name[pred_id]))
```