SIT789 – Robotics, Computer Vision and Speech Processing

Pass Task 9.1: Speech recognition with IBM Watson

Objectives

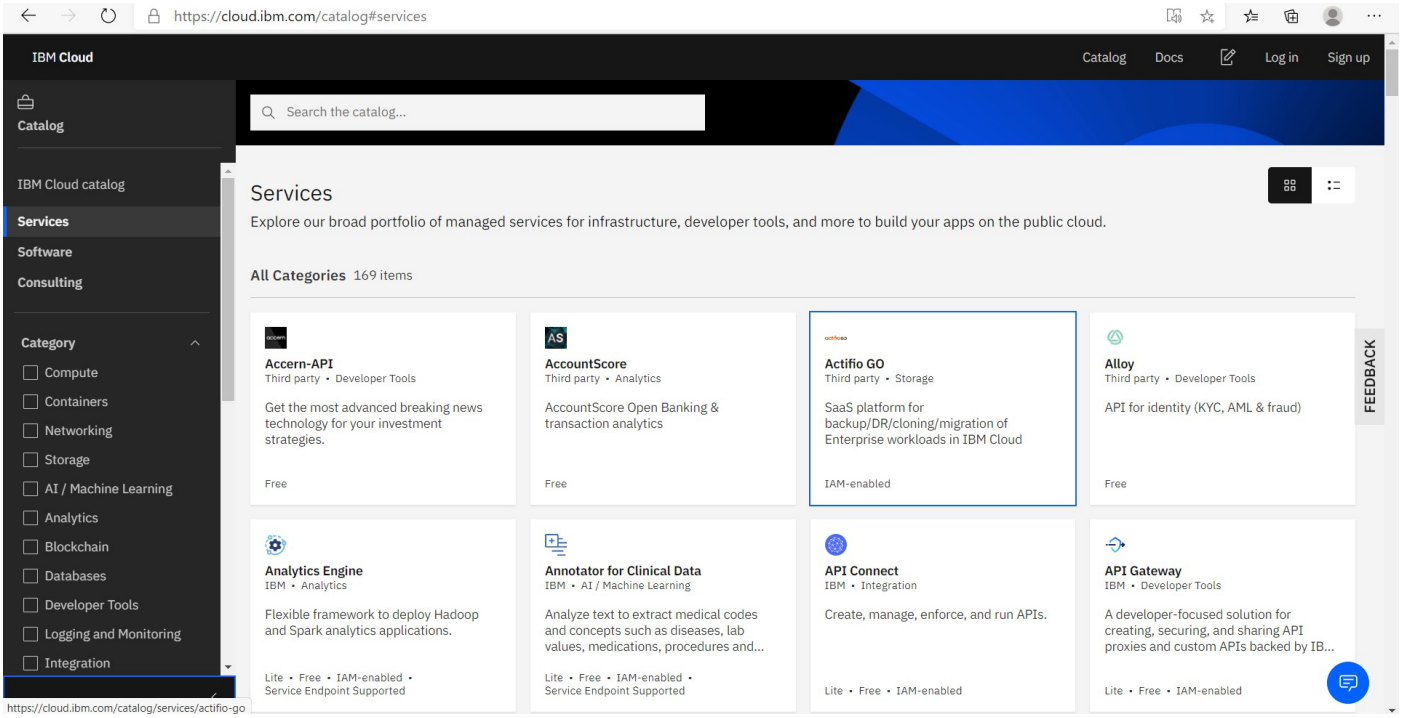
The objectives of this lab include:

- Getting hands-on experience with speech-to-text service developed by IBM Watson
- Building a speech recognition application using Watson's API

Tasks

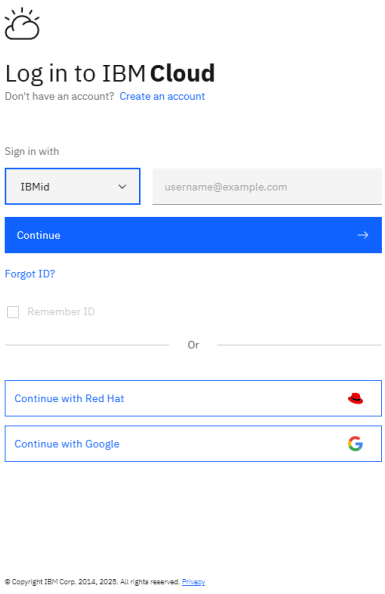
1. Speech-to-text with IBM Watson

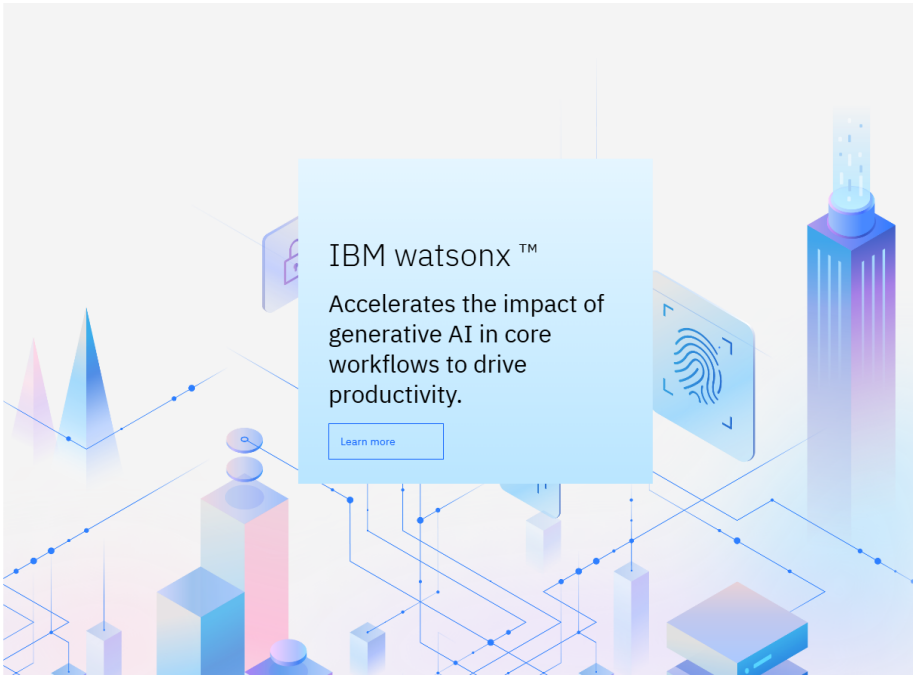
IBM cloud is a cloud computing platform that provides solutions for real-world problems and drives business value with applications, infrastructure and services across IBM Watson (AI), IoT, mobile, and more.



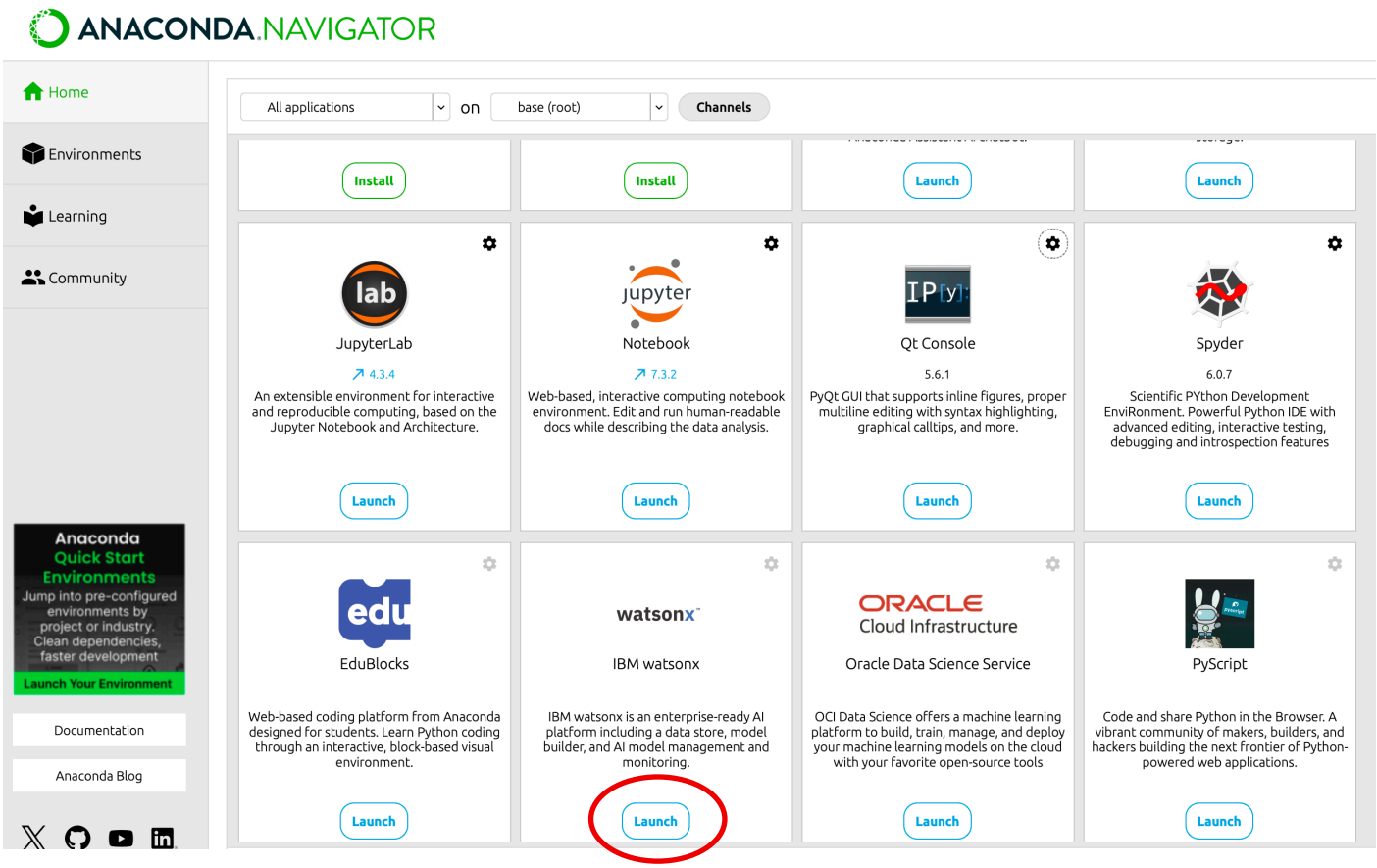
Watson was originally a question answering computer system built by IBM.

With Watson on IBM Cloud, you have access to the widest range of cognitive technologies available today to quickly and securely build smart applications. In this workshop, you will learn to use speech-to-text service by IBM Watson. First, you need to register an IBM Cloud account by following instructions given at https://cloud.ibm.com/login. You need to use your Deakin email for account registration.

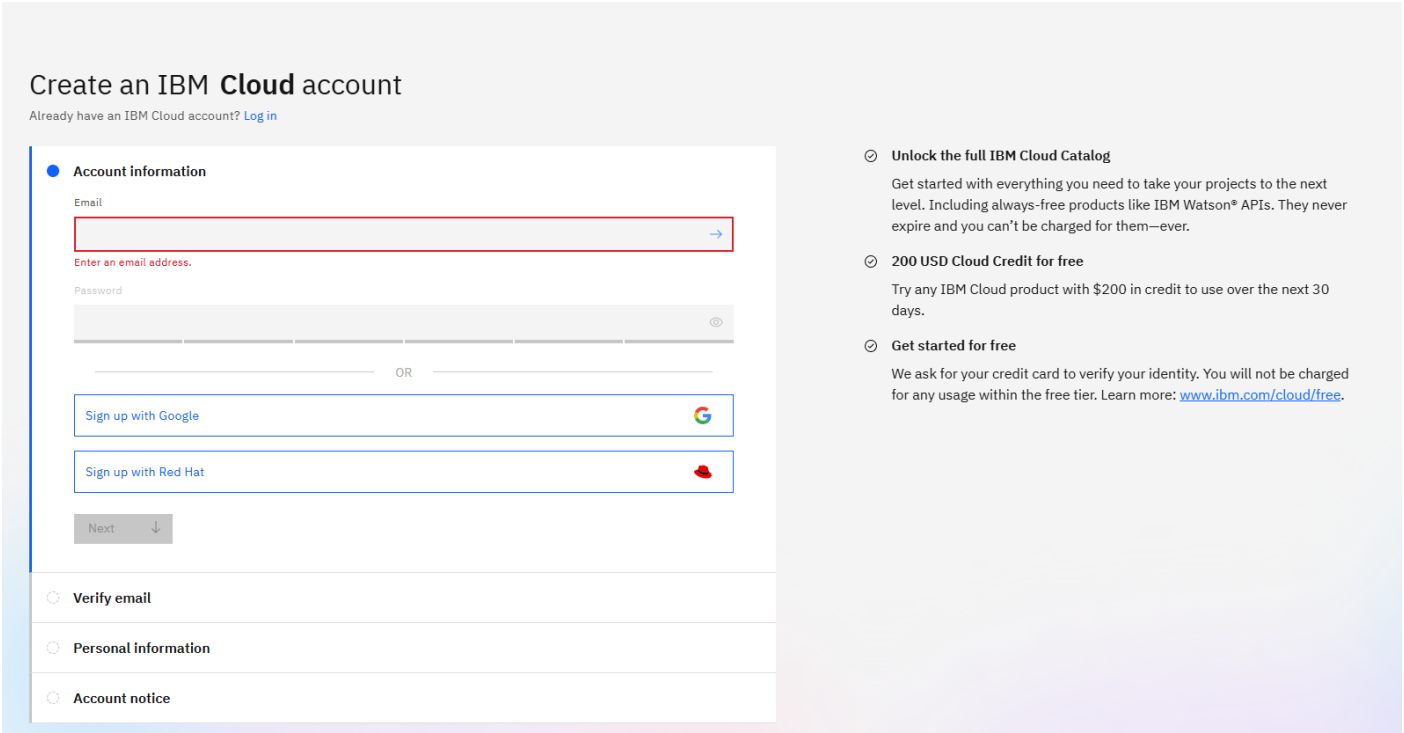




You can also navigate to IBM Cloud directly from the anaconda navigator (which can be launched by the command: “anaconda-navigator” from a terminal)

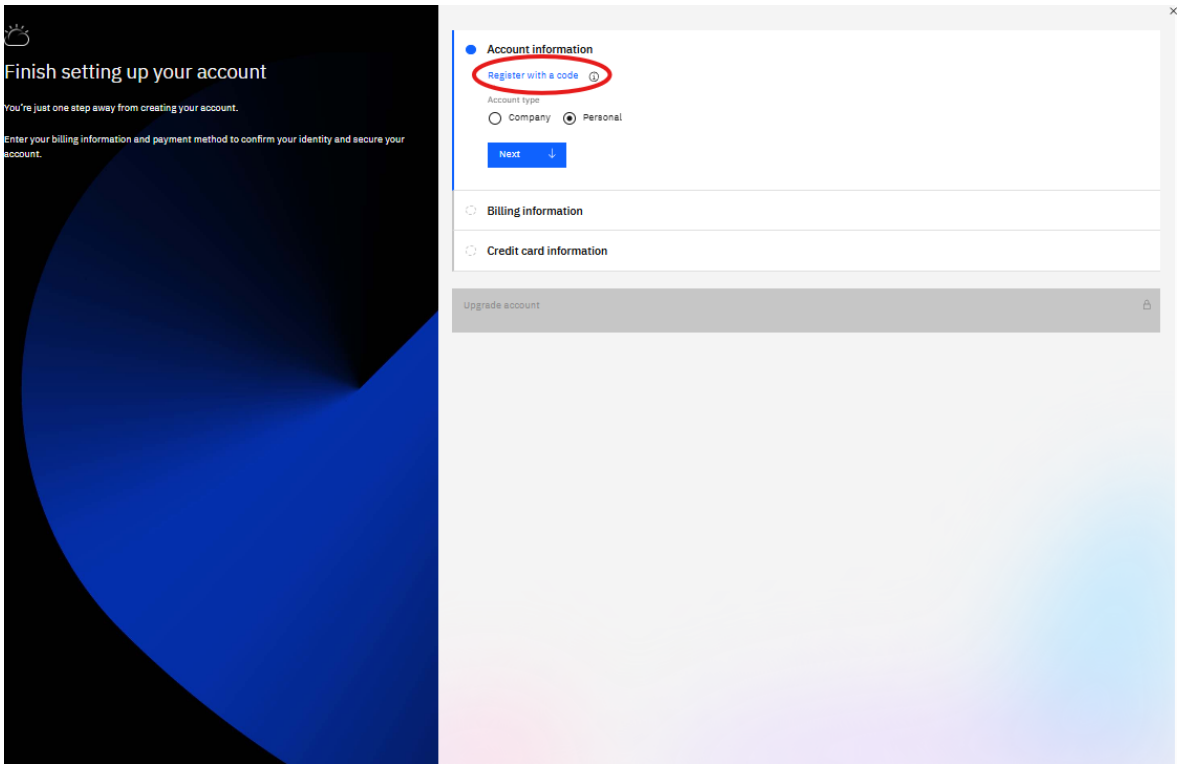


While registering your account, you can choose “Personal” for the Account Type (see the figure below).

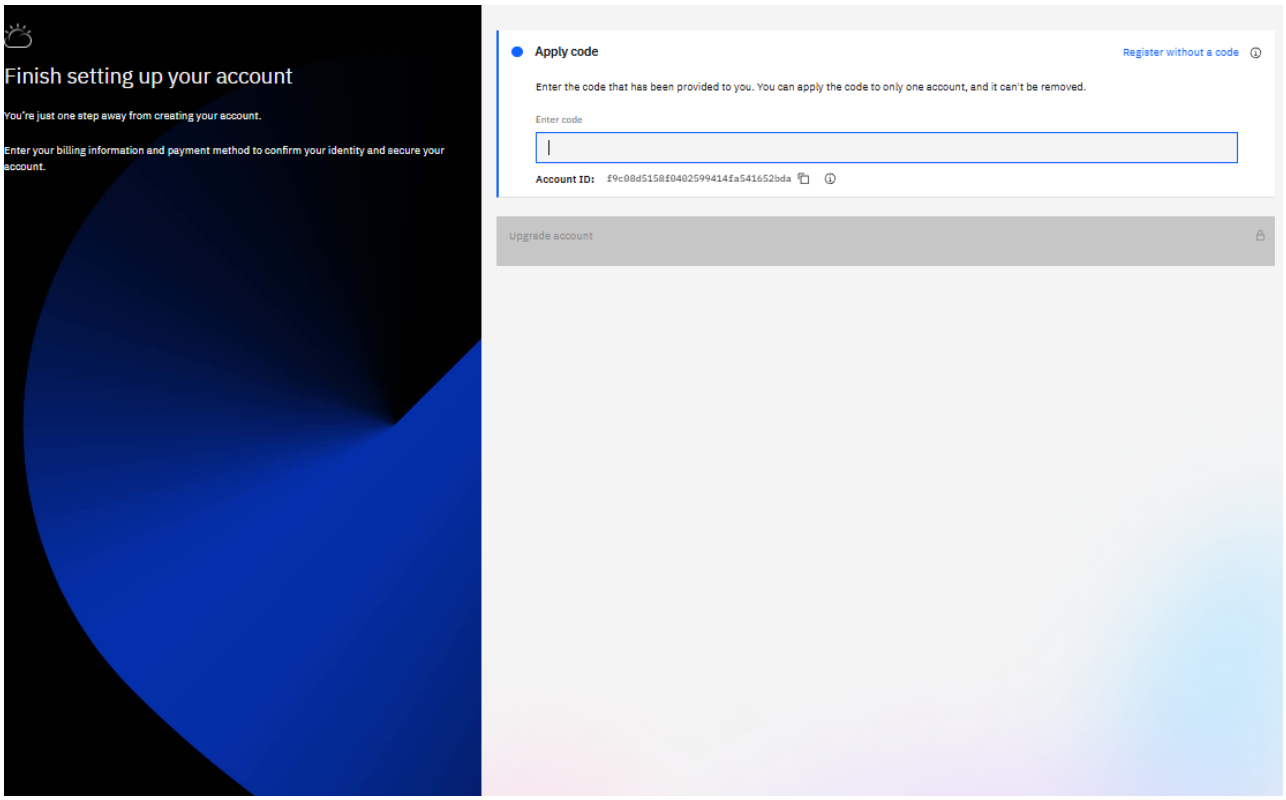


You then may be asked to provide credit card information. Note that we will use a free service for this workshop and thus your will not be charged even if you provide a credit card. However, in case that you do not want to provide a credit card, you can apply for a free feature code as follows.

- Under Account Information, select Registration with a code (see the highlighted ellipse in the figure above)
- Apply for a free feature code by following the instructions here. Note that you need to use your Deakin email to get a free feature code.

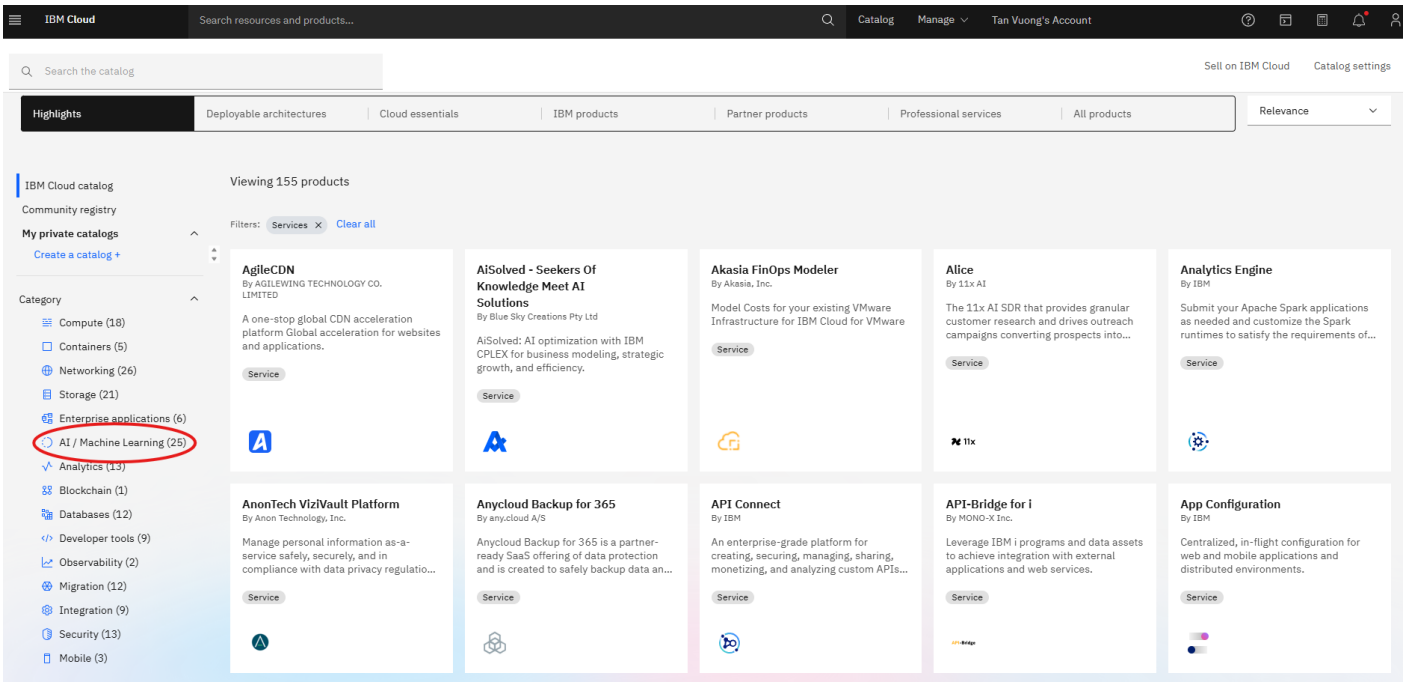


After obtaining the feature code, come back to the Account settings page and enter the code in the provided text box as shown in the figure below.

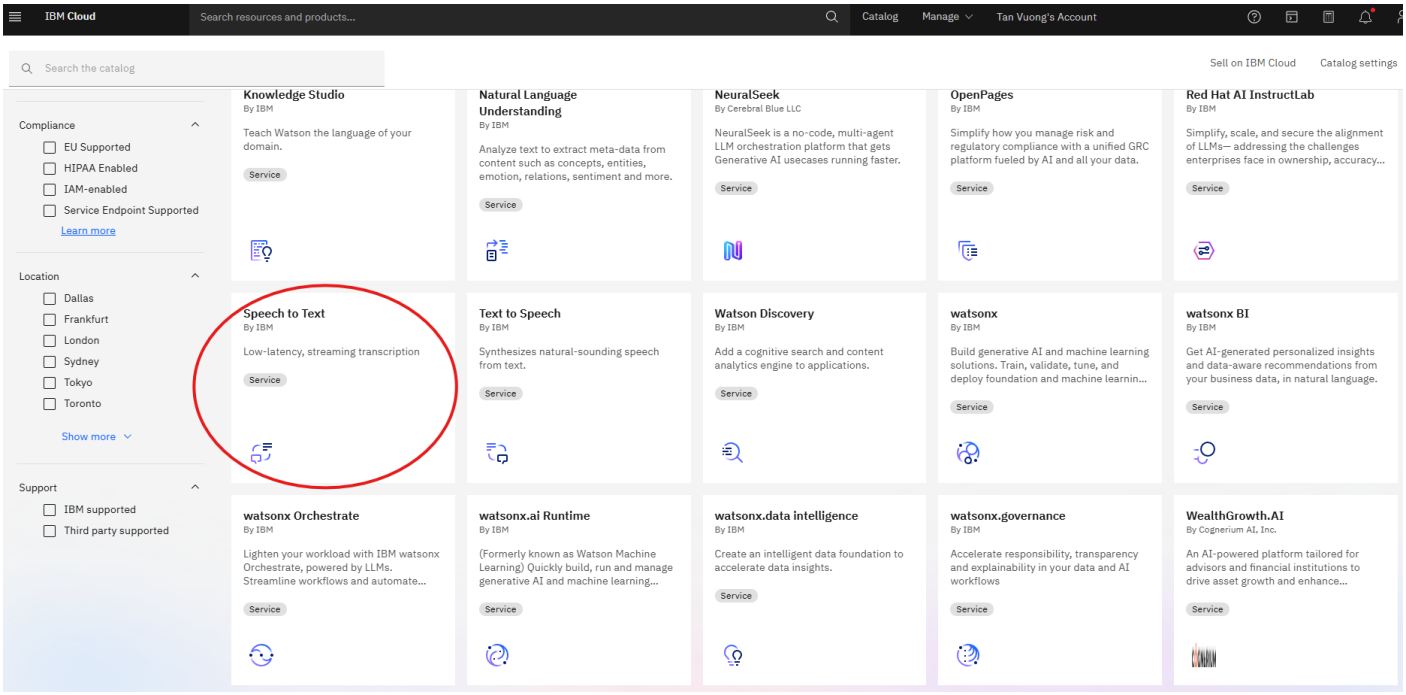


After registering your account, you can find services provided by IBM Cloud at https://cloud.ibm.com/catalog#services.

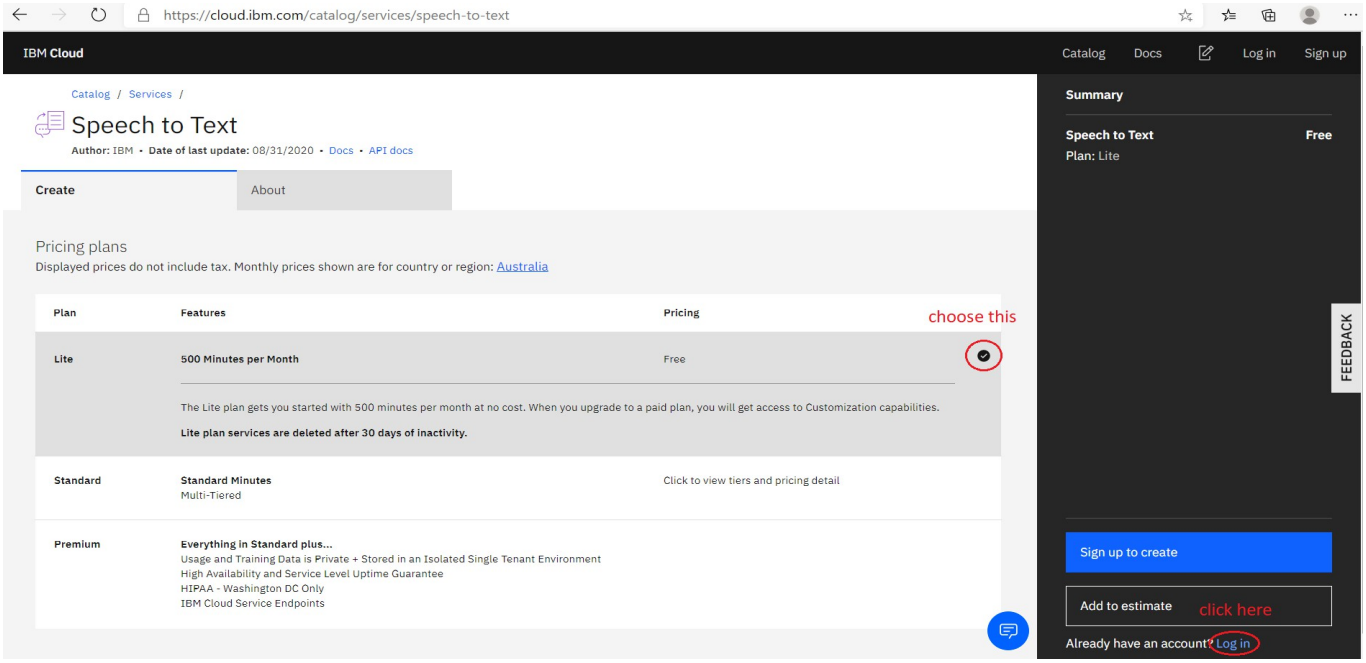
You then select **AI/Machine Learning** for **Category** as shown below,



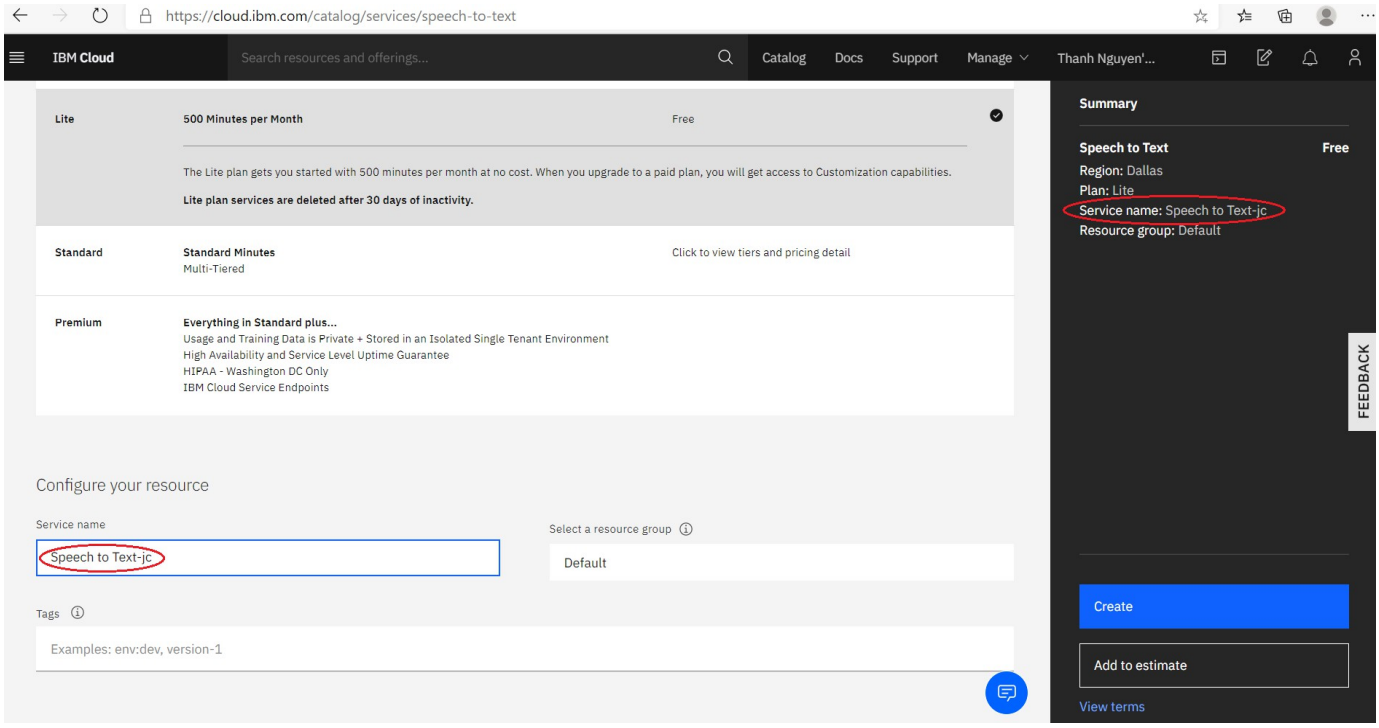
and then **Speech to Text** as shown below,



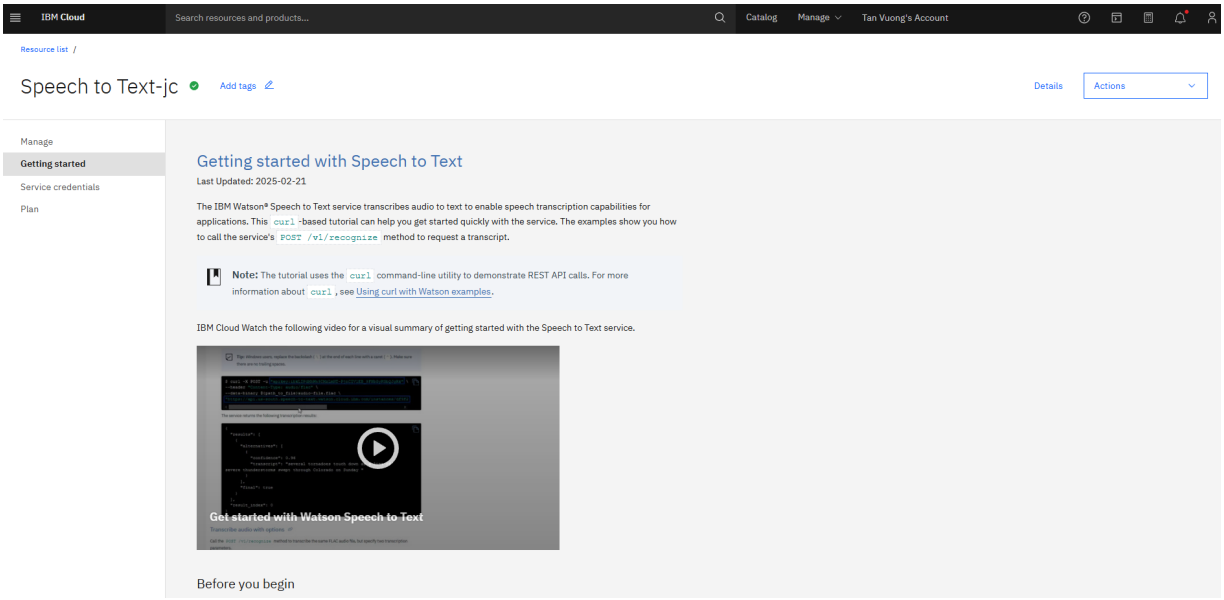
Select **Lite** with free pricing for your **Plan** (unless you wish more advanced features) and then click **Log in** (if you are not logged-in) as shown below.



After being logged in successfully, you will be given a name for your service (as highlighted in the following figure). You then need to click **Create** to create a service,



You are now linked to a tutorial page that shows you how to use the **Speech to Text** service.



Following the instructions given in the tutorial, you need to test the service with the speech data in the supplied SpeechtoTextData folder. The supplied data is collected from various well-known datasets including: the CMU ARCTIC, the LibriSpeech, and the Valentini dataset. Speech clips also vary in their quality, e.g., clips from the Valentini dataset contains various levels of noise. Transcripts for the clips are also provided.

Note:

1. Depending on the format of clips, you need to choose the right content type. In particular, for .flac files, you need to use "Content-Type: audio/flac" for the header option. For .wav files, you need to set "Content-Type: audio/wav".
2. Command lines need to be executed one by one, i.e., one command is executed at a time. For example, after entering the API key, you will be asked to provide the content type (e.g., wav).
3. You could make your own data to test this service. However, make sure that your test clips are sampled at (or above) 16kHz.

Below is a screenshot of the speech recognition results of the file arctic_a005.wav.


```
{
   "result_index": 0,
   "results": [
      {
         "final": true,
         "alternatives": [
            {
               "transcript": "will we ever forget it ",
               "confidence": 0.95
            }
         ]
      }
   ]
}
```

2. Speech recognition using Speech to Text API

In this section, we will learn how to use **Speech to Text** API for speech recognition. You can find the documentation of this API at https://cloud.ibm.com/apidocs/speech-to-text?code=python. The API is compatible with different programming languages such as Java, Python, Ruby, etc.

First, you need to install **watson_developer_cloud** and **ibm-watson** by using the following command lines.

```
pip install --upgrade watson_developer_cloud
```

```
pip install --upgrade "ibm-watson>=4.6.0"
```

You then need to authenticate your service using your API key and url as follows.

```
from ibm_watson import SpeechToTextV1
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator

authenticator = IAMAuthenticator('{APIkey}') #replace {APIkey} by your API key
speech_to_text = SpeechToTextV1(authenticator=authenticator)
speech_to_text.set_service_url('{url}') #replace {url} by your URL
```

You can test the **Speech to Text** service with a speech clip, e.g., arctic_a0005.wav, as follows.

```
import json
with open('SpeechtoTextData/arctic_a0005.wav', 'rb') as audio_file:
    speech_recognition_results = speech_to_text.recognize(
        audio = audio_file,
        content_type='audio/wav').get_result()
print(json.dumps(speech_recognition_results, indent = 2))
```

The speech recognition result can be saved to file (in json format) using the command:

```
with open('SpeechtoTextData/arctic_a0005.json', 'w') as outfile:
    json.dump(speech_recognition_results, outfile)
```

To load the result from file, you can use the following code

```
with open('SpeechtoTextData/arctic_a0005.json') as infile:
  data = json.load(infile) # load data from a json file
print(data)
```

Like Section 1, your task is to test the **Speech to Text** service using the provided API with the speech data supplied in the SpeechtoTextData folder and discuss your observations on recognition results.

Submission instructions

1. Perform tasks required in Section 1 and 2.
2. Complete the supplied answer sheet and submit it (in .pdf format) to OnTrack.