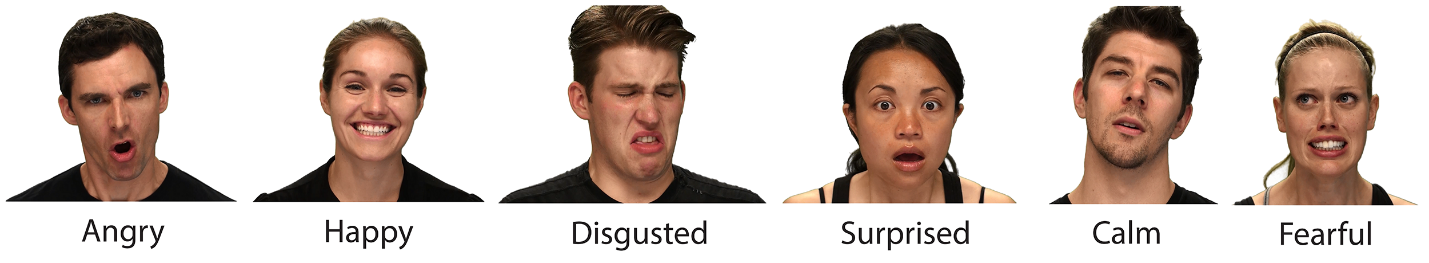
**Applied Research Project**

**CSIS 4495-091**

**Project Final Report**

**Song Emotion Recognition**



**Submitted by:**

**Saurav Dubey (300304233)**

**Submitted to:**

**Stephen Chiong**

|  |  |
| --- | --- |
| **TABLE OF CONTENTS** | |
| 1. **Statement of the problem** | **3** |
| 1. **Significance of the study** | **3** |
| 1. **Machine Learning Design Architecture** | **3** |
| 1. **Actual Report and Working application** | **4** |
| 1. **Key Findings** | **8** |
| 1. **Steps to setting the project on another environment** | **9** |
| 1. **Steps to run the project** | **10** |
| 1. **Appendix A: Progress Report 3** | **12** |
| 1. **Appendix B: Progress Report 2** | **17** |
| 1. **Appendix C: Progress Report 1** | **23** |

1. **Abstract / Statement of the problem**

Speech Emotion Recognition, abbreviated as SER, is the demonstration of endeavoring to perceive human emotion and full of feeling states from speech. This is profiting by the way that voice frequently reflects basic emotion through tone and pitch. This is additionally the wonder that creatures like canines and ponies utilize to have the option to comprehend human emotion. SER is intense on the grounds that emotions are abstract and clarifying sound is testing.

The purpose of the project is to build a machine learning model to determine the emotions of the song being played. The model trained in this project can help us determine the type of songs can be used to calm down a person and which one can make him happy.

1. **Significance of the study**

The machine learning model trained in this project can be used the used by the in the mental health hospital to determine those songs that can make the patients happy and calm. This model can also be used by the different music company to suggest customers the songs depending on their mood.

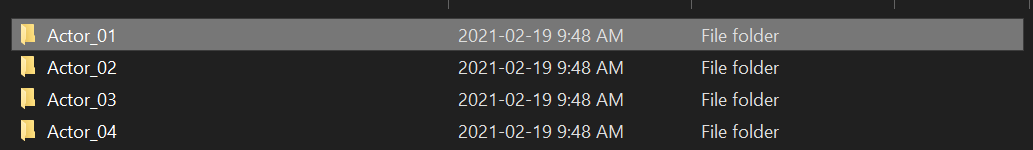
1. **Machine Learning Design Architecture:**

**Diagram

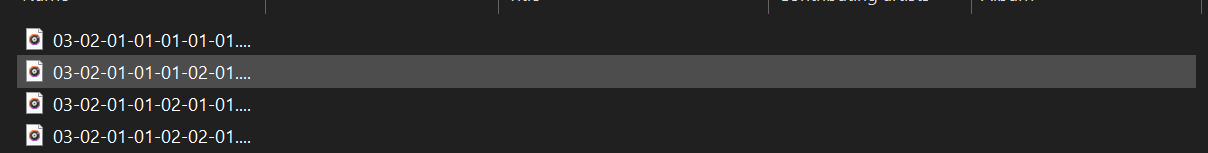
Description automatically generated**

1. **Actual Report and Working application:**

* Understanding the audio files extracted from the RAVDESS. There are various emotions recorded for around 24 actors which would be used to train the machine learning model to predict the emotions of the sample case.



Filename identifiers:



* + Modality (01 = full-AV, 02 = video-only, 03 = audio-only).
  + Vocal channel (01 = speech, 02 = song).
  + Emotion (01 = neutral, 02 = calm, 03 = happy, 04 = sad, 05 = angry, 06 = fearful, 07 = disgust, 08 = surprised).
  + Emotional intensity (01 = normal, 02 = strong). NOTE: There is no strong intensity for the 'neutral' emotion.
  + Statement (01 = "Kids are talking by the door", 02 = "Dogs are sitting by the door").
  + Repetition (01 = 1st repetition, 02 = 2nd repetition).
  + Actor (01 to 24. Odd numbered actors are male, even numbered actors are female).
* Install and Imported required library.

**Librosa** is a python package for music and sound analysis. It gives the structure blocks important to make music data recovery frameworks.

**SoundFile** can be used to read and write sound files.

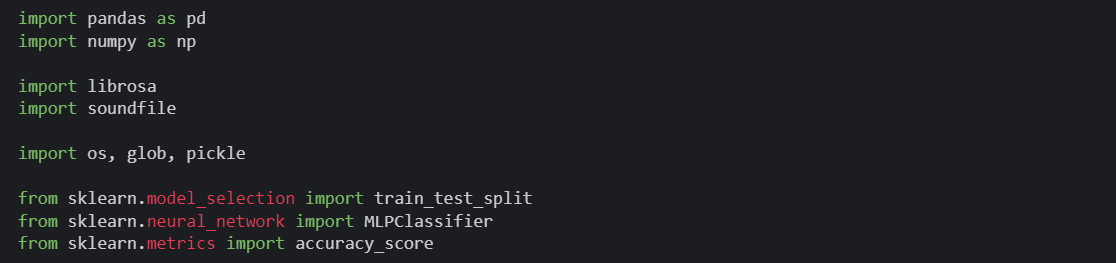
**Pandas** is a fast, powerful, flexible, and easy to use open-source data analysis and manipulation tool, built on top of the Python programming language.

**NumPy** is a Python library used for working with arrays.

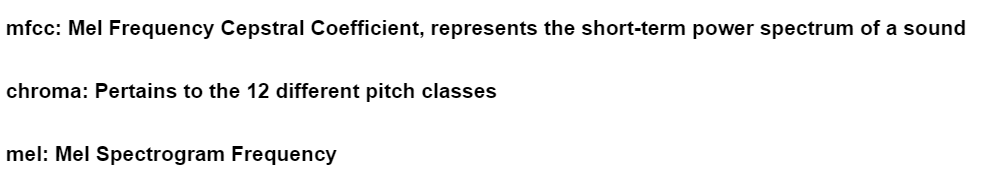
**Scikit-learn** is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering, and dimensionality reduction.

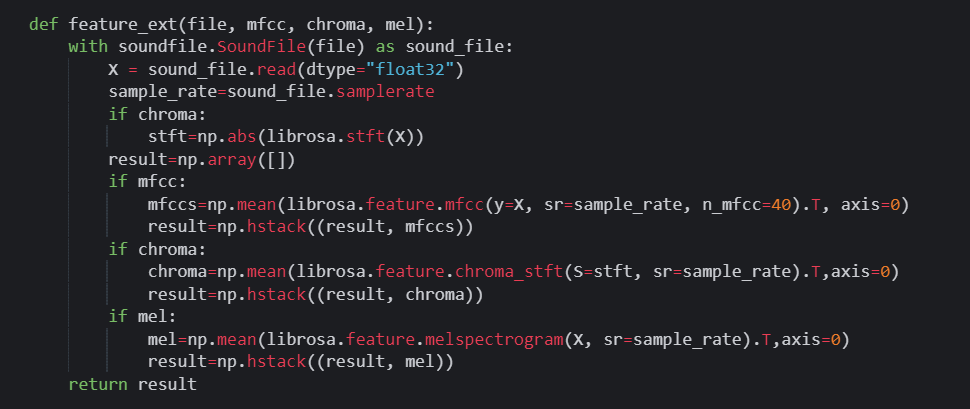
**MLP** is a type of artificial neural network (ANN). Simplest MLP consists of at least three layers of nodes: an input layer, a hidden layer, and an output layer.

In Scikit-learn “**MLP Classifier**” is available for Multilayer Perceptron (MLP) classification scenarios.



* Defined a function to extract the three features of the audio files in the data set. These features would help to create the machine learning model using MLP classifier that would predict the emotion of sample file.



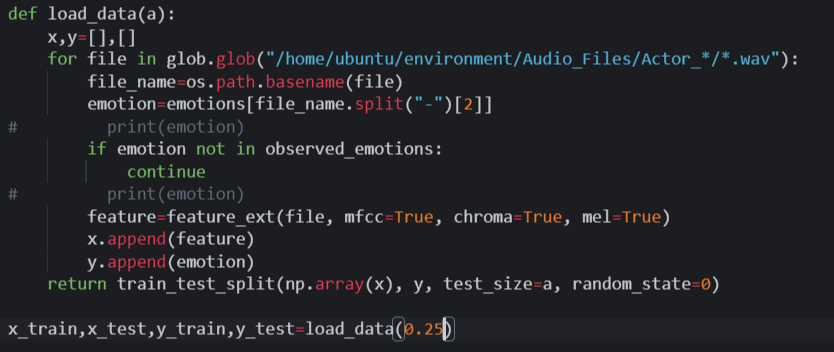


* Created a list to hold the emotions in shown in the files as well as an array of emotions that would be observed using the model.



* To create train and test dataset, there is a need to load the data (**load\_data(a)**) so I created a function which would access the file based on the filename identifiers. The features of the files are extracted using the feature extraction method (**feature\_ext(file, mfcc, chroma, mel)**) created earlier. While creating a train and test dataset the test size is set to 25% of the dataset.

Train and Test data:



* The size of train and test dataset is mentioned below:



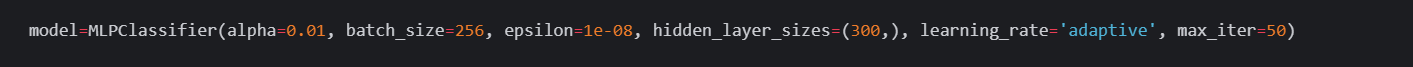


* The number of features extracted from all the files are:





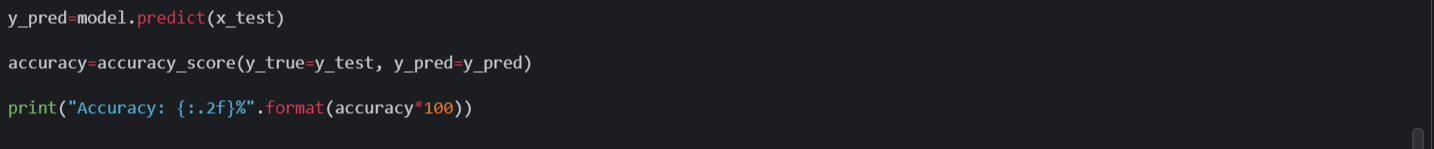
* The machine learning model is created using MLP Classifier which uses adaptive learning rate.



* The train dataset created earlier is fit into this model so that the model is trained, and we would be able to predict the emotions of the sample sound file.

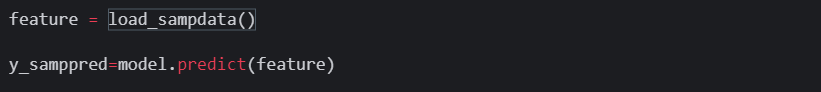


* Now a prediction is made on the test dataset so that we would be able to check the accuracy of the model





* Now the sample file “**creature-sobbing-in-fear.wav**” using a method (**load\_sampdata()**) that extracts the feature and the features are then used to predict the emotions.



* After the prediction is made on the sample file, we now know the emotions being resonated from the file which here seems to be a “**fearful**” emotion.
* 
* 

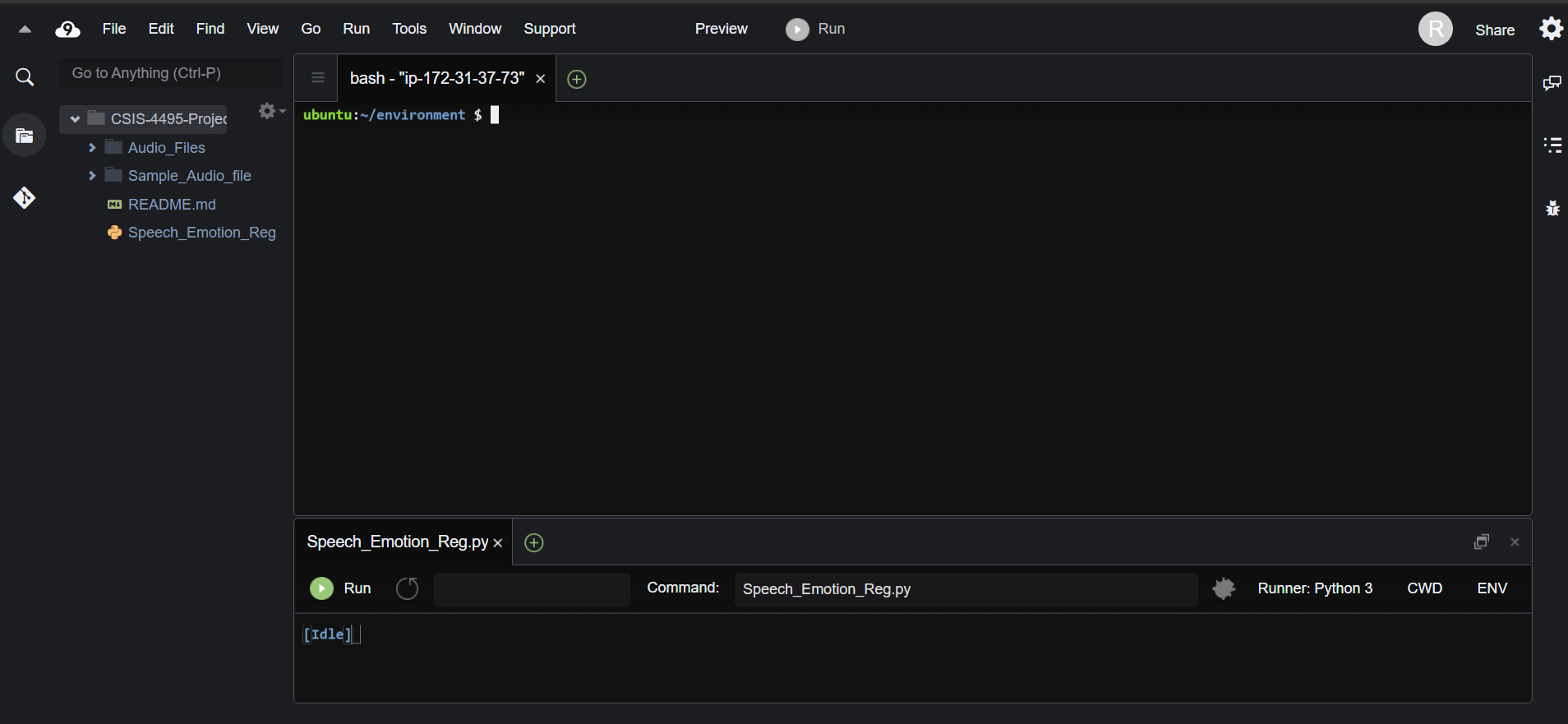
1. **Key Findings:**

The project can deploy a machine learning model which was successful in determining the emotions of the audio/song being played. Here a creatures fearful crying was used as a sample and the was able to determine the fearful emotions in the audio. So, this model can be used by any company that hoping to use AI to influence their customer to provide best customer service through their automated teller machine. Also, this project can help us determine the type of songs can be used to calm down a person or which one can make him happy.

1. **Steps to setting the project on another environment:**

To setup the project on another Ubuntu environment on AWS cloud9, follow below steps:

* First and foremost, create an [AWS account](https://aws.amazon.com/resources/create-account/) if you do not have it.
* Now create an environment in Cloud9 - [Ubuntu Environment](https://docs.aws.amazon.com/cloud9/latest/user-guide/create-environment-main.html)



* Now we need check if the environment has python since the project was implemented using python programming language. Most of the environment would already have python but to be sure run below command to check.

**python3 --version**

* Now we need to install all libraries required for the project. Which are **pandas**, **numpy**, **librosa**, **sndfile** using python pip. Since python3 is used so pip3 should be used to install any required library.

**pip3 install pandas**

**pip3 install librosa**

**pip3 install sndfile**

* While install **sndfile** library, there are few errors in Linux environment, so a workaround needs to be done. Kindly run below commands and then again install the **sndfile** library. **sudo** here provides root privileges in Linux.
* **sudo apt-get install python3.6-dev**
* **sudo apt-get install build-essential autoconf libtool pkg-config python-opengl python-pil python-pyrex python-pyside.qtopengl idle-python3.6 qt4-dev-tools qt4-designer libqtgui4 libqtcore4 libqt4-xml libqt4-test libqt4-script libqt4-network libqt4-dbus python-qt4 python-qt4-gl libgle3 python-dev libssl-dev**
* **sudo apt-get install build-essential libssl-dev libffi-dev python3.6-dev**
* **sudo apt-get install lib32ncurses5-dev**
* **sudo apt-get install libsndfile-dev**
* **sudo pip3 install libsndfile-dev**
* **sudo pip3 install libsndfile**
* **sudo pip3 install sndfile**
* Now the python file needs to be written and run to execute the project.

1. **Steps to run the project.**

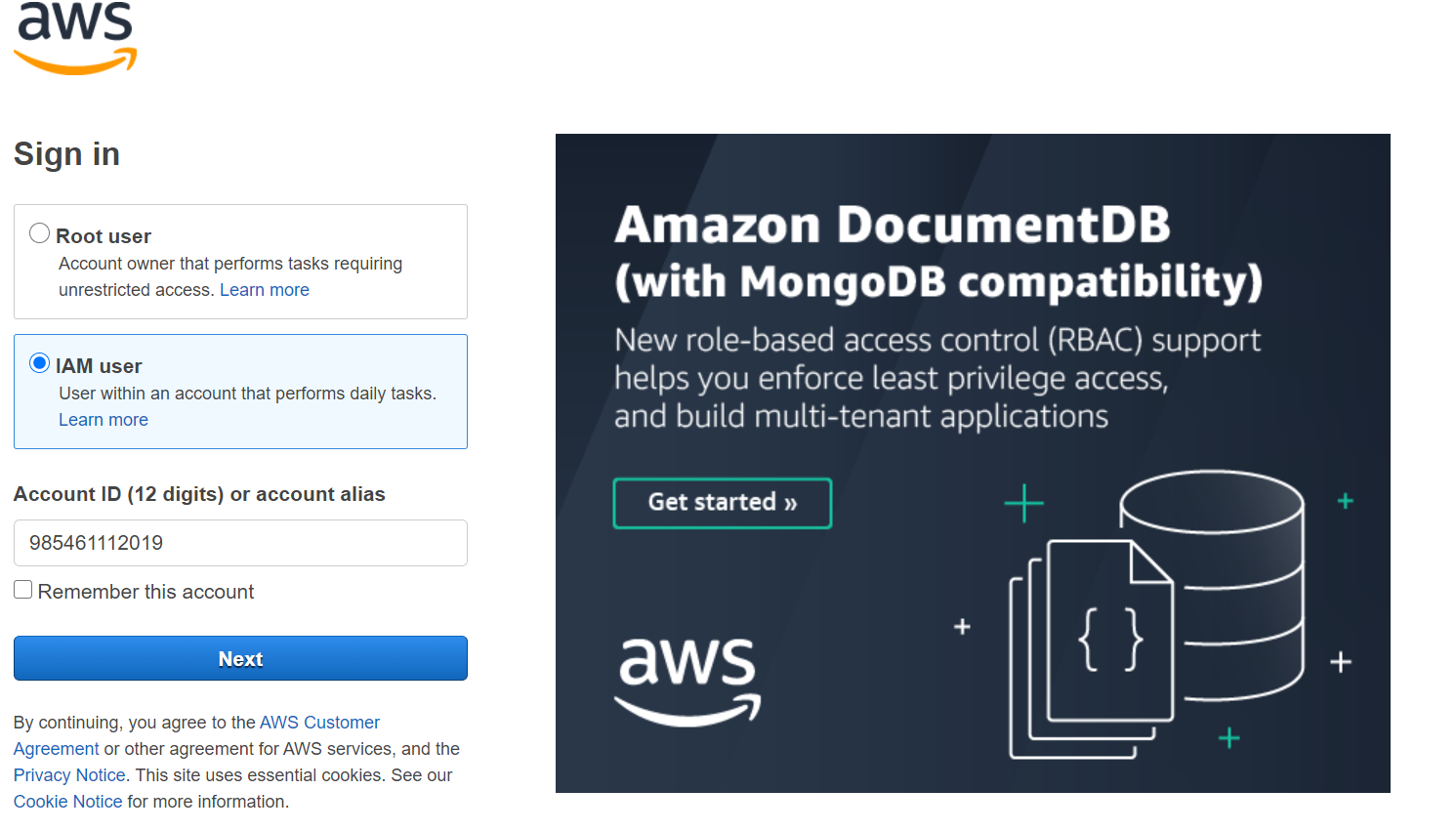
The project is implemented using AWS Cloud9 coding environment. Cloud9 is a cloud-based integrated development environment (IDE) that lets you write, run, and debug your code with just a browser.

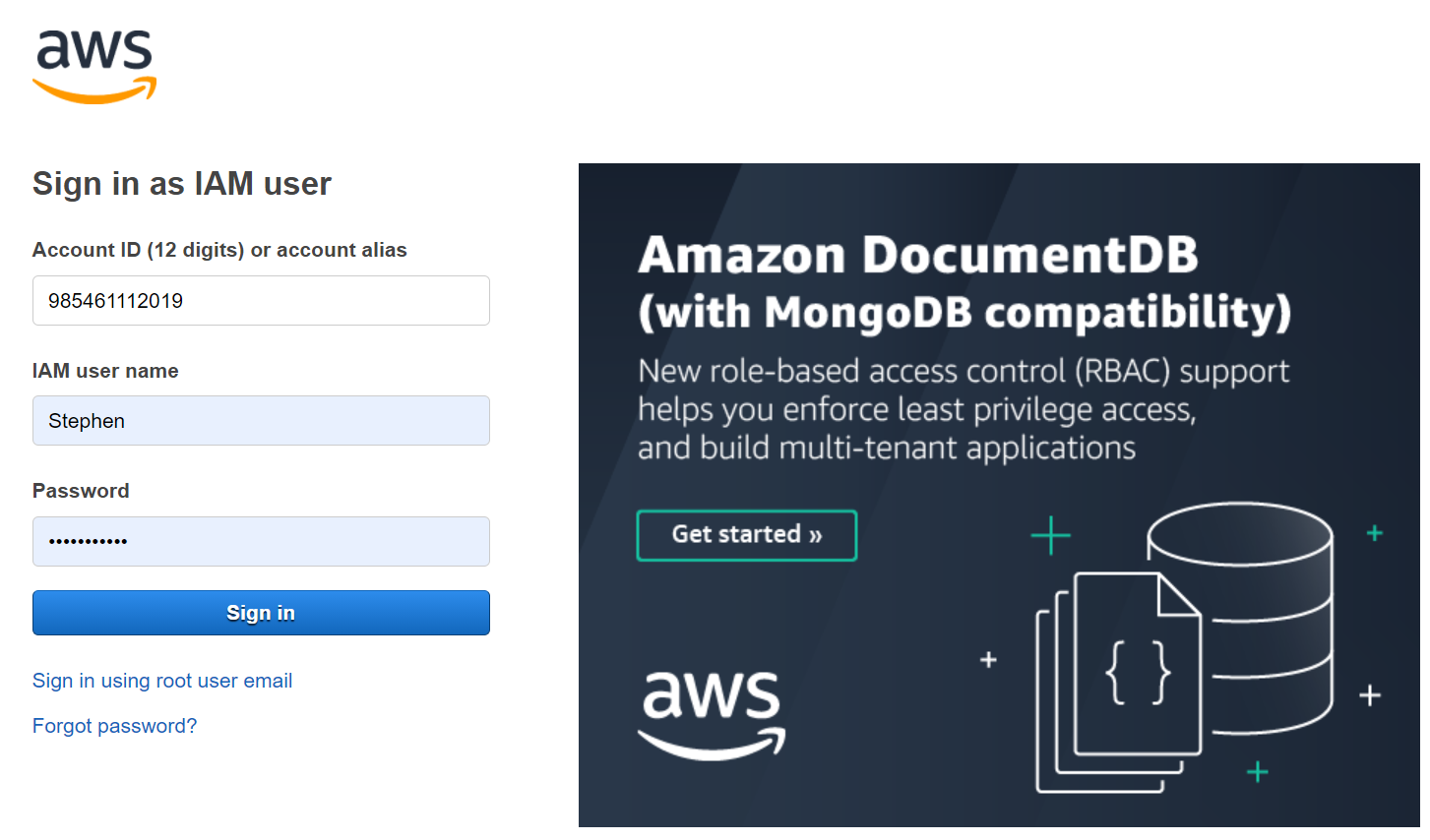
* + - I have created a separate IAM user on AWS, whose credentials are as follows:

Account Number: 985461112019

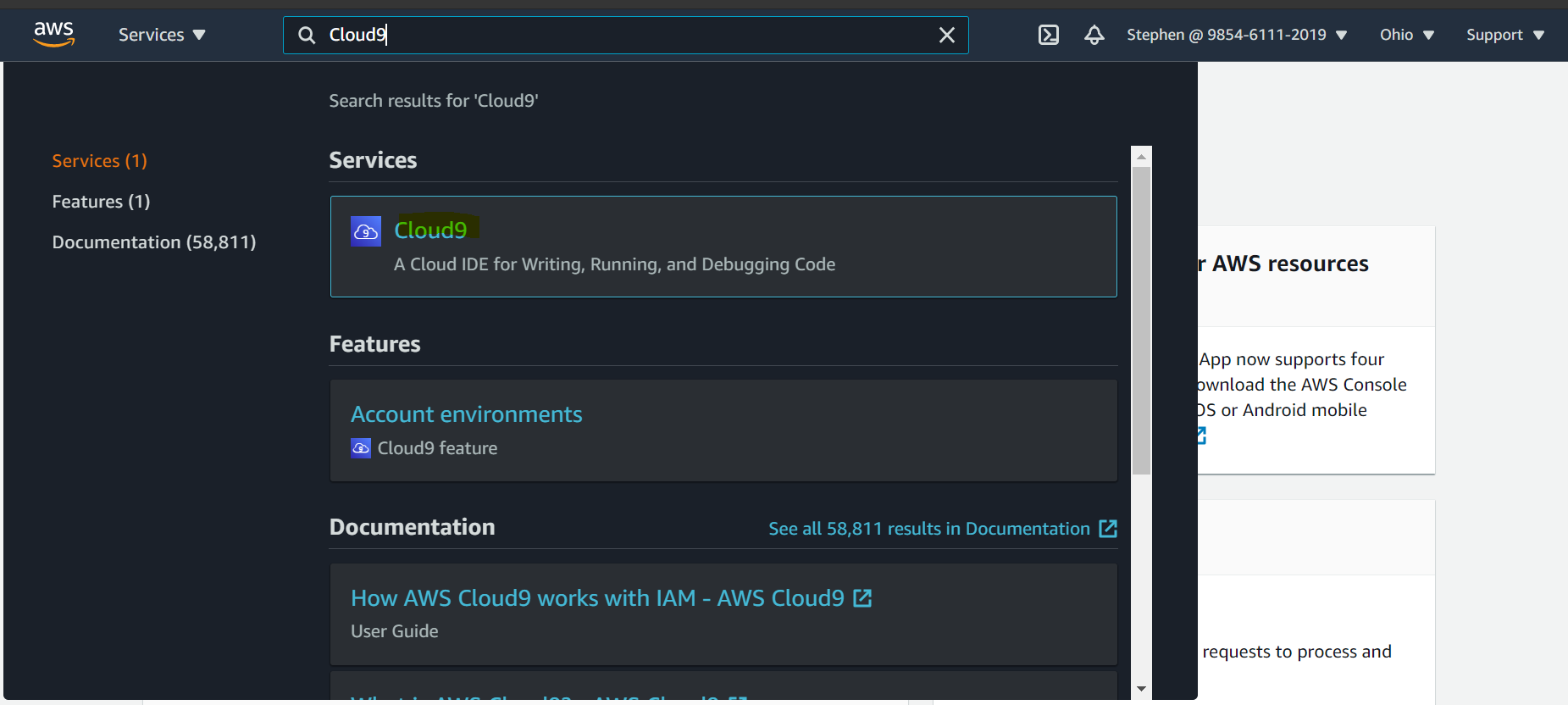
Username: Stephen

Password: stephen@123

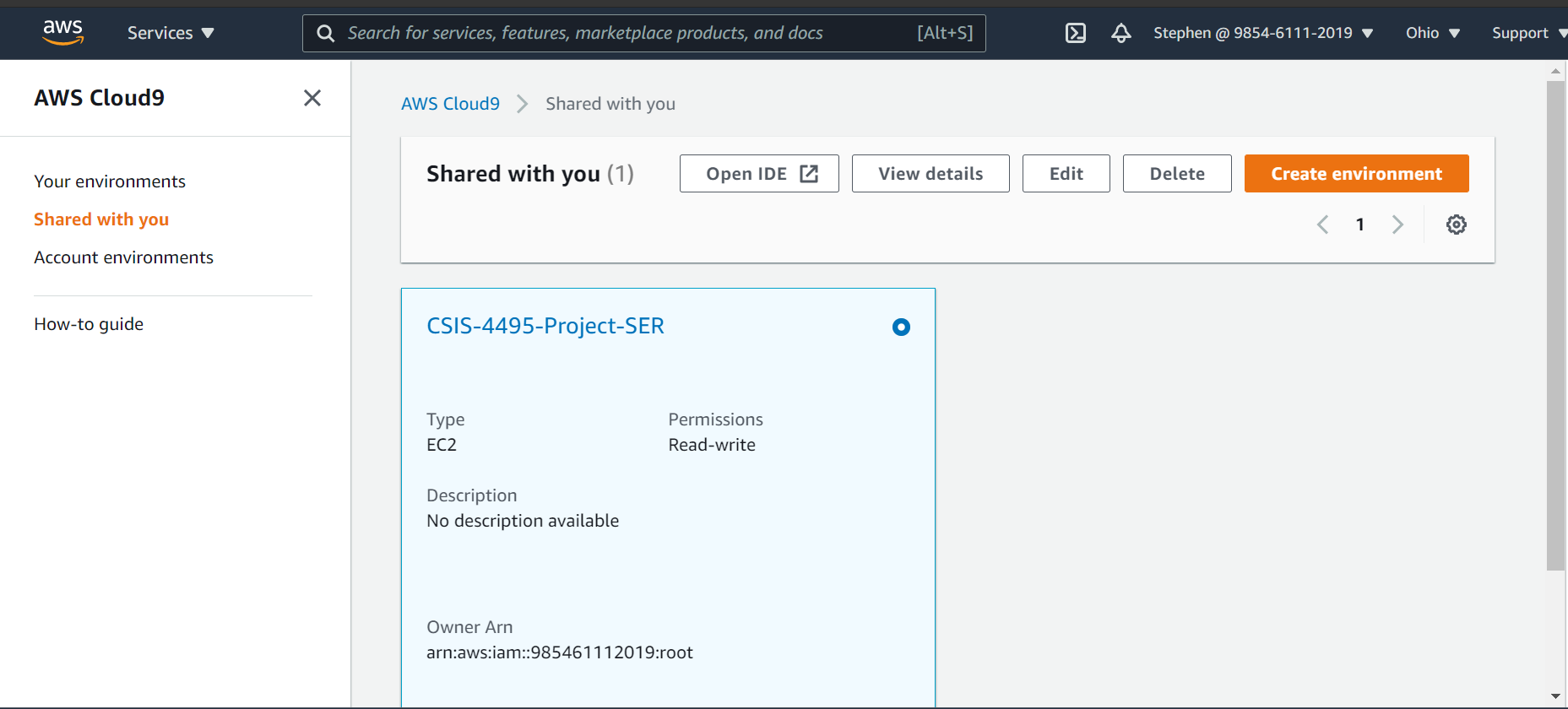




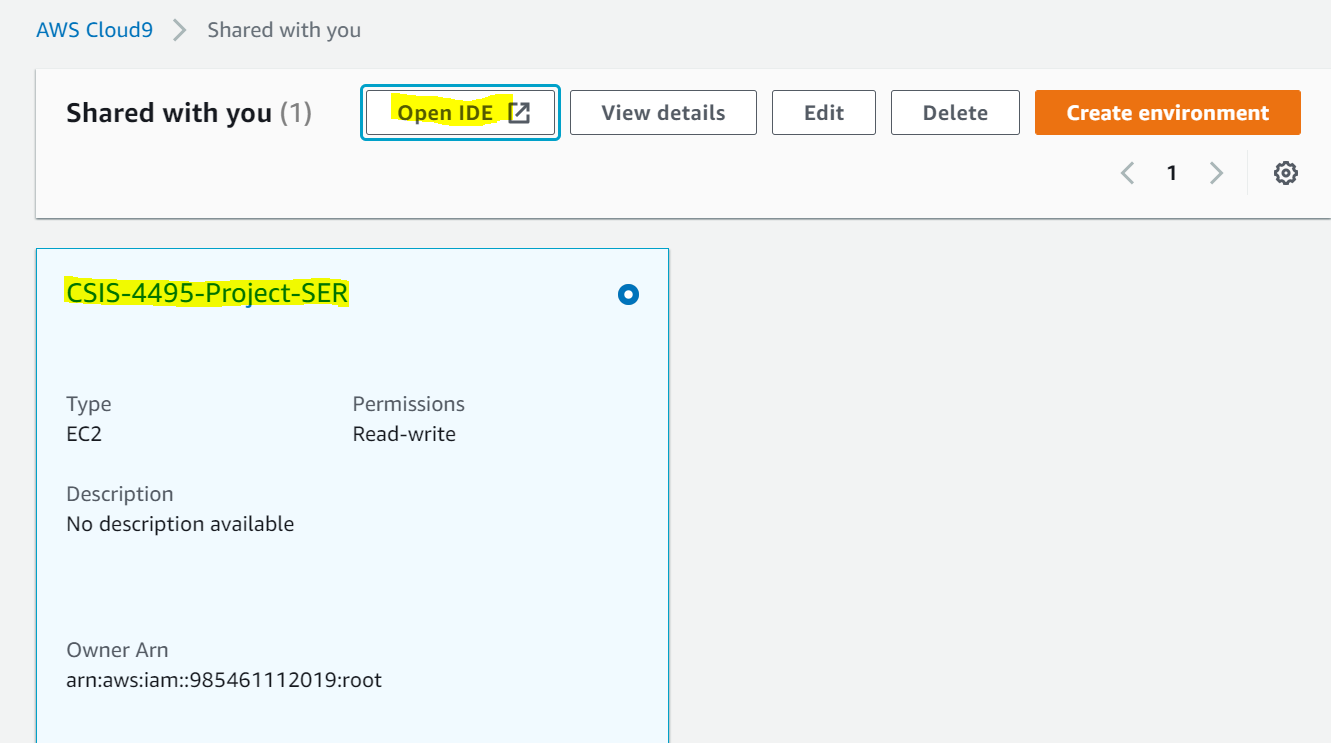
* After logging in, open the cloud9 service.



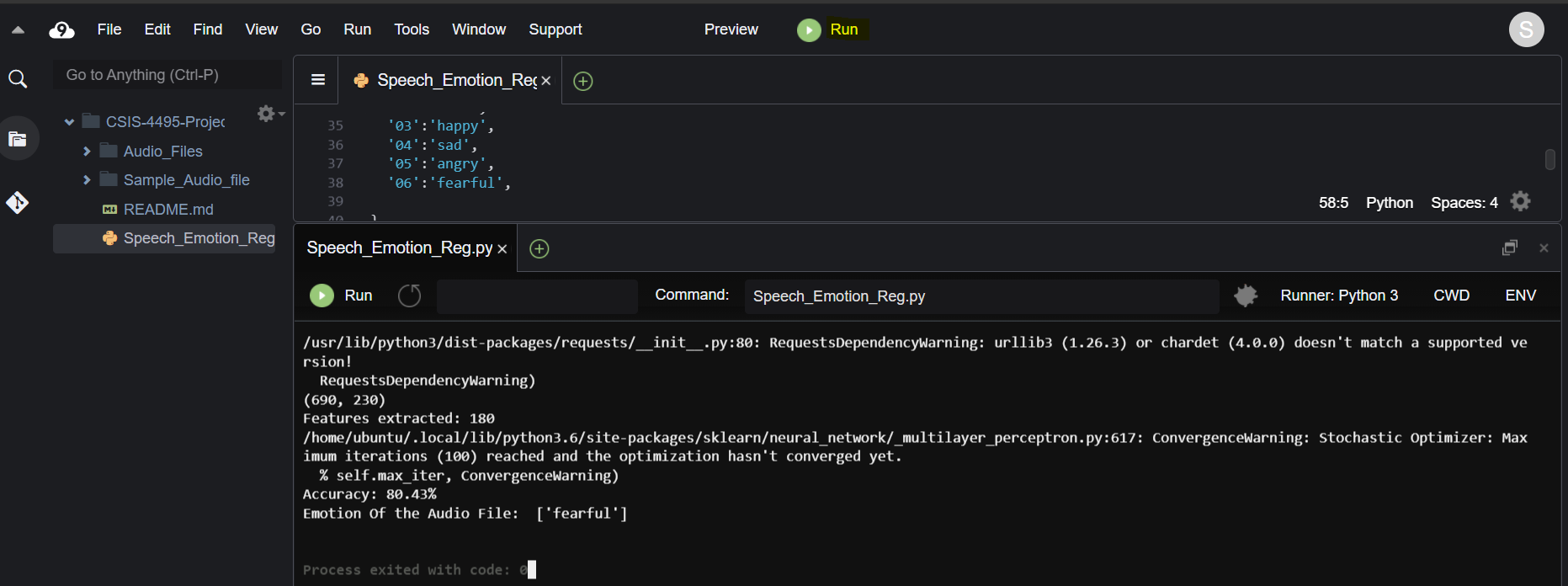
* Since the project was initially developed using root credentials it has been shared with this IAM user for security purposes.



* Kindly open the IDE



* Run the Speech Emotion Recognition python file for the output of this project.



1. **Appendix A Progress Report 3:**
2. **Abstract / Statement of the problem**

Speech Emotion Recognition, abbreviated as SER, is the demonstration of endeavoring to perceive human emotion and full of feeling states from speech. This is profiting by the way that voice frequently reflects basic emotion through tone and pitch. This is additionally the wonder that creatures like canines and ponies utilize to have the option to comprehend human emotion. SER is intense on the grounds that emotions are abstract and clarifying sound is testing.

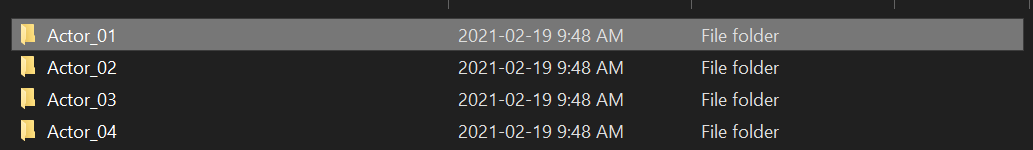
The purpose of the project is to build a machine learning model to determine the emotions of the song being played. The model trained in this project can help us determine the type of songs can be used to calm down a person and which one can make him happy.

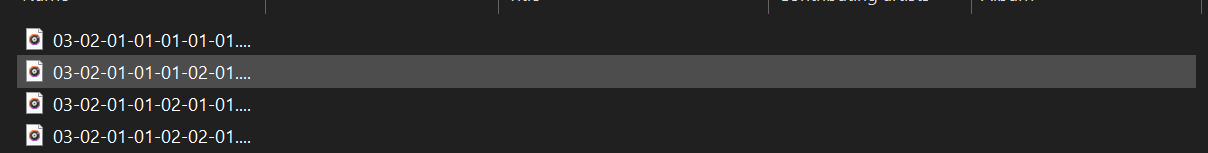
1. **Significance of the study**

The machine learning model trained in this project can be used the used by the in the mental health hospital to determine those songs that can make the patients happy and calm. This model can also be used by the different music company to suggest customers the songs depending on their mood.

1. **Project Implementation and timelines:**
2. **Things done so far:**

* The project is implemented using AWS Cloud9 coding environment. Cloud9 is a cloud-based integrated development environment (IDE) that lets you write, run, and debug your code with just a browser.
* Understanding the audio files extracted from the RAVDESS. There are various emotions recorded for around 24 actors which would be used to train the machine learning model to predict the emotions of the sample case.





Filename identifiers:

* + Modality (01 = full-AV, 02 = video-only, 03 = audio-only).
  + Vocal channel (01 = speech, 02 = song).
  + Emotion (01 = neutral, 02 = calm, 03 = happy, 04 = sad, 05 = angry, 06 = fearful, 07 = disgust, 08 = surprised).
  + Emotional intensity (01 = normal, 02 = strong). NOTE: There is no strong intensity for the 'neutral' emotion.
  + Statement (01 = "Kids are talking by the door", 02 = "Dogs are sitting by the door").
  + Repetition (01 = 1st repetition, 02 = 2nd repetition).
  + Actor (01 to 24. Odd numbered actors are male, even numbered actors are female).
* Install and Imported required library.

**Librosa** is a python package for music and sound analysis. It gives the structure blocks important to make music data recovery frameworks.

**SoundFile** can be used to read and write sound files.

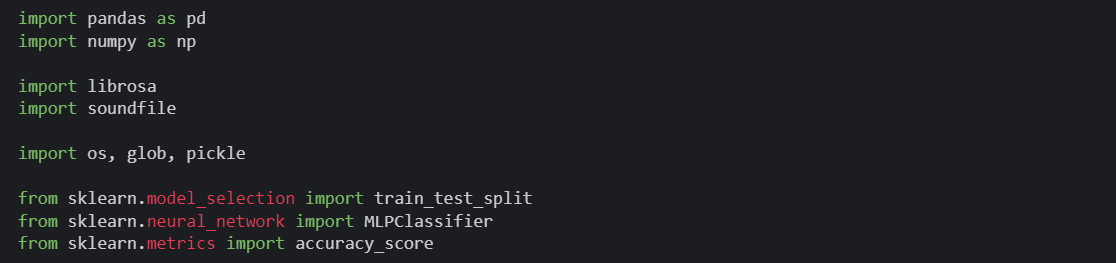
**Pandas** is a fast, powerful, flexible, and easy to use open-source data analysis and manipulation tool, built on top of the Python programming language.

**NumPy** is a Python library used for working with arrays.

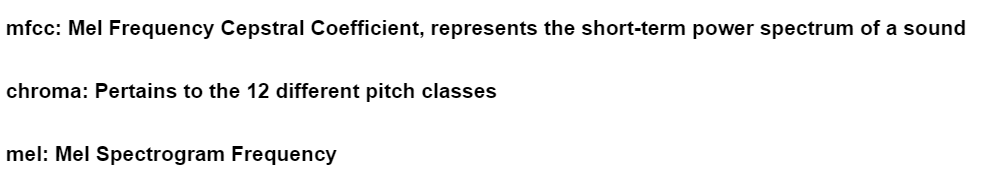
**Scikit-learn** is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering, and dimensionality reduction.

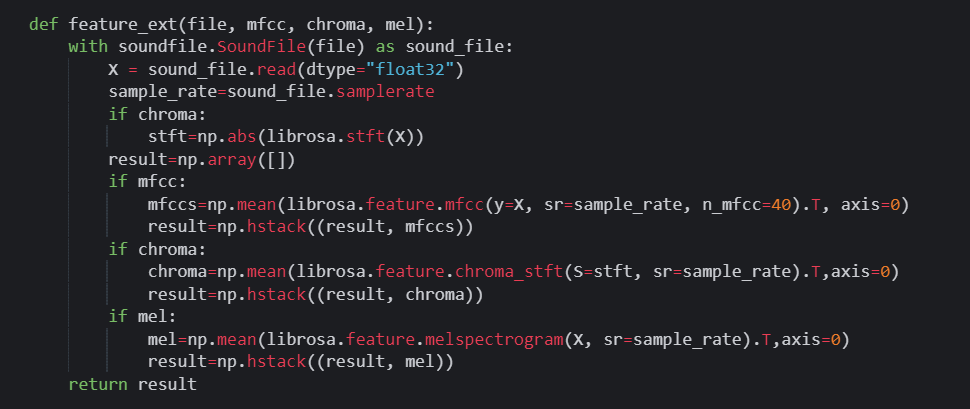
**MLP** is a type of artificial neural network (ANN). Simplest MLP consists of at least three layers of nodes: an input layer, a hidden layer, and an output layer.

In Scikit-learn “**MLP Classifier**” is available for Multilayer Perceptron (MLP) classification scenarios.



* Defined a function to extract the three features of the audio files in the data set. These features would help to create the machine learning model using MLP classifier that would predict the emotion of sample file.



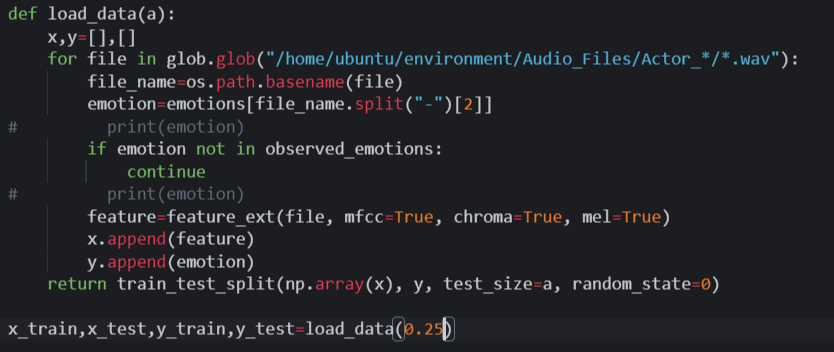


* Created a list to hold the emotions in shown in the files as well as an array of emotions that would be observed using the model.



* To create train and test dataset, there is a need to load the data (**load\_data(a)**) so I created a function which would access the file based on the filename identifiers. The features of the files are extracted using the feature extraction method (**feature\_ext(file, mfcc, chroma, mel)**) created earlier. While creating a train and test dataset the test size is set to 25% of the dataset.

Train and Test data:



* The size of train and test dataset is mentioned below:



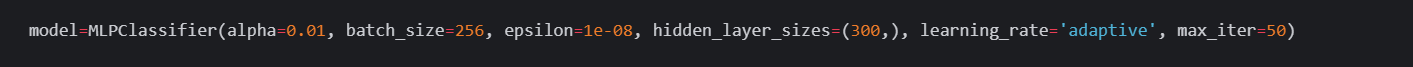


* The number of features extracted from all the files are:





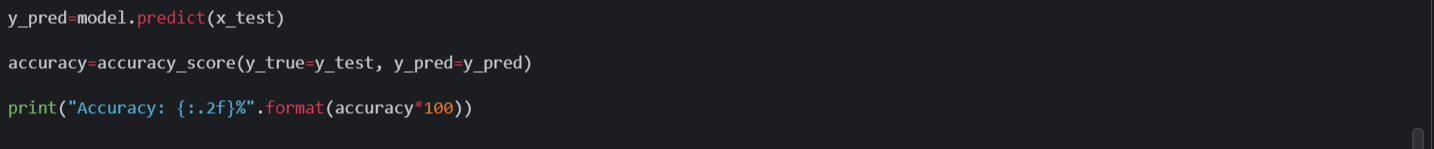
* The machine learning model is created using MLP Classifier which uses adaptive learning rate.



* The train dataset created earlier is fit into this model so that the model is trained, and we would be able to predict the emotions of the sample sound file.

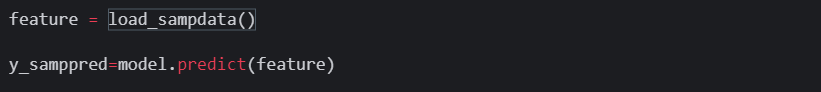


* Now a prediction is made on the test dataset so that we would be able to check the accuracy of the model.





* Now the sample file “**creature-sobbing-in-fear.wav**” using a method (**load\_sampdata()**) that extracts the feature and the features are then used to predict the emotions.



* After the prediction is made on the sample file we now know the emotions being resonated from the file which here seems to be a “**fearful**” emotion.





1. **Previous project implementation:**

Earlier the whole project was implemented using jupyter notebook but to deploy on the server as per industry standards the project was to be migrated to AWS cloud9 services.

1. **Pending implementations:**

Nothing

1. **Appendix B Progress Report 2:**
2. **Abstract / Statement of the problem**

Speech Emotion Recognition, abbreviated as SER, is the demonstration of endeavoring to perceive human emotion and full of feeling states from speech. This is profiting by the way that voice frequently reflects basic emotion through tone and pitch. This is additionally the wonder that creatures like canines and ponies utilize to have the option to comprehend human emotion. SER is intense on the grounds that emotions are abstract and clarifying sound is testing.

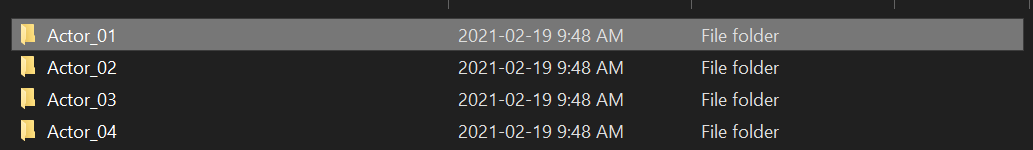
The purpose of the project is to build a machine learning model to determine the emotions of the song being played. The model trained in this project can help us determine the type of songs can be used to calm down a person and which one can make him happy.

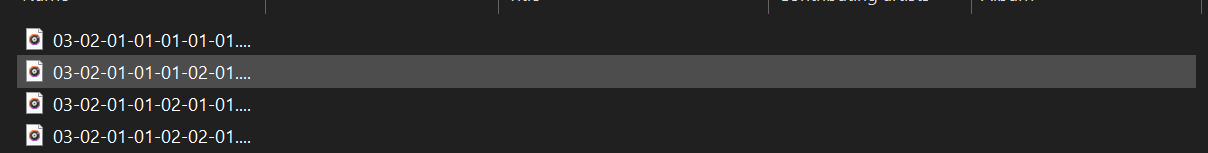
1. **Significance of the study**

The machine learning model trained in this project can be used the used by the in the mental health hospital to determine those songs that can make the patients happy and calm. This model can also be used by the different music company to suggest customers the songs depending on their mood.

1. **Project Implementation and timelines:**
2. **Things done so far:**

* The project is implemented using AWS Cloud9 coding environment. Cloud9 is a cloud-based integrated development environment (IDE) that lets you write, run, and debug your code with just a browser.
* Understanding the audio files extracted from the RAVDESS. There are various emotions recorded for around 24 actors which would be used to train the machine learning model to predict the emotions of the sample case.





Filename identifiers:

* + Modality (01 = full-AV, 02 = video-only, 03 = audio-only).
  + Vocal channel (01 = speech, 02 = song).
  + Emotion (01 = neutral, 02 = calm, 03 = happy, 04 = sad, 05 = angry, 06 = fearful, 07 = disgust, 08 = surprised).
  + Emotional intensity (01 = normal, 02 = strong). NOTE: There is no strong intensity for the 'neutral' emotion.
  + Statement (01 = "Kids are talking by the door", 02 = "Dogs are sitting by the door").
  + Repetition (01 = 1st repetition, 02 = 2nd repetition).
  + Actor (01 to 24. Odd numbered actors are male, even numbered actors are female).
* Install and Imported required library.

**Librosa** is a python package for music and sound analysis. It gives the structure blocks important to make music data recovery frameworks.

**SoundFile** can be used to read and write sound files.

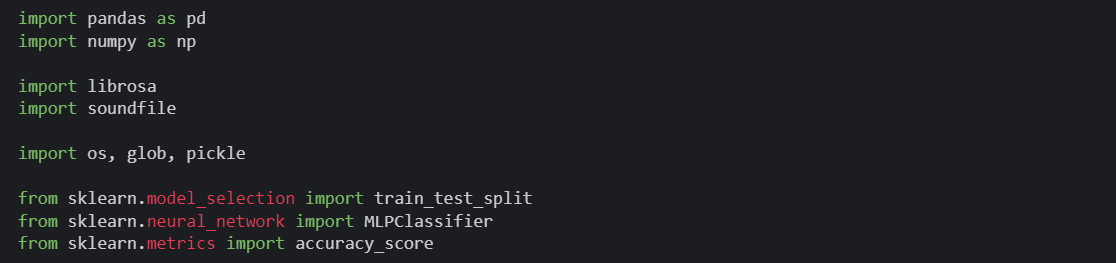
**Pandas** is a fast, powerful, flexible, and easy to use open-source data analysis and manipulation tool, built on top of the Python programming language.

**NumPy** is a Python library used for working with arrays.

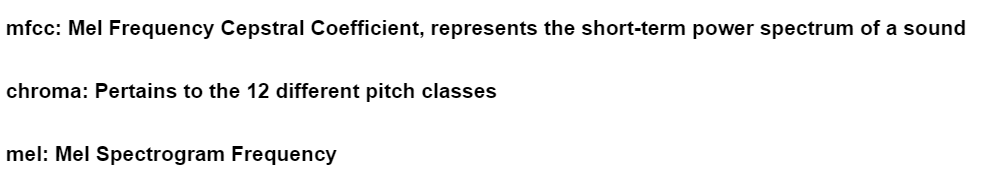
**Scikit-learn** is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering, and dimensionality reduction.

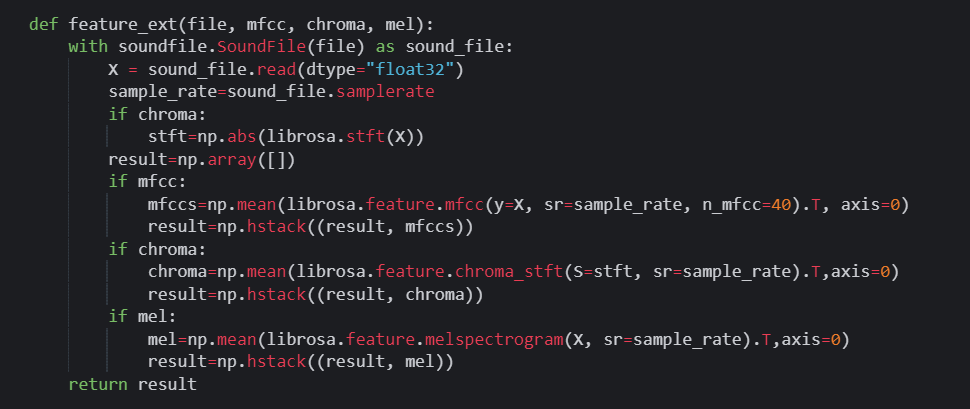
**MLP** is a type of artificial neural network (ANN). Simplest MLP consists of at least three layers of nodes: an input layer, a hidden layer, and an output layer.

In Scikit-learn “**MLP Classifier**” is available for Multilayer Perceptron (MLP) classification scenarios.



* Defined a function to extract the three features of the audio files in the data set. These features would help to create the machine learning model using MLP classifier that would predict the emotion of sample file.



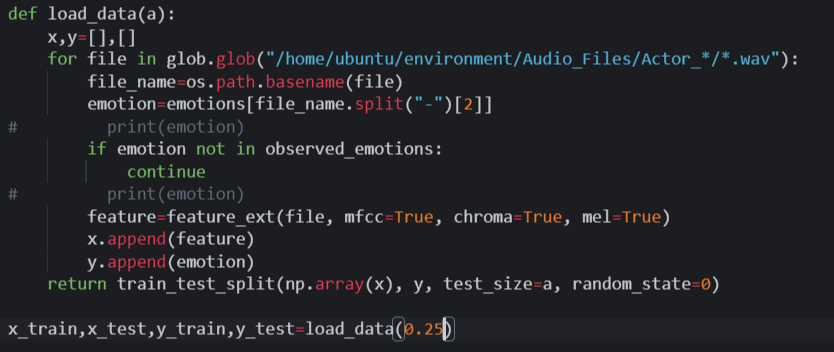


* Created a list to hold the emotions in shown in the files as well as an array of emotions that would be observed using the model.



* To create train and test dataset, there is a need to load the data (**load\_data(a)**) so I created a function which would access the file based on the filename identifiers. The features of the files are extracted using the feature extraction method (**feature\_ext(file, mfcc, chroma, mel)**) created earlier. While creating a train and test dataset the test size is set to 25% of the dataset.

Train and Test data:



* The size of train and test dataset is mentioned below:



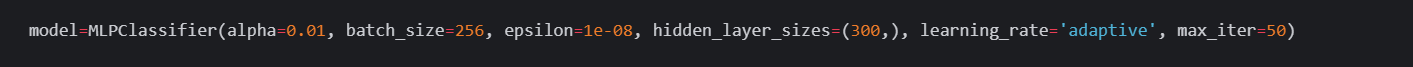


* The number of features extracted from all the files are:





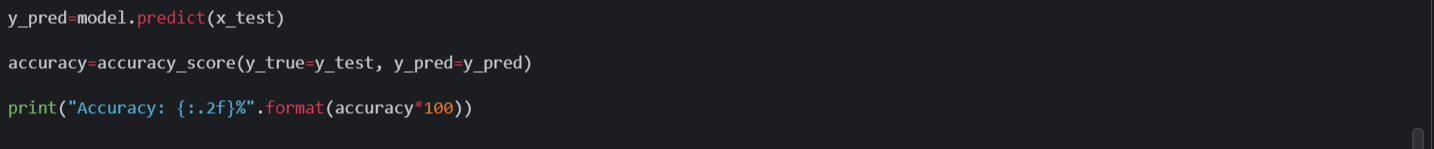
* The machine learning model is created using MLP Classifier which uses adaptive learning rate.



* The train dataset created earlier is fit into this model so that the model is trained, and we would be able to predict the emotions of the sample sound file.

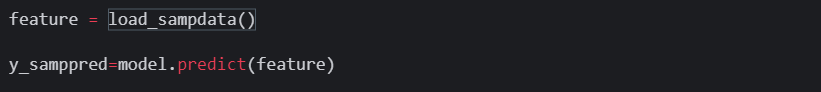


* Now a prediction is made on the test dataset so that we would be able to check the accuracy of the model.





* Now the sample file “**creature-sobbing-in-fear.wav**” using a method (**load\_sampdata()**) that extracts the feature and the features are then used to predict the emotions.



* After the prediction is made on the sample file we now know the emotions being resonated from the file which here seems to be a “**fearful**” emotion.



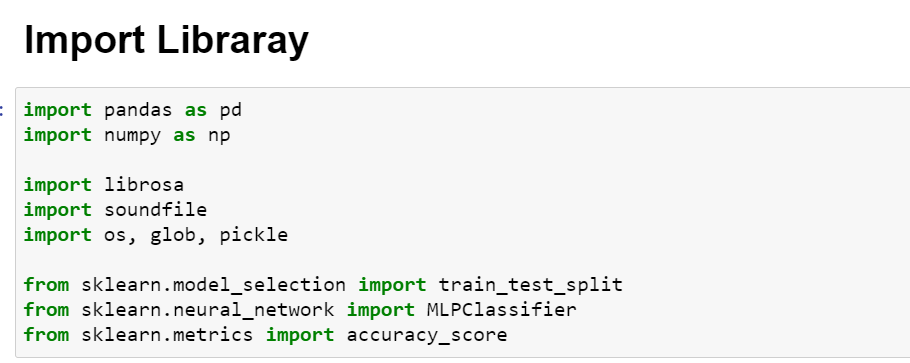


1. **Previous project implementation:**

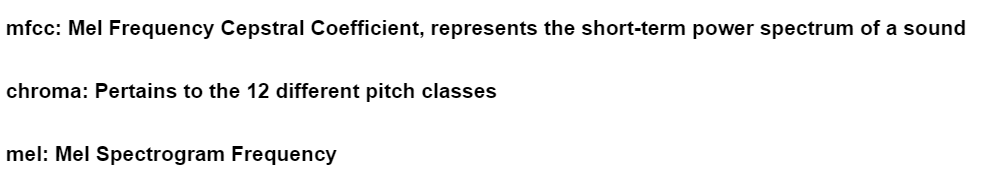
* Understanding the audio files provided. There are various emotions are recorded for around 24 actors which would be used to train the machine learning model to predict the emotions of the test case.
* Install and Imported required library.

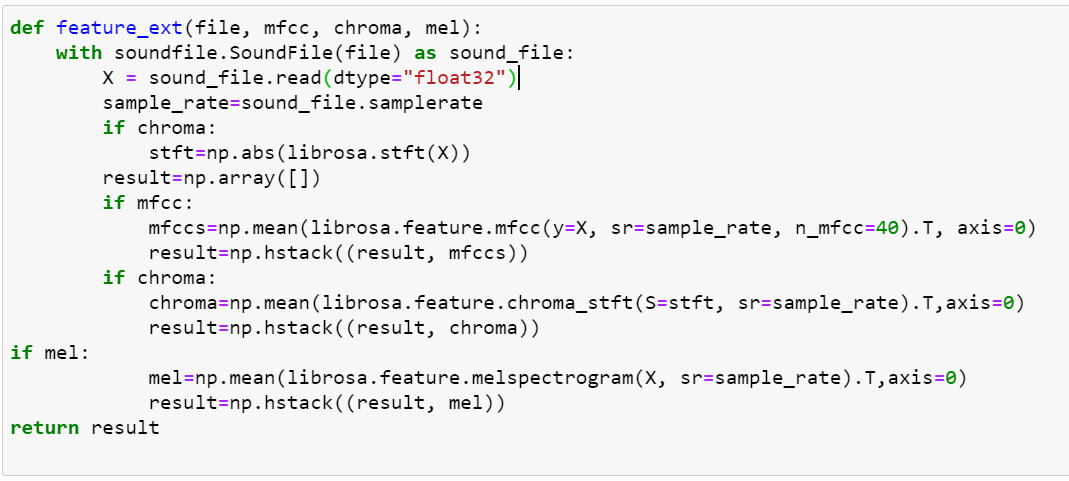
**Librosa** is a python package for music and sound analysis. It gives the structure blocks important to make music data recovery frameworks.

**SoundFile** can be used to read and write sound files.



* Defined a function to extract the three features of the audio files in the data set.





1. **Pending implementations:**

The project is still in data cleaning and loading phase and the model is to train and implemented.

1. **Appendix C Progress Report 1:**
2. **Abstract / Statement of the problem**

Speech Emotion Recognition, abbreviated as SER, is the demonstration of endeavoring to perceive human emotion and full of feeling states from speech. This is profiting by the way that voice frequently reflects basic emotion through tone and pitch. This is additionally the wonder that creatures like canines and ponies utilize to have the option to comprehend human emotion. SER is intense on the grounds that emotions are abstract and clarifying sound is testing.

The purpose of the project is to build a machine learning model to determine the emotions of the song being played. The model trained in this project can help us determine the type of songs can be used to clam down a person and which one can make him happy.

1. **Significance of the study**

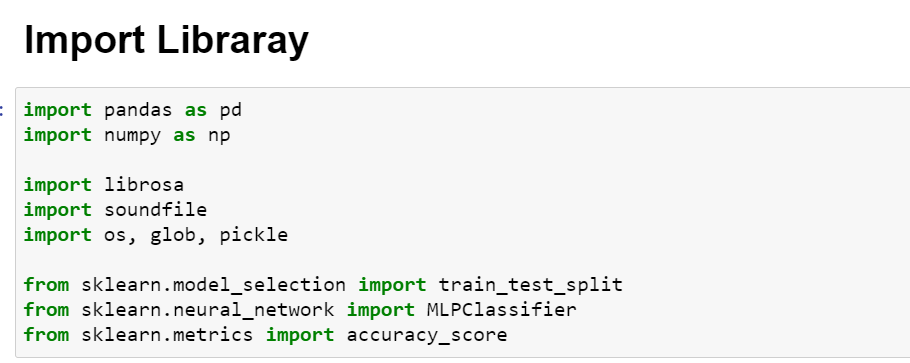
The machine learning model trained in this project can be used the used by the in the mental health hospital to determine those songs that can make the patients happy and calm. This model can also be used by the different music company to suggest customers the songs depending on their mood.

1. **Project Implementation and timelines:**
2. **Things done so far:**

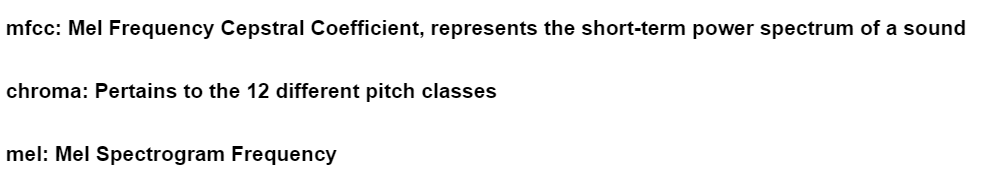
* Understanding the audio files provided. There are various emotions are recorded for around 24 actors which would be used to train the machine learning model to predict the emotions of the test case.
* Install and Imported required library.

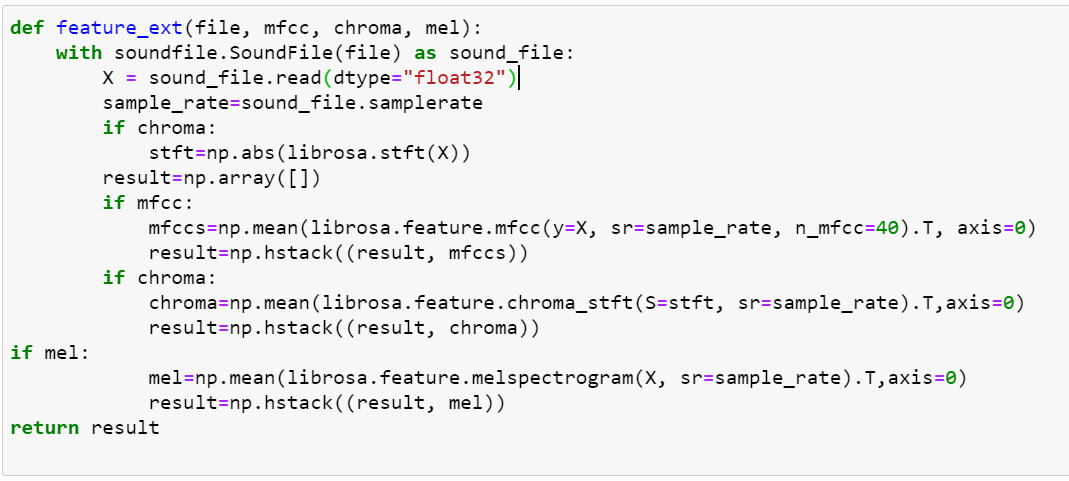
**Librosa** is a python package for music and sound analysis. It gives the structure blocks important to make music data recovery frameworks.

**SoundFile** can be used to read and write sound files.



* Defined a function to extract the three features of the audio files in the data set.





1. **Previous project implementation**

None

1. **Pending implementations**

The project is still me implemented and has its various points still in progress.

1. **Proposed Revisions:**

None