Now, add the code to create the list of all the images in `images_path` and edit the call to the `brain()` function to pass the new arguments that are needed by your module:

```
    imgur_handler.img_list_gen(images_path)
    brain(name, speech_text, music_path, city_name, city_code, proxy_
    username, proxy_password, consumer_key, consumer_secret, access_token,
    access_token_secret, client_id, client_secret, images_path)
```

Note that the module and the function you add here have not yet been constructed.

Create a file named `imgur_handler.py` in the `GreyMatter` folder, and type the following code in it:

```
import os
import sqlite3
from datetime import datetime

from imgurpython import ImgurClient

from SenseCells.tts import tts

def img_list_gen(images_path):

    image_list = []
    for root, dirs, files in os.walk(images_path):
        for filename in files:
            if os.path.splitext(filename)[1] == ".tiff" or os.path.
            splitext(filename)[1] == ".png" or os.path.splitext(filename)[1]
            == ".gif" or os.path.splitext(filename)[1] == ".jpg":
                image_list.append(os.path.join(root, filename.lower()))
    return image_list

def image_uploader(speech_text, client_id, client_secret, images_path):

    words_of_message = speech_text.split()
    words_of_message.remove('upload')
    cleaned_message = ' '.join(words_of_message)
    image_listing = img_list_gen(images_path)

    client = ImgurClient(client_id, client_secret)

    for i in range(0, len(image_listing)):
        if cleaned_message in image_listing[i]:
            result = client.upload_from_path(image_listing[i], config=None,
            anon=True)
```