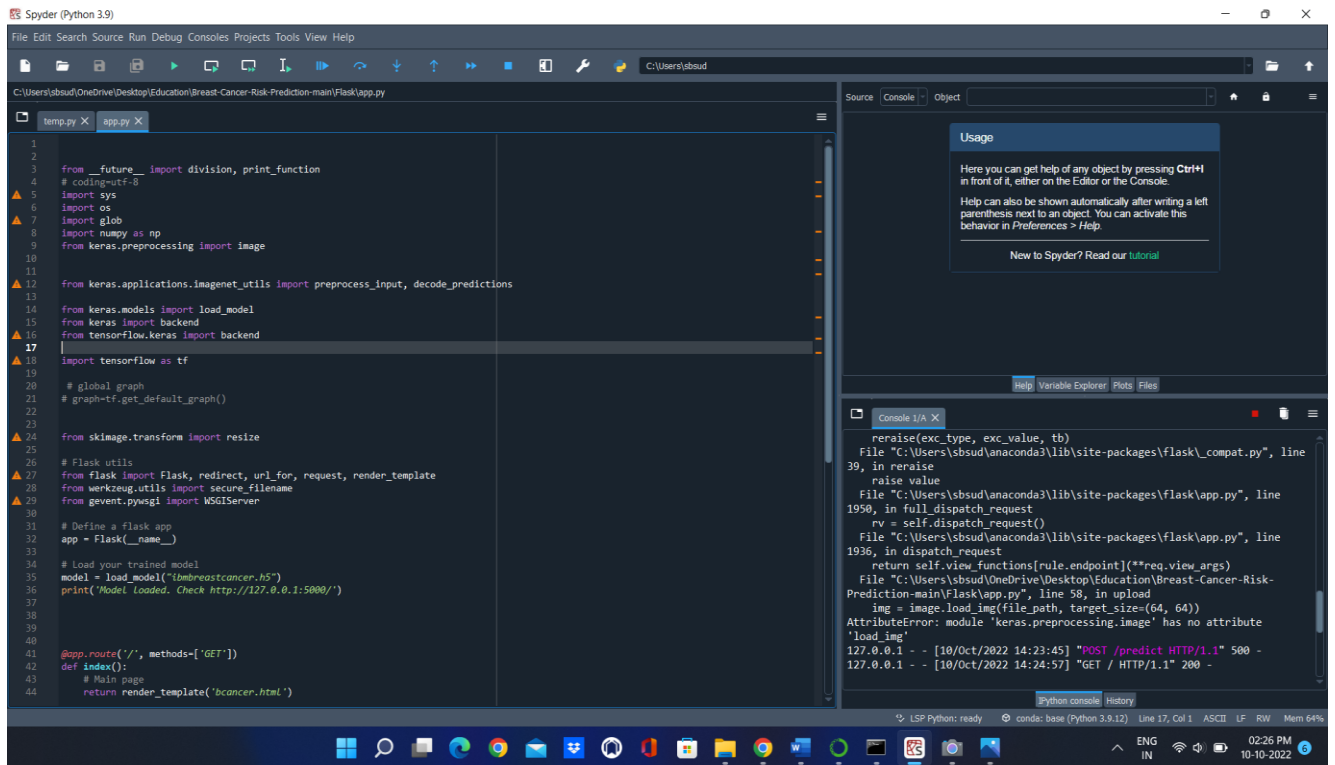


# Deep Learning Techniques for Breast Cancer Risk Prediction using IBM Cloud

## OUTPUTS:



The screenshot displays the Spyder Python IDE interface. The main editor window shows a Python script named `app.py` which is a Flask web application. The code includes imports for `Flask`, `Image`, `load_model`, and `image`. It defines a `predict` endpoint that takes an image file path as input, processes it using a pre-trained model, and returns the prediction. The application is configured to run on `0.0.0.0:5000`.

The console window at the bottom right shows the output of the application. It displays the standard Flask startup message, followed by a `POST` request to `/predict` from `127.0.0.1` with a status of `500`. This is followed by a `GET` request to `/` from `127.0.0.1` with a status of `200`.

```
1 from _future_ import division, print_function
2
3 # coding=utf-8
4 import sys
5 import os
6 import glob
7 import numpy as np
8 from keras.preprocessing import image
9
10
11
12 from keras.applications.imagenet_utils import preprocess_input, decode_predictions
13
14 from keras.models import load_model
15 from keras import backend
16 from tensorflow.keras import backend
17
18 import tensorflow as tf
19
20 # global graph
21 # graph=tf.get_default_graph()
22
23
24 from skimage.transform import resize
25
26 # Flask utils
27 from flask import Flask, redirect, url_for, request, render_template
28 from werkzeug.utils import secure_filename
29 from gevent.pywsgi import WSGIServer
30
31 # Define a Flask app
32 app = Flask(__name__)
33
34 # Load your trained model
35 model = load_model("imbreastcancer.h5")
36 print('Model Loaded. Check http://127.0.0.1:5000/')
37
38
39
40
41 @app.route('/', methods=['GET'])
42 def index():
43     # Main page
44     return render_template('bcancer.html')
```

Usage

Here you can get help of any object by pressing **Ctrl+H** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in **Preferences > Help**.

[New to Spyder? Read our tutorial](#)

Console 1/A X

```
reraise(exc_type, exc_value, tb)
File "C:\Users\sbsud\anaconda3\lib\site-packages\flask\compat.py", line
39, in reraise
    raise value
File "C:\Users\sbsud\anaconda3\lib\site-packages\flask\app.py", line
1950, in full_dispatch_request
    rv = self.dispatch_request()
File "C:\Users\sbsud\anaconda3\lib\site-packages\flask\app.py", line
1936, in dispatch_request
    return self.view_functions[rule.endpoint](**req.view_args)
File "C:\Users\sbsud\OneDrive\Desktop\Education\Breast-Cancer-Risk-
Prediction-main\Flask\app.py", line 58, in upload
    img = image.load_img(file_path, target_size=(64, 64))
AttributeError: module 'keras.preprocessing.image' has no attribute
'load_img'
127.0.0.1 - - [10/Oct/2022 14:23:45] "POST /predict HTTP/1.1" 500 -
127.0.0.1 - - [10/Oct/2022 14:24:57] "GET / HTTP/1.1" 200 -
```

Python console History

LSP Python: ready conda: base (Python 3.9.12) Line 17, Col 1 ASCII LF RW Mem 64%

ENG IN 02:26 PM 10-10-2022

