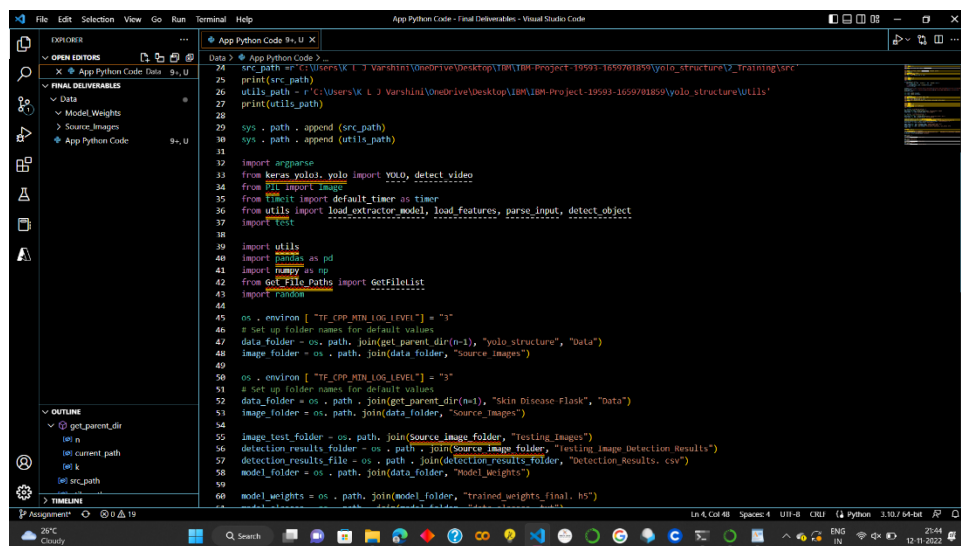


PROJECT NAME : AI-based localization and classification of skin disease with erythema.

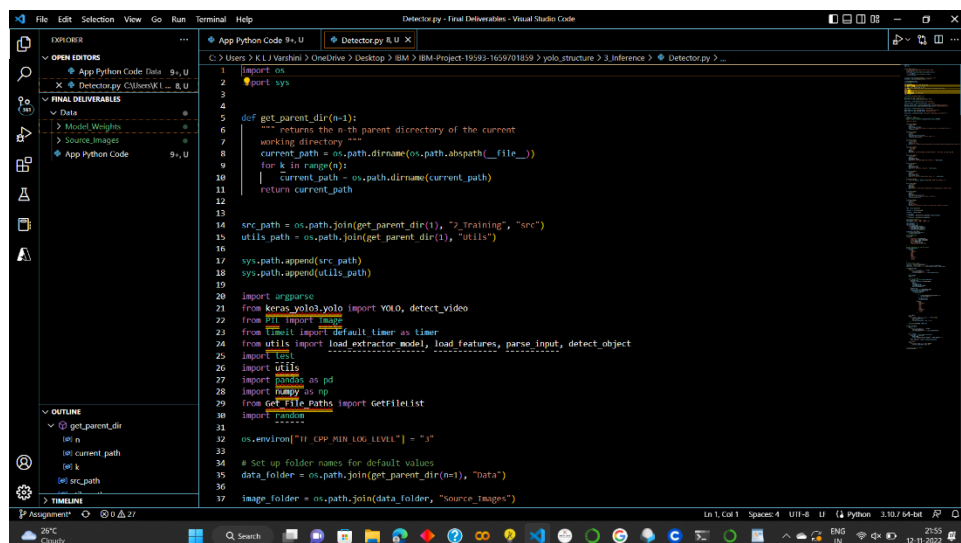
TEAM ID : PNT2022TMID07139

PYTHON CODE WEB PAGE



This screenshot shows the Visual Studio Code editor with a Python file named 'App Python Code - Final Deliverables'. The code is a script for training a YOLO model. It includes imports for 'os', 'sys', 'argparse', 'keras', 'yolo', 'PIL', 'timeit', 'utils', 'load_extractor_model', 'load_features', 'parse_input', 'detect_object', 'utils', 'pandas', 'numpy', 'get_file_paths', 'get_file_list', 'random', and 'env'. The script sets up folder names for 'data', 'image', 'model', and 'weights'. It then defines a function 'get_parent_dir' to find the parent directory of the current file. The main logic of the script is to load the YOLO model, parse the input arguments, and detect objects in the source images.

```
24 src_path = r'C:\Users\K L J Varshini\OneDrive\Desktop\IBM\IBM-Project-19593-1659701859\yolo_structure\2_training\src'
25 print(src_path)
26 utils_path = r'C:\Users\K L J Varshini\OneDrive\Desktop\IBM\IBM-Project-19593-1659701859\yolo_structure\utils'
27 print(utils_path)
28
29 sys.path.append(src_path)
30 sys.path.append(utils_path)
31
32 import argparse
33 from keras.yolo3.yolo import YOLO, detect_video
34 from PIL import Image
35 from timeit import default_timer as timer
36 from utils import load_extractor_model, load_features, parse_input, detect_object
37 import test
38
39 import utils
40 import pandas as pd
41 import numpy as np
42 from get_file_paths import GetFileList
43 import random
44
45 os.environ["TF_CPP_MIN_LOG_LEVEL"] = "3"
46 # Set up folder names for default values
47 data_folder = os.path.join(get_parent_dir(n=1), "yolo_structure", "data")
48 image_folder = os.path.join(data_folder, "source_images")
49
50 os.environ["TF_CPP_MIN_LOG_LEVEL"] = "3"
51 # Set up folder names for default values
52 data_folder = os.path.join(get_parent_dir(n=1), "Skin Disease flask", "data")
53 image_folder = os.path.join(data_folder, "source_images")
54
55 image_test_folder = os.path.join(source_image_folder, "testing_images")
56 detection_results_folder = os.path.join(source_image_folder, "testing_image_detection_results")
57 detection_results_file = os.path.join(detection_results_folder, "detection_results.csv")
58 model_folder = os.path.join(data_folder, "model_weights")
59
60 model_weights = os.path.join(model_folder, "trained_weights_final.h5")
```



This screenshot shows the Visual Studio Code editor with a Python file named 'Detector - Final Deliverables'. The code is a script for detecting objects in source images. It includes imports for 'os', 'sys', 'argparse', 'keras', 'yolo', 'PIL', 'timeit', 'utils', 'load_extractor_model', 'load_features', 'parse_input', 'detect_object', 'utils', 'pandas', 'numpy', 'get_file_paths', 'get_file_list', 'random', and 'env'. The script sets up folder names for 'data', 'image', 'model', and 'weights'. It then defines a function 'get_parent_dir' to find the parent directory of the current file. The main logic of the script is to load the YOLO model, parse the input arguments, and detect objects in the source images.

```
1 import os
2 import sys
3
4
5 def get_parent_dir(n=1):
6     """ returns the n-th parent directory of the current
7     working directory """
8     current_path = os.path.dirname(os.path.abspath(__file__))
9     for k in range(n):
10         current_path = os.path.dirname(current_path)
11     return current_path
12
13
14 src_path = os.path.join(get_parent_dir(1), "2_training", "src")
15 utils_path = os.path.join(get_parent_dir(1), "utils")
16
17 sys.path.append(src_path)
18 sys.path.append(utils_path)
19
20 import argparse
21 from keras.yolo3.yolo import YOLO, detect_video
22 from PIL import Image
23 from timeit import default_timer as timer
24 from utils import load_extractor_model, load_features, parse_input, detect_object
25 import test
26
27 import utils
28 import pandas as pd
29 import numpy as np
30 from get_file_paths import GetFileList
31 import random
32
33 os.environ["TF_CPP_MIN_LOG_LEVEL"] = "3"
34
35 # Set up folder names for default values
36 data_folder = os.path.join(get_parent_dir(n=1), "data")
37
38 image_folder = os.path.join(data_folder, "source_images")
```