

 DR. D. Y. PATIL INSTITUTE OF ENGINEERING, MANAGEMENT AND RESEARCH	<b>Dr D Y Patil Pratishthan's Dr. D.Y. Patil Institute of Engineering, Management and Research, Akurdi, Pune</b>	DI No.: <b>ACAD/DI/72</b>
<b>Academic Year:</b> 2025-26	<b>Weekly Report Format for Internship</b>	<b>Revision : 00 Dated :</b> 20/11/2019
<b>Term - II</b>	<b>Department : Computer Engineering</b>	<b>Date of Preparation :</b> 14/02/2026

### **Weekly Report Format for Internship**

**Roll No:** 23101

**Name of the Student:** Ruhan Saad Dave

**Name of the Company:** CODTECH IT SOLUTIONS PVT.LTD

**Domain of Internship:** Artificial Intelligence

**Name of the Internal Guide:** Akanksha Kulkarni

**Name of the External Guide:** Neela Santhosh Kumar

**Duration of Internship:** 03 Jan 2026 to 14 Feb 2026

**Week-IV Dates:** 24/01/2026 to 31/01/2026

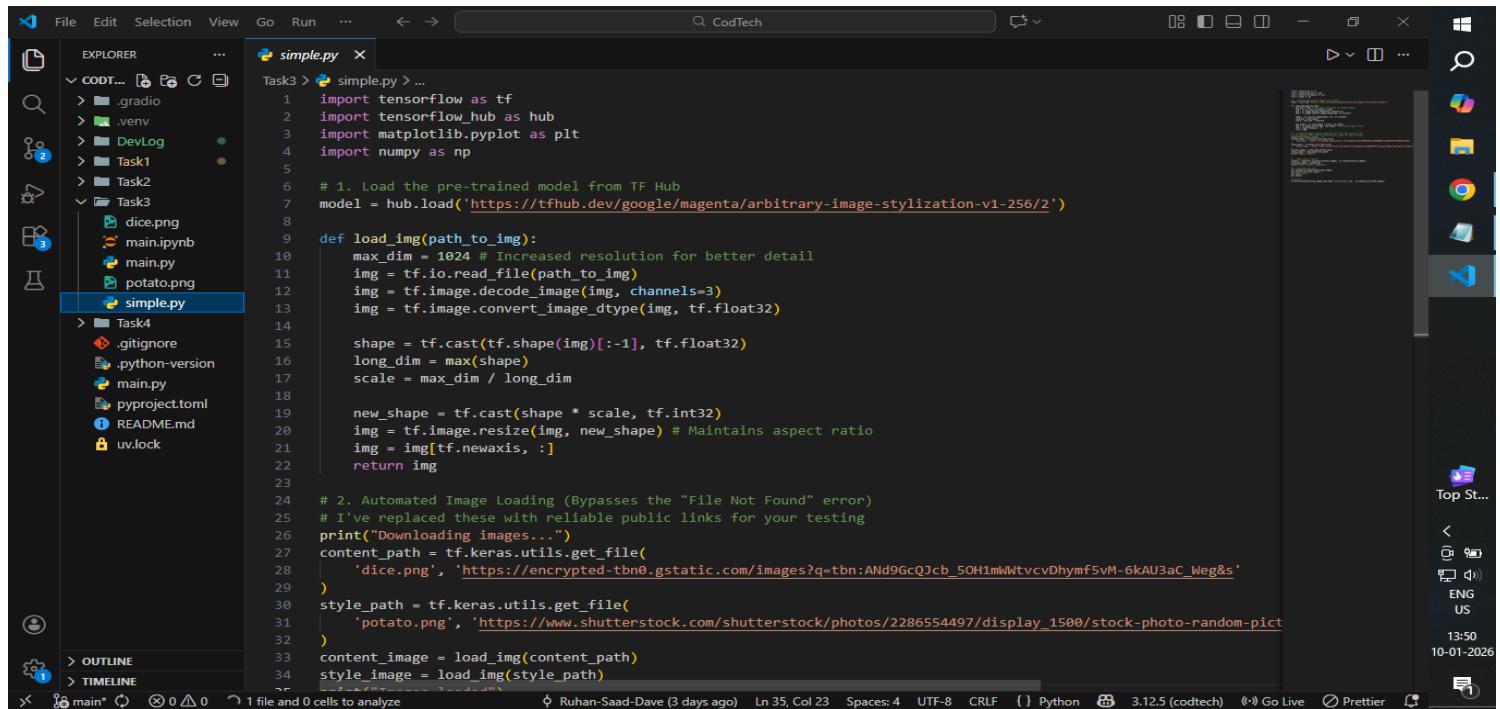
#### **Description of Work Done Till Date (Week 4):**

**Task 3 - Neural Style Transfer:**

- Focused entirely on the Neural Style Transfer task due to its complexity.
- Installed and configured dependencies, including ‘tensorflow’, ‘tensorflow\_hub’, and ‘matplotlib’.
- Researched and selected a pre-trained model from TensorFlow Hub suitable for style transfer.
- Developed a script to load a content image and a style image.
- Successfully implemented the logic to apply the artistic style from the style image onto the content image, generating a new, stylized picture.
- Addressed challenges related to image processing and model execution.

 <b>DR D. Y. PATIL INSTITUTE OF ENGINEERING, MANAGEMENT AND RESEARCH</b>	<b>Dr D Y Patil Pratishthan's Dr. D.Y. Patil Institute of Engineering, Management and Research, Akurdi, Pune</b>	<b>DI No.: ACAD/DI/72</b>
<b>Academic Year: 2025-26</b>	<b>Weekly Report Format for Internship</b>	<b>Revision : 00 Dated : 20/11/2019</b>
<b>Term - II</b>	<b>Department : Computer Engineering</b>	<b>Date of Preparation : 14/02/2026</b>

## Supporting Documents:

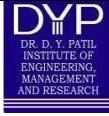


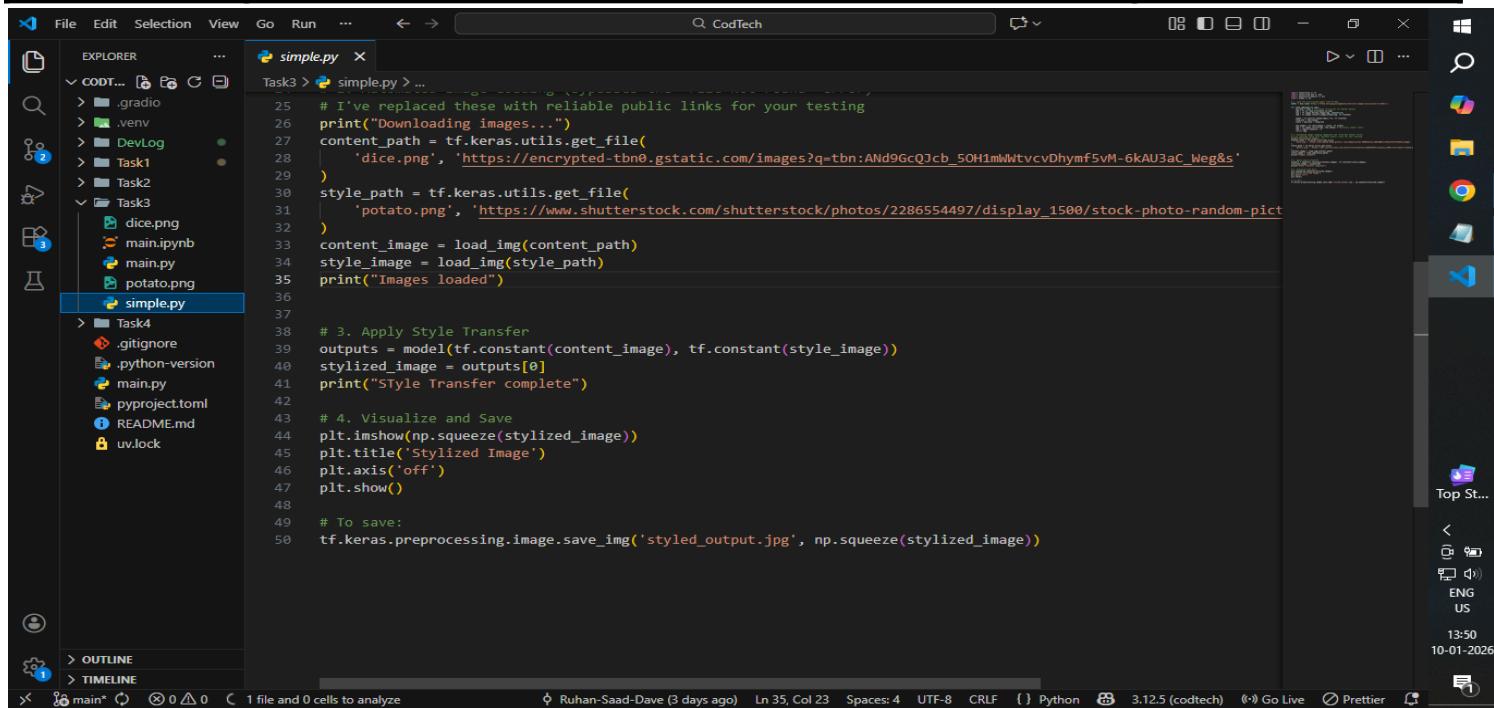
```

File Edit Selection View Go Run ... < > Q CodTech
EXPLORER simple.py ...
Task3 > simple.py > ...
1 import tensorflow as tf
2 import tensorflow_hub as hub
3 import matplotlib.pyplot as plt
4 import numpy as np
5
6 # 1. Load the pre-trained model from TF Hub
7 model = hub.load('https://tfhub.dev/google/magenta/arbitrary-image-stylization-v1-256/2')
8
9 def load_img(path_to_img):
10     max_dim = 1024 # Increased resolution for better detail
11     img = tf.io.read_file(path_to_img)
12     img = tf.image.decode_image(img, channels=3)
13     img = tf.image.convert_image_dtype(img, tf.float32)
14
15     shape = tf.cast(tf.shape(img)[::-1], tf.float32)
16     long_dim = max(shape)
17     scale = max_dim / long_dim
18
19     new_shape = tf.cast(shape * scale, tf.int32)
20     img = tf.image.resize(img, new_shape) # Maintains aspect ratio
21     img = img[tf.newaxis, :]
22     return img
23
24 # 2. Automated Image Loading (Bypasses the "File Not Found" error)
25 # I've replaced these with reliable public links for your testing
26 print("Downloading images...")
27 content_path = tf.keras.utils.get_file(
28     'dice.png', 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQJcb_5OH1mWtvcvDhymf5vM-6kAU3aC_lweg&s'
29 )
30 style_path = tf.keras.utils.get_file(
31     'potato.png', 'https://www.shutterstock.com/shutterstock/photos/2286554497/display_1500/stock-photo-random-pict'
32 )
33 content_image = load_img(content_path)
34 style_image = load_img(style_path)

```

Ruhan-Saad-Dave (3 days ago) Ln 35, Col 23 Spaces: 4 UTF-8 CRLF {} Python 3.12.5 (codtech) ⚡ Go Live ⚡ Prettier

 <b>DR D. Y. PATIL INSTITUTE OF ENGINEERING, MANAGEMENT AND RESEARCH</b>	<b>Dr D Y Patil Pratishthan's Dr. D.Y. Patil Institute of Engineering, Management and Research, Akurdi, Pune</b>	<b>DI No.: ACAD/DI/72</b>
<b>Academic Year: 2025-26</b>	<b>Weekly Report Format for Internship</b>	<b>Revision : 00 Dated : 20/11/2019</b>
<b>Term - II</b>	<b>Department : Computer Engineering</b>	<b>Date of Preparation : 14/02/2026</b>

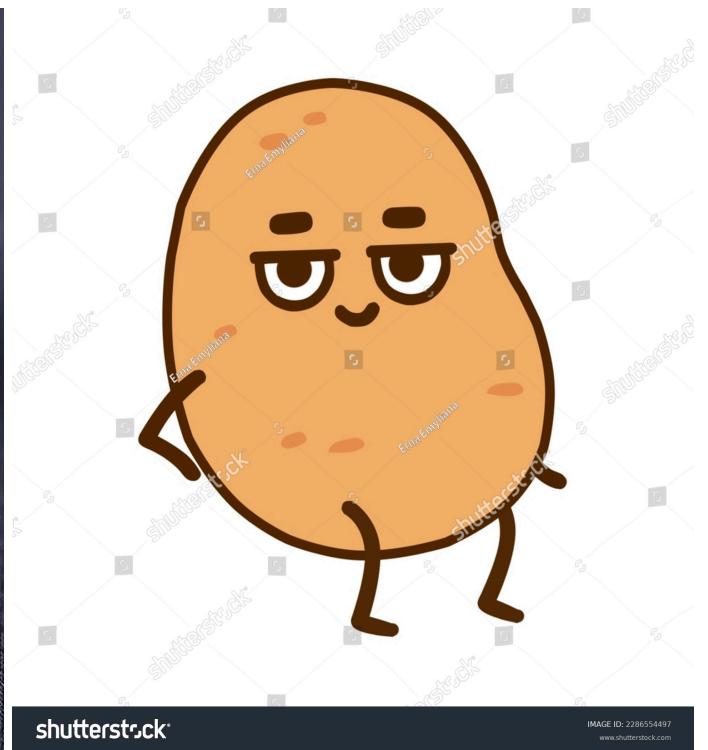


```

simple.py
25 # I've replaced these with reliable public links for your testing
26 print("Downloading images...")
27 content_path = tf.keras.utils.get_file(
28     'dice.png', 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQJcb_5OH1mWtvcvDhymf5vM-6kAU3aC_Weg&s'
29 )
30 style_path = tf.keras.utils.get_file(
31     'potato.png', 'https://www.shutterstock.com/photos/2286554497/display_1500/stock-photo-random-pict'
32 )
33 content_image = load_img(content_path)
34 style_image = load_img(style_path)
35 print("Images loaded")
36
37
38 # 3. Apply Style Transfer
39 outputs = model(tf.constant(content_image), tf.constant(style_image))
40 stylized_image = outputs[0]
41 print("Style Transfer complete")
42
43 # 4. Visualize and Save
44 plt.imshow(np.squeeze(stylized_image))
45 plt.title('Stylized Image')
46 plt.axis('off')
47 plt.show()
48
49 # To save:
50 tf.keras.preprocessing.image.save_img('stylized_output.jpg', np.squeeze(stylized_image))

```

The screenshot shows a Jupyter Notebook interface with a dark theme. The left sidebar displays a file tree with files like .gradle, .venv, DevLog, Task1, Task2, Task3 (containing dice.png, main.ipynb, main.py, potato.png), simple.py, Task4, .gitignore, .python-version, main.py, pyproject.toml, README.md, and uv.lock. The main area shows the code for 'simple.py'. The bottom status bar indicates the code was run by Ruhan-Saad-Dave 3 days ago, with 35 lines, 23 columns, 4 spaces, and UTF-8 encoding. The version is 3.12.5 (codtech). The right sidebar shows system status like battery level, signal strength, and date/time (13:50, 10-01-2026).



<b>DYP</b> DR. D. Y. PATIL INSTITUTE OF ENGINEERING, MANAGEMENT AND RESEARCH	<b>Dr D Y Patil Pratishthan's Dr. D.Y. Patil Institute of Engineering, Management and Research, Akurdi, Pune</b>	DI No.: <b>ACAD/DI/72</b>
<b>Academic Year:</b> 2025-26	<b>Weekly Report Format for Internship</b>	<b>Revision : 00 Dated :</b> <b>20/11/2019</b>
<b>Term - II</b>	<b>Department : Computer Engineering</b>	<b>Date of Preparation :</b> <b>14/02/2026</b>

Stylized Image



**Student Sign**

**Internal Guide Sign**