

## **Advanced Python Programming**

### **Lab Exercise-4**

---

Name: **Tushar Mahajan**

Class: **3 MCA A**

Register No.: **2447156**

---

***Question:***

Identify an interesting StreamlitApp from the given gallery and customize it.  
Create a word document with the original screen and customized screen.

Streamlit Templates

<https://streamlit.io/gallery>

---

***Changes Made:***

So, I have added CSS to make it appear a bit more beautiful.

Now coming on functionalities:

- Multiple images can be added
  - Not just removing background, now you can add new background, blur the previous one or even put a solid color as background.
  - Also now you can add AI filters in it (like grey scale, sepia, blur and cartoon).
- 

***Code:***

```
import streamlit as st
from rembg import remove
from PIL import Image, ImageFilter, ImageOps
import numpy as np
from io import BytesIO

# Page Configuration
st.set_page_config(layout="wide", page_title="Advanced AI Background Remover")

# Custom CSS Styling
st.markdown("""
<style>
    body { background-color: #f4f4f4; font-family: 'Arial', sans-serif; }
    .main-title { font-size: 40px; text-align: center; color: #ff6600; font-weight: bold; }
    .sub-title { font-size: 18px; text-align: center; color: #666; }
    .sidebar-title { font-size: 22px; font-weight: bold; color: #333; }
    .stButton>button { background-color: #ff6600; color: white; border-radius: 10px; font-size: 18px; font-weight: bold; padding: 10px; }
</style>
""", unsafe_allow_html=True)

# Header
st.markdown('<p class="main-title">✨ AI Background Remover & Editor ✨</p>',
unsafe_allow_html=True)
st.markdown('<p class="sub-title">Upload images, remove backgrounds, and apply cool effects!</p>',
unsafe_allow_html=True)
st.markdown("---")
```

```

# Sidebar Section
st.sidebar.markdown('<p class="sidebar-title">Upload & Settings</p>', unsafe_allow_html=True)
st.sidebar.write("⚠️ Upload multiple images to process")

# Maximum file size
MAX_FILE_SIZE = 5 * 1024 * 1024 # 5MB

# Function to convert image for download
def convert_image(img):
    buf = BytesIO()
    img.save(buf, format="PNG")
    byte_im = buf.getvalue()
    return byte_im

# Function to process image
def process_image(image, bg_option, bg_upload):
    original = Image.open(image).convert("RGBA")

    with col1:
        st.markdown("### 📸 Original Image")
        st.image(original, use_column_width=True)
        st.write(f"**Resolution:** {original.size[0]} x {original.size[1]}")

    # Remove Background
    with st.spinner("Removing background..."):
        processed = remove(original)

    # Replace Background if chosen
    if bg_option == "Solid Color":
        color = st.sidebar.color_picker("Pick a Background Color", "#ffffff")
        new_bg = Image.new("RGBA", processed.size, color)
        processed = Image.alpha_composite(new_bg, processed)

    elif bg_option == "Upload Background" and bg_upload is not None:
        new_bg = Image.open(bg_upload).convert("RGBA").resize(processed.size)
        processed = Image.alpha_composite(new_bg, processed)

    with col2:
        st.markdown("### 🎨 Processed Image")
        st.image(processed, use_column_width=True)

    # Apply Filters
    filter_choice = st.sidebar.selectbox("🖌️ Apply Filter", ["None", "Grayscale", "Sepia", "Blur", "Cartoon"])
    if filter_choice == "Grayscale":
        processed = ImageOps.grayscale(processed)
    elif filter_choice == "Sepia":
        sepia_filter = np.array(processed)
        sepia_filter = np.dot(sepia_filter[...,:3], [[0.393, 0.769, 0.189], [0.349, 0.686, 0.168], [0.272, 0.534, 0.131]])
        sepia_filter = np.clip(sepia_filter, 0, 255)
        processed = Image.fromarray(sepia_filter.astype('uint8'))

```

```

elif filter_choice == "Blur":
    processed = processed.filter(ImageFilter.GaussianBlur(2))
elif filter_choice == "Cartoon":
    processed = processed.filter(ImageFilter.CONTOUR)

# Display Filtered Image
with col2:
    st.markdown("#### 🎨 Filtered Image")
    st.image(processed, use_column_width=True)

# Sidebar Download Option
st.sidebar.download_button("📥 Download Processed Image", convert_image(processed),
"background_removed.png", "image/png")

# Columns Layout
col1, col2 = st.columns(2)

# File uploader for multiple images
uploaded_files = st.sidebar.file_uploader("Choose images...", type=["png", "jpg", "jpeg"],
accept_multiple_files=True)

# Background replacement options
bg_option = st.sidebar.radio("🎨 Background Options", ["Transparent", "Solid Color", "Upload Background"])
bg_upload = None
if bg_option == "Upload Background":
    bg_upload = st.sidebar.file_uploader("Upload a new background", type=["png", "jpg", "jpeg"])

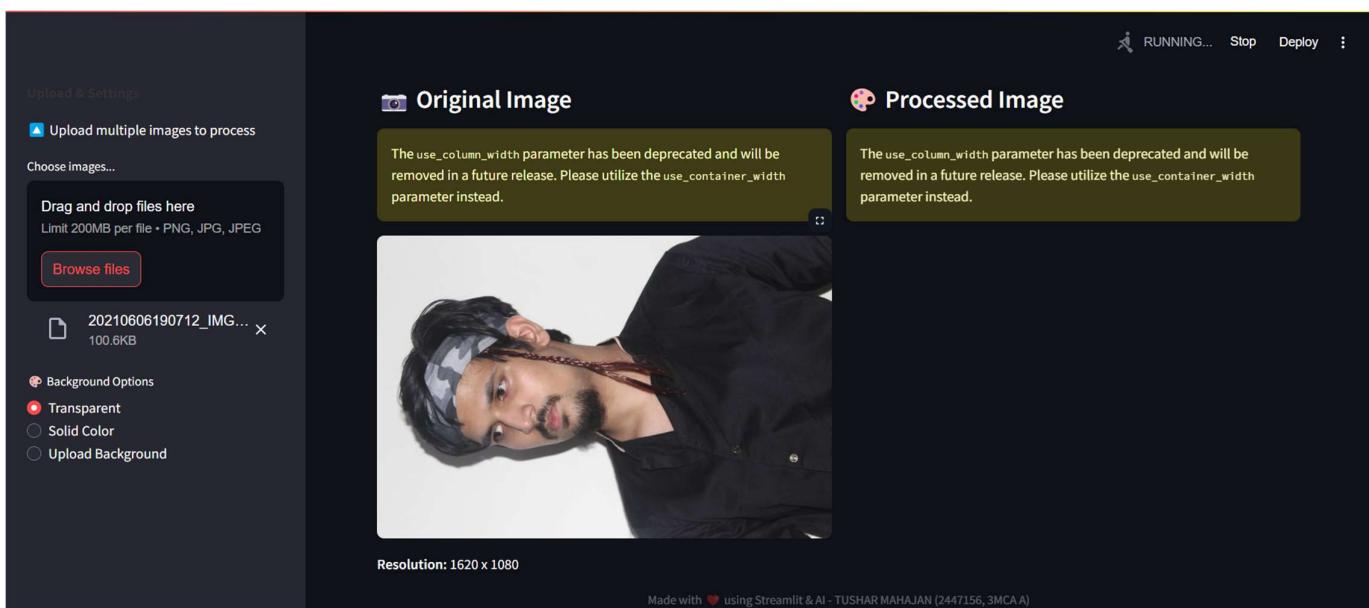
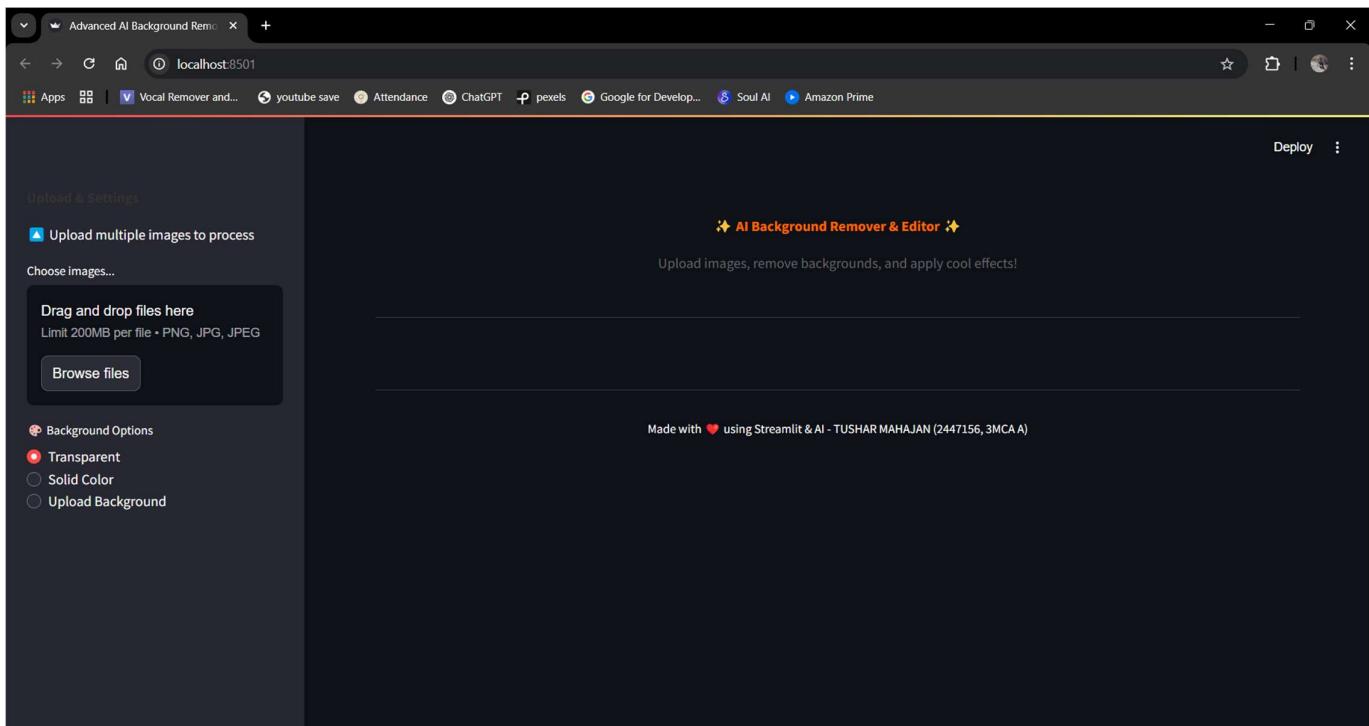
if uploaded_files:
    for uploaded_file in uploaded_files:
        if uploaded_file.size > MAX_FILE_SIZE:
            st.sidebar.error(f"❌ {uploaded_file.name} is too large! (Max 5MB)")
        else:
            process_image(uploaded_file, bg_option, bg_upload)

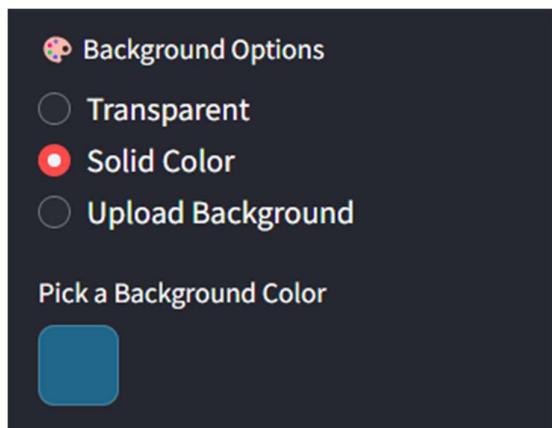
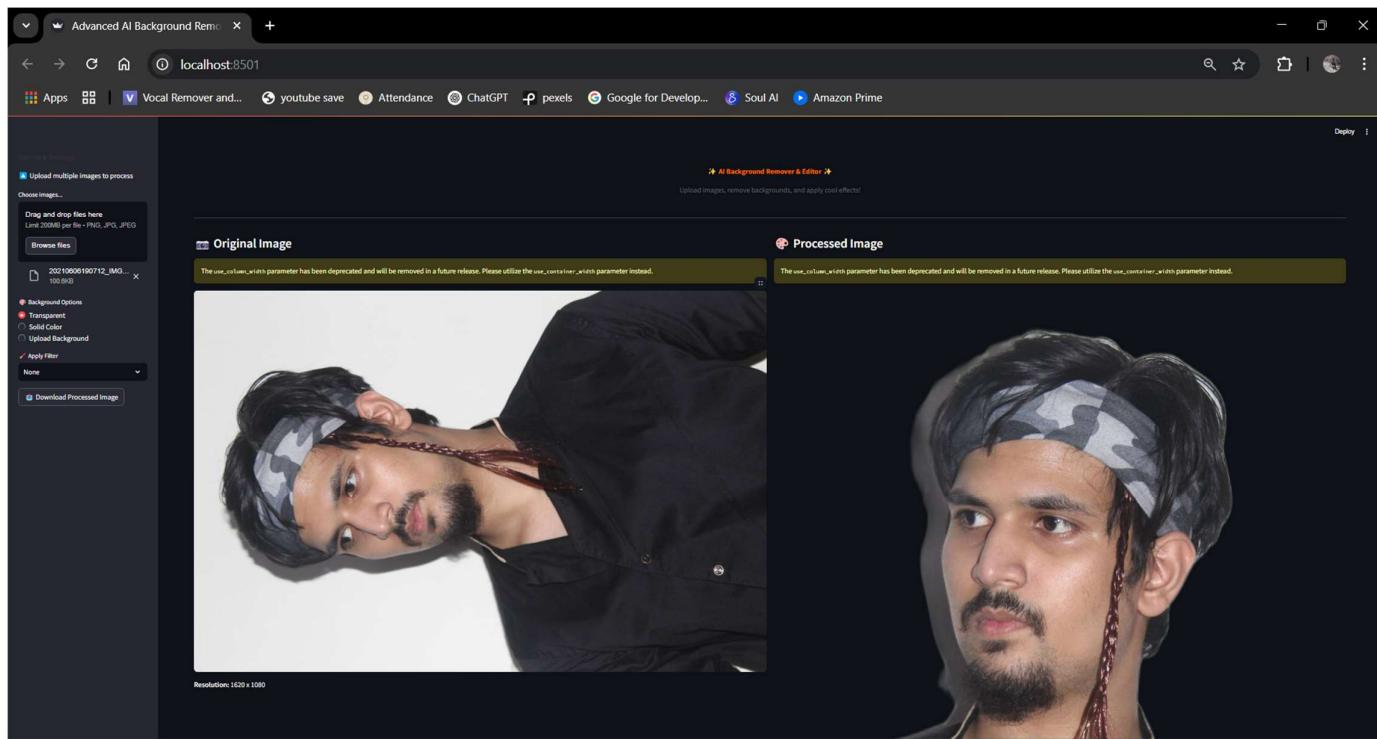
# Footer
st.markdown("---")
st.markdown('<p style="text-align:center; font-size:14px;">Made with ❤️ using Streamlit & AI - TUSHAR MAHAJAN (2447156, 3MCA A)</p>', unsafe_allow_html=True)

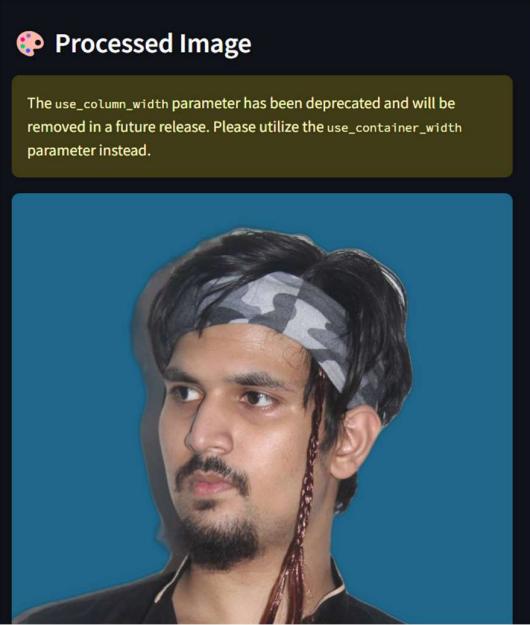
```

---

### ***Output Screenshots:***







#### Background Options

- Transparent
- Solid Color
- Upload Background

Upload a new background

Drag and drop file here  
Limit 200MB per file • PNG, JPG, JPEG

Browse files

#### Background Options

- Transparent
- Solid Color
- Upload Background

Upload a new background

Drag and drop file here  
Limit 200MB per file • PNG, JPG, JPEG

Browse files

 IMG\_20220210\_1304... X

306.1KB



Advanced AI Background Remover x +

localhost:8501

Choose images...

Drag and drop files here  
Limit 200MB per file - PNG, JPG, JPEG

Browse files

20210606190712\_IM... X 100.6KB

Background Options

Transparent

Solid Color

Upload Background

Upload a new background

Drag and drop file here  
Limit 200MB per file - PNG, JPG, JPEG

Browse files

IMG\_20220210\_1304... X 306.1KB

Apply Filter

None

Original Image

The `use_column_width` parameter has been deprecated and will be removed in a future release. Please utilize the `use_container_width` parameter instead.

Resolution: 1620 x 1080

Processed Image

The `use_column_width` parameter has been deprecated and will be removed in a future release. Please utilize the `use_container_width` parameter instead.

Apply Filter

Grayscale | ▾

