Edit | Preview | Code 55% faster with GitHub Copilot | Spaces | 4 | Soft wrap

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Image Recognition</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            text-align: center;
        }
        .container {
            max-width: 500px;
            margin: 0 auto;
            padding: 20px;
        }
        h1 {
            font-size: 24px;
        }
        form {
            margin-top: 20px;
        }
        input[type="file"] {
            width: 100%;
            padding: 10px;
            border: 1px solid #ccc;
```

Use Control + Shift + m to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.

Edit | Preview | Code 55% faster with GitHub Copilot | Spaces | 4 | Soft wrap

```
            border: 1px solid #ccc;
            border-radius: 5px;
        }
        input[type="submit"] {
            background-color: #007BFF;
            color: #fff;
            border: none;
            padding: 10px 20px;
            border-radius: 5px;
            cursor: pointer;
        }
        p {
            margin-top: 20px;
            font-weight: bold;
        }
        .error {
            color: red;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Image Recognition</h1>
        <form method="POST" enctype="multipart/form-data">
            <input type="file" name="image" accept="image/*" required>
            <input type="submit" value="Upload and Analyze">
```

Use `Control + Shift + m` to toggle the `tab` key moving focus. Alternatively, use `esc` then `tab` to move to the next interactive element on the page.

Edit | Preview | Code 55% faster with GitHub Copilot | Spaces | 4 | Soft wrap

```
        p {
            margin-top: 20px;
            font-weight: bold;
        }
        .error {
            color: red;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Image Recognition</h1>
        <form method="POST" enctype="multipart/form-data">
            <input type="file" name="image" accept="image/*" required>
            <input type="submit" value="Upload and Analyze">
        </form>
        {% if error %}
        <p class="error">{{ error }}</p>
        {% endif %}
        {% if caption %}
        <p>{{ caption }}</p>
        {% endif %}
    </div>
</body>
</html>

```

Use control + shift + m to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.

Code | Blame | 49 lines (38 loc) · 1.65 KB | Code 55% faster with GitHub Copilot | Raw

```python
# Install necessary libraries: pip install Flask ibm-watson ibm-cloud-sdk-core

from flask import Flask, render_template, request
from ibm_watson import VisualRecognitionV4
from ibm_watson.visual_recognition_v4 import FileWithMetadata
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator

app = Flask(__name)

# Replace with your actual API key and service URL
api_key = "YOUR_API_KEY"
service_url = "YOUR_SERVICE_URL"

authenticator = IAMAuthenticator(api_key)
visual_recognition = VisualRecognitionV4(
    version="2022-08-20",
    authenticator=authenticator
)
visual_recognition.set_service_url(service_url)

@app.route("/", methods=["GET", "POST"])
def index():
    caption = None
    if request.method == "POST":
        # Handle image upload
        if "image" not in request.files:
            return render_template("index.html", error="No file part")

        image = request.files["image"]
```

main ▾ CAD_NM_Project / SOURCE CODE_CAD_PROJECT / app.py ↑ Top

Code Blame 49 lines (38 loc) · 1.65 KB Code 55% faster with GitHub Copilot Raw

```
22  def index():
26          if "image" not in request.files:
27              return render_template("index.html", error="No file part")
28
29          image = request.files["image"]
30
31          if image.filename == "":
32              return render_template("index.html", error="No selected file")
33
34          try:
35              # Use Visual Recognition to analyze the uploaded image
36              response = visual_recognition.classify(
37                  images_file=FileWithMetadata(image)
38              ).get_result()
39
40              # Extract the top caption from the response
41              top_class = response["images"][0]["classifiers"][0]["classes"][0]["class"]
42              caption = f"AI-generated Caption: {top_class}"
43          except Exception as e:
44              return render_template("index.html", error=str(e))
45
46      return render_template("index.html", caption=caption)
47
48  if __name__ == "__main__":
49      app.run(debug=True)
```