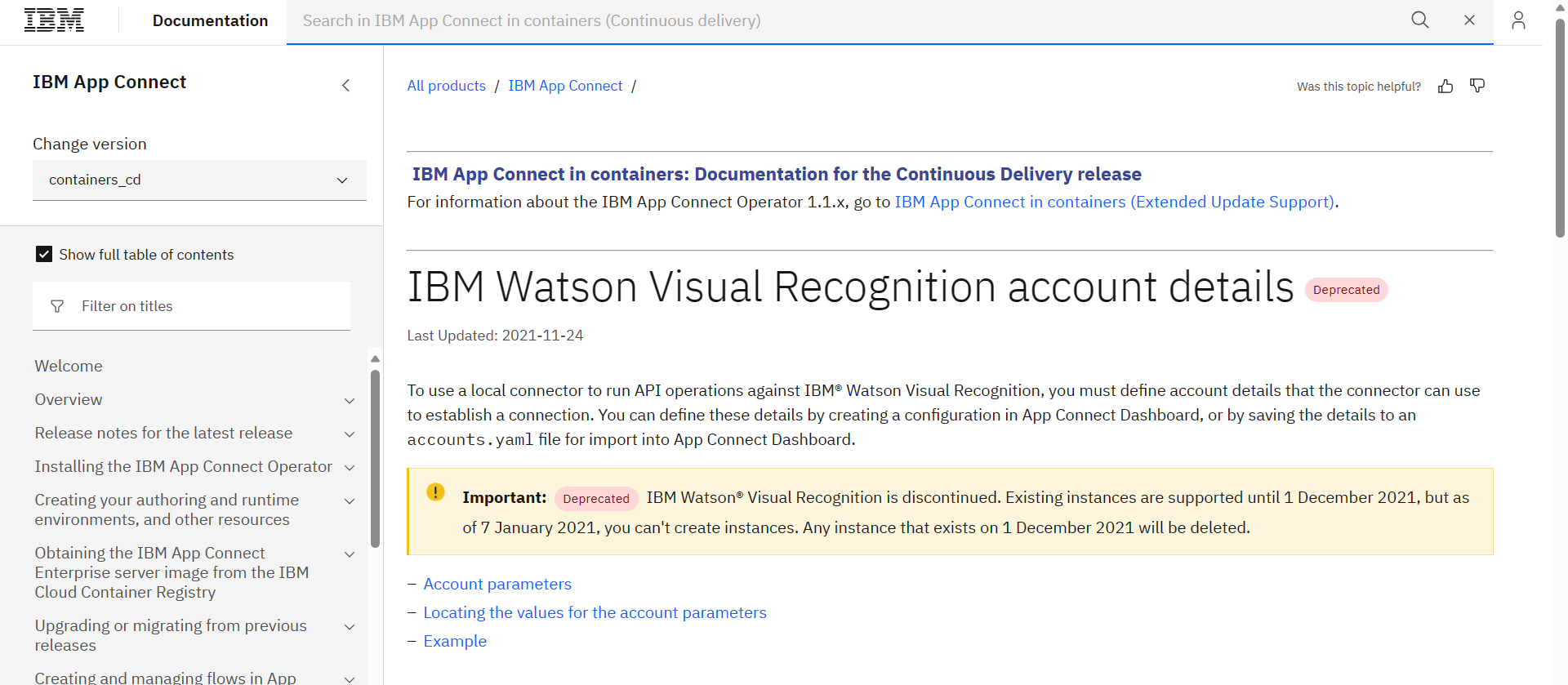
**IMAGE RECOGNITION WITH IBM CLOUD VISUAL RECOGNITION:**

**Waston visual recognition:**

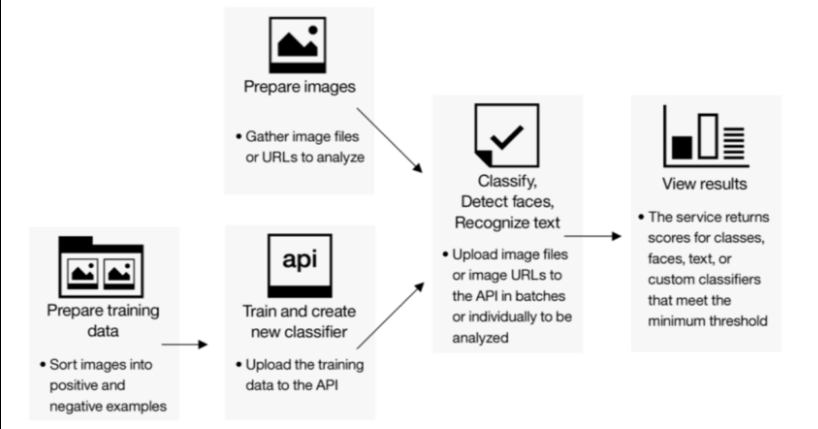
Given IBM Watson Visual Recognition is discontinued. Existing instances are supported until 1 December 2021, but as of 7 January 2021, you can't create instances. Any instance that exists on 1 December 2021 will be deleted. But we will put the frontend code for image recognition using HTML and CSS. The code will paste at the end of the page.

****

And also, there is no another solution for the image recognition. So won’t able generate any API key and Resource group in the IBM cloud. We know about the visual recognition using IBM cloud. There is few explanation about IBM visual recognition.

1. IBM Watson Visual Recognition is a powerful tool that can help businesses automate **image tagging**. This technology uses machine learning algorithms to identify objects, scenes, and other content in images. With Visual Recognition, businesses can quickly and accurately tag images, making it easier to organize and manage their digital assets.



1. IBM Watson Visual Recognition is a powerful tool that can enhance image analysis workflows. This cloud-based service uses deep learning algorithms to identify objects, faces, and scenes in images. It can also detect text and recognize logos.
2. Using Watson Visual Recognition, businesses can quickly and accurately analyze images and videos to gain insights into their operations. This can be used to detect anomalies, identify trends, and make better decisions. For example, a retail store can use Watson Visual Recognition to monitor the performance of their shelves and identify which products are selling well.
3. The service also provides a range of customization options, allowing businesses to tailor the image analysis to their specific needs. For example, businesses can train the service to recognize custom objects or faces, or to detect specific types of text. This allows businesses to gain more accurate insights from their images. 
4. Watson Visual Recognition is also easy to use. It has a simple user interface that makes it easy to upload images and videos, and to view the results. It also has an API that can be used to integrate the service into existing workflows.
5. Overall, IBM Watson Visual Recognition is a powerful tool that can help businesses gain more insights from their images and videos. With its range of customization options and easy-to-use interface, it can help businesses improve their image analysis workflows.

**HTML AND CSS CODE:**

**HTML:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <h1>AI Image Recognition</h1>

    <input type="file" id="image-upload" accept="image/\*" />

    <div id="image-container">

        <img id="uploaded-image" src="#" alt="Uploaded Image" style="display: none; max-width: 300px; margin: 10px;" />

    </div>

    <p id="image-caption"></p>

    <script>

        const imageUpload = document.getElementById('image-upload');

        const uploadedImage = document.getElementById('uploaded-image');

        const imageCaption = document.getElementById('image-caption');

        imageUpload.addEventListener('change', (e) => {

            const file = e.target.files[0];

            const reader = new FileReader();

            reader.onload = (e) => {

                uploadedImage.src = e.target.result;

                uploadedImage.style.display = 'block';

                imageCaption.textContent = 'Analyzing image...';

                // Send the image to the back-end for analysis (see back-end code below).

                fetch('/analyze-image', {

                    method: 'POST',

                    body: file

                })

                .then(response => response.json())

                .then(data => {

                    imageCaption.textContent = `AI-generated Caption: ${data.caption}`;

                })

                .catch(error => {

                    imageCaption.textContent = 'Error analyzing image.';

                });

            };

            if (file) {

                reader.readAsDataURL(file);

            }

        });

    </script>

</body>

</html>

**CSS:**

body {

    font-family: Arial, sans-serif;

    text-align: center;

}

#image-container {

    margin: 20px;

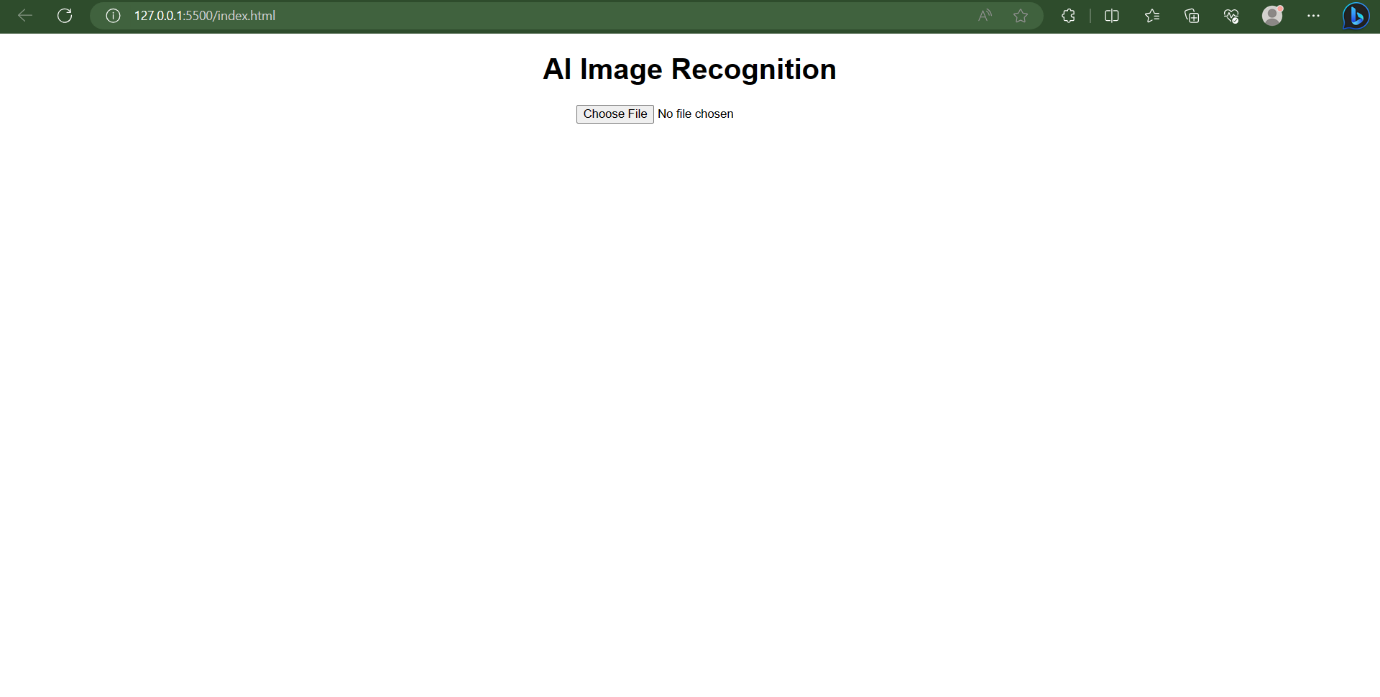
}

#image-caption {

    font-weight: bold;

}

**OUTPUT:**

****