

# Getting Started with Unsplash API

Integration and Best Practices

Presented by Cindy Kandie

# Welcome to our technical deep dive on the Unsplash API.

I am Cindy Kandie, frontend developer, a social engineer, and tech blogger. My background includes extensive experience in software development, community management, and API integration.

# Overview of Unsplash API

## **What is Unsplash API?**

- Provides programmatic access to a large collection of high-quality photos.

## **Capabilities**

- Fetch a list of photos.
- Search for specific images.
- Access various photo metadata.

# Setting Up Access

## **Access Key:**

- Obtain an access key by signing up on the Unsplash Developers portal.

## **Authentication:**

- Include the access key in your API requests for authentication.

# Example: Setting Up Access

```
```javascript
const fetchPhotos = async () => {
  const response = await fetch('https://api.unsplash.com/photos', {
    headers: {
      Authorization: 'Client-ID YOUR_ACCESS_KEY',
    },
  });
  const data = await response.json();
  console.log(data);
};
fetchPhotos();
```

# Key Endpoints

- ````markdown

## # Key Endpoints

- GET /photos:

- Fetch a list of photos.

- GET /photos/:id:

- Fetch a single photo by ID.

- GET /search/photos:

- Search for photos based on a query.

# Code Examples - Searching Photos with Key Endpoints

```
```javascript
const searchPhotos = async (query) => {
  const response = await fetch(`https://api.unsplash.com/search/photos?query=${query}`, {
    headers: {
      Authorization: 'Client-ID YOUR_ACCESS_KEY',
    },
  });
  const data = await response.json();
  console.log(data.results);
};
searchPhotos('nature');
```

# Authentication and Rate Limits

## **Authentication:**

- Use the Client-ID for authentication.

## **Rate Limits:**

- Understand the rate limits to avoid exceeding them (e.g., 50 requests per hour).

## **Handling Rate Limits::**

- Implement strategies to handle rate limits, such as caching responses.



# Code Examples - Fetching Photos

```
```javascript
const fetchPhotos = async () => {
  const response = await fetch('https://api.unsplash.com/photos', {
    headers: {
      Authorization: 'Client-ID YOUR_ACCESS_KEY',
    },
  });
  const data = await response.json();
  console.log(data);
};
fetchPhotos();
```

# Handling Errors and Rate Limits

## Error Handling::

- Implement error handling to manage API errors gracefully.

```
```javascript
const fetchPhotos = async () => {
  try {
    const response = await fetch('https://api.unsplash.com/photos', {
      headers: {
        Authorization: 'Client-ID YOUR_ACCESS_KEY',
      },
    });
    if (!response.ok) {
      throw new Error('Network response was not ok');
    }
    const data = await response.json();
    console.log(data);
  } catch (error) {
    console.error('Error fetching photos:', error);
  }
};

fetchPhotos();
```
```

# Q & A

## **Open Floor:**

- Open to questions from you.

**Feel free to ask anything about leveraging Unsplash API  
for your business.**