Vyaguta Leave Data API Documentation

Table of Contents

1. Introduction

The Vyaguta Leave Data API allows users to fetch, process, and visualize data related to employee leave from the Vyaguta application. The API is designed to automate data acquisition, ETL (Extract, Transform, Load) processes, and provide sample visualizations in a user-friendly manner.

This document provides detailed information on the available API endpoints, their parameters, usage examples, and response formats.

2. Authentication

All API requests require an `X-Custom-Passcode` header for authorization. The passcode ensures secure access to the API resources. Failure to provide the correct passcode will result in an authentication error.

Sample Header:

```
X-Custom-Passcode: passcode_here
```

3. API Endpoints

3.1 Fetching and Uploading Data

Triggers the function to acquire data from the Vyaguta application.

If `start_date` and `end_date` are not provided, the API defaults to using yesterday's date for `start_date` and today's date for `end_date`.

URL: `/api/v1/acquire/insert`

Method: `POST`

Headers:

- `X-Custom-Passcode` : Required, string.

Parameters:

- `start_date`: Optional, string in `YYYY-MM-DD` format. Defaults to yesterday.
- `end_date`: Optional, string in `YYYY-MM-DD` format. Defaults to today.

Response:

```
 {
    "message": "Insert operation started for leave Table initiated as 2024-09-12"
}
```

Usage Example:

```
curl --location 'http://0.0.0.0:4448/api/v1/acquire/insert' \
--header 'X-Custom-Passcode: passcode_here' \
--form 'start_date="2021-01-01"' \
--form 'end_date="2024-10-02"'
```

3.2 ETL Process

Triggers the ETL process for specific data sets.

If `inserted_date` is not provided, the current date is used. If `etl_name` is not specified, the

API processes all ETL jobs including `user`, `leave`, `designation`and `leave_txn`.

URL: `/api/v1/etl/load`

Method: `POST`

Headers:

- `X-Custom-Passcode` : Required, string.

Parameters:

- `inserted_date`: Optional, string in `YYYY-MM-DD` format. Defaults to today.
- `etl_name`: Optional, string. If not provided, processes `user`, `leave`, `designation`,and `leave_txn`.

Response:

```
{
    "message": "ETL process started for user Table as of 2024-09-07"
}
```

Usage Example:

```
curl --location 'http://0.0.0.0:4448/api/v1/etl/load' \
--header 'X-Custom-Passcode: passcode_here' \
--form 'inserted_date="2024-09-07"' \
--form 'etl_name="user"'
```

3.3 Sample Visualization

Provides a sample of the data in HTML format.

Returns an HTML page with sample data visualizations. The data is for sample purposes and can be viewed as a webpage.

URL:

- `/sample` - Returns a sample of the data without any filters.
- `/sample?startdate={startdate}&enddate={enddate}` - Returns a sample of the data filtered by the specified date range.

Method: `GET`

Query Parameters:

- `startdate` (optional) - The start date of the date range in the format YYYY-MM-DD.
- `enddate` (optional) - The end date of the date range in the format YYYY-MM-DD.

Headers:

- None

Response:

Returns an HTML page with embedded images of various visualizations based on the provided date range. If no date range is provided, the data is not filtered.

Usage Examples:

```
curl --location 'http://0.0.0.0:4448/sample'
```

```
curl --location 'http://0.0.0.0:4448/sample?startdate=2023-03-01&enddate=2024-07-01'
```

3.4 Visualize Download

Downloads a specific type of visualization.

Provides downloadable visualizations based on the specified plot type. Supported plot types include `department`, `supervisor`, `designation`, and `leave`.

URL: `/api/v1/viz/sample/download/{plot_type}`

Method: `GET`

Headers:

- `X-Custom-Passcode` : Required, string.

Parameters:

- `plot_type`: Required, string. The type of plot to download. Valid values are `department`, `supervisor`, `designation`, and `leave`.

Response:

> Returns a downloadable file containing the requested visualization.

Usage Example:

```
curl --location 'http://0.0.0.0:4448/api/v1/viz/sample/download/department' \
--header 'X-Custom-Passcode: passcode_here'
```

4. Response Formats

The API responses are generally in JSON format for data operations and downloadable files for visualizations. HTML format is used for sample visualization.