Detailed Explanation of the Crawler Configuration

This Glue crawler is configured to scan and catalog raw data stored in the S3 bucket (`s3://de-on-youtube-raw-us-east-1-dev-vignesh/youtube/raw_statistics/`) into the **AWS Glue Data Catalog**. Here's what each property means:

General Properties

1. Name:

- **`de-on-youtube-raw-us-east-1-dev-vignesh`**
  - The name indicates this crawler is part of a pipeline that deals with raw YouTube statistics data in the development environment for the `vignesh` user/project.

2. IAM Role:

- **`de-on-youtube-glue-s3-role`**
  - This IAM role allows the crawler to:
    - Access the specified S3 bucket and folders.
    - Update the Glue Data Catalog with the schema and metadata.
    - Perform other Glue-specific operations, like marking tables as deprecated.

3. Database:

- **`de_youtube_raw`**
  - The Glue database where the crawler stores metadata about the tables it creates or updates.
  - A logical grouping for raw data metadata.

4. State:

- **`READY`**
  - Indicates the crawler is correctly configured and ready to be executed.

Advanced Settings

1. Create Single Schema for Each S3 Path:

- **`False`**
  - This means that if there are multiple data formats or schemas within the S3 path, the crawler will create separate tables for each distinct schema it detects.

2. Inherit Schema from Table:

- `False`
    - The crawler does not attempt to inherit schemas from existing tables. Instead, it will infer schemas afresh with each crawl.

3. Schema Updates in the Data Store:

- **Update the table definition in the data catalog**
    - If the underlying data's schema changes (e.g., new columns are added), the table definition in the Glue Data Catalog will be updated.

4. Object Deletion in the Data Store:

- **Mark the table as deprecated in the data catalog**
    - If the crawler finds that the data for a specific table no longer exists in the S3 bucket, it will mark the table as deprecated in the Glue Data Catalog. This prevents accidental use of stale metadata.

5. Repeat Crawls of S3 Data Stores:

- **Crawl all folders again with every subsequent crawl**
    - The crawler will scan the entire directory structure of the specified S3 path each time it runs, ensuring the catalog reflects the most up-to-date state of the data.

6. Create Partition Index:

- `True`
    - Enables Glue to create a **partition index** for tables with partitioned data. This improves query performance in Athena by making partition pruning more efficient.

Data Source Configuration

Type:

- `S3`
    - The crawler scans data stored in Amazon S3.

Data Source:

- `s3://de-on-youtube-raw-us-east-1-dev-vignesh/youtube/raw_statistics/`
    - This is the S3 path where the raw statistics data is stored.
    - The crawler will scan all files and subfolders within this path.

Recrawl Behavior:

- **Recrawl all**
    - Every time the crawler runs, it scans the entire directory structure to detect:

- New files.
- Updated files.
- Schema changes.
- Deleted files or folders.

Key Features of This Crawler

1. **Automatic Schema Updates:**
   - Any new columns or changes in the data format are reflected in the Glue Data Catalog.
2. **Partition Awareness:**
   - The crawler creates partition indexes for faster queries.
3. **Stale Data Handling:**
   - If data is removed from S3, the corresponding Glue table is marked as deprecated to avoid errors during queries.
4. **Comprehensive Recrawling:**
   - The **recrawl all** setting ensures the catalog is always up to date, even if files or folders are added or removed in the S3 path.

Use Case Workflow

Step 1: Raw Data Ingestion

- Raw statistics data (e.g., CSV, JSON, or Parquet files) is uploaded to:
  - `s3://de-on-youtube-raw-us-east-1-dev-vignesh/youtube/raw_statistics/`

Step 2: Crawler Execution

- The crawler scans the S3 bucket, infers the schema, and creates or updates Glue tables in the **`de_youtube_raw`** database.

Step 3: Metadata Update

- Metadata about the raw data (e.g., columns, data types, partitions) is saved in the Glue Data Catalog.

Step 4: Query Raw Data

- Tools like Athena can query the raw data directly:
- `SELECT *`
- `FROM "de_youtube_raw"."raw_statistics"`
- `LIMIT 10;`

Step 5: Transformation

- Downstream processes (e.g., Lambda functions or Glue ETL jobs) can transform the raw data into cleaned or aggregated formats.
