**1.Create a System design flow by reverse engineering the data ingestion process**

* Create an S3 bucket within Amazon Simple Storage Service (Amazon S3), providing scalable object storage for storing and retrieving data from the web.
* Simulate creating a folder in the S3 bucket by creating an object with a key ending in a slash ("/"), visually representing a folder structure within the S3 console or programmatically.
* Upload CSV files into the designated folder within Amazon S3, storing objects with unique keys that identify each file.
* Access AWS Identity and Access Management (IAM) to securely manage access to AWS services and resources.
* Define an IAM role within IAM, specifying permissions that authorize entities like AWS services or users from other accounts to make AWS service requests.
* Create an IAM policy within IAM to define detailed permissions for specific actions, resources, and conditions.
* Attach the IAM policy to the IAM role, granting the defined permissions to the role for accessing AWS resources.
* Navigate to AWS Glue, a fully managed ETL service for preparing and loading data for analysis.
* Establish a database within AWS Glue to serve as a logical container for organizing and managing metadata about related tables.
* Create a table within the Glue database, representing structured data with defined schemas that can be queried using SQL or other query languages.
* Implement a crawler in AWS Glue to automatically discover and catalog data from specified sources, infer schemas, and populate the Glue Data Catalog with metadata.
* Complete the data ingestion and processing pipeline setup within AWS, enabling access, transformation, and analysis of data based on application or use case requirements.

**3.Understand the JSON to explain the code.**

In the policy:

* The **Version** field specifies the version of the IAM policy language being used (2012-10-17).
* The **Statement** array contains two permission statements:
  1. The first statement allows a comprehensive list of S3 actions (**Action**) on specific S3 bucket and object resources (**Resource**). Actions include operations like deleting objects, listing buckets, putting objects, and configuring S3 settings. The **Effect** of this statement is to "Allow" these actions.
  2. The second statement grants more limited permissions (**Action**) restricted to describing, getting, and listing operations (**Resource**) on specific S3 resources (**Effect** is "Allow").

The **Resource** field in each statement specifies the ARNs (Amazon Resource Names) of the S3 resources to which the actions apply. For example:

* **arn:aws:s3:::query-result-65d41890** refers to the specific S3 bucket named **query-result-65d41890**.
* **arn:aws:s3:::query-result-\*/\*** allows actions on objects within the **query-result-\*** bucket.
* **arn:aws:s3:::global-electric-car/us/washington/** refers to the S3 bucket **global-electric-car** within the **us/washington** path.