1. AWS CLI Configuration:

- **Command**: `aws configure`
- **Purpose**: This command is used to configure the AWS CLI with necessary credentials (Access Key, Secret Access Key, region, and output format). When executed, the user is prompted to enter their AWS credentials and settings.
- **How it works**:
  - AWS Access Key ID: Identifies your AWS account for programmatic access.
  - AWS Secret Access Key: A secret associated with your Access Key ID.
  - Region: Defines which AWS region you will interact with (e.g., `us-west-2`).
  - Output Format: Specifies the format of output data (e.g., `json`, `text`).

2. List All IAM Users:

- **Command**: `aws iam list-users`
- **Purpose**: Lists all IAM users in your AWS account. IAM users are individual entities in AWS, such as employees or applications, who need access to AWS services.
- **How it works**: When executed, this command displays a list of users with details like user creation date, user name, and ARN.

3. List IAM Groups:

- **Command**: `aws iam list-groups`
- **Purpose**: Lists all the IAM groups in your AWS account. Groups allow you to assign permissions to multiple users by attaching policies to the group rather than to individual users.
- **How it works**: This command shows all the groups and their associated metadata like creation time.

4. Inspect User Details:

- **Command**: `aws iam get-user --user-name [username]`
- **Purpose**: Retrieves details about a specific IAM user. Replace `[username]` with the actual username of the user.
- **How it works**: Provides specific information about the IAM user such as the user's ARN, creation date, and path.

5. Inspect Group Details:

- **Command**: `aws iam get-group --group-name [groupname]`
- **Purpose**: Retrieves details about a specific IAM group, including all the users that belong to the group. Replace `[groupname]` with the actual name of the group.
- **How it works**: Displays information about the group along with a list of users who are members of it.

6. List Policies Attached to a Group:

- **Command**: `aws iam list-attached-group-policies --group-name [group name]`
- **Purpose**: Lists all the policies attached to a specific IAM group. Policies define what actions are allowed or denied for a user or group in AWS.
- **How it works**: This command shows the list of policies (permissions) associated with the group, helping to determine what access levels are granted to members of the group.

7. Retrieve the Policy Document:

- **Command**: `aws iam get-policy-version --policy-arn [policy-arn] --version-id [version-id]`
- **Purpose**: Retrieves the full policy document associated with a policy attached to a group, in JSON format. Replace `[policy-arn]` and `[version-id]` with actual values.
- **How it works**: Displays the detailed contents of the policy, which includes actions, resources, and conditions that define what the group can or cannot do in AWS.

8. Add Users to Groups:

- **Commands**:
  - `aws iam add-user-to-group --user-name user-1 --group-name S3-Support`
  - `aws iam add-user-to-group --user-name user-2 --group-name EC2-Support`
  - `aws iam add-user-to-group --user-name user-3 --group-name EC2-Admin`
- **Purpose**: These commands assign specific users to corresponding IAM groups. Each group can have its own permissions and policies.
- **How it works**: The user is added to the group, granting them the permissions that are attached to the group.

9. List Users in Each Group:

- **Commands**:
  - `aws iam get-group --group-name S3-Support`
  - `aws iam get-group --group-name EC2-Support`
  - `aws iam get-group --group-name EC2-Admin`
- **Purpose**: These commands list the users who are members of the specified IAM groups.
- **How it works**: It will show the names of users in each group, helping to verify group memberships.