AWS S3 Redshift Zillow

1. Log on to AWS IAM → User Groups → Create Group → Name(zillow-group) → Under attach policies (Administrator Access) → Create Group
2. Navigate to Users → Create User → Name(zillow-user) → Check the Provide User Access option → Check the I want to create an IAM User option → Set a custom password → Add user to group → Create User
3. Under User, Go to Security Credentials → Create Access Key → Use Command Line Interface (CLI) option → Keep a note of the access key and the secret access key for future uses → Create Key → Logout of the Root User and Sign-In with the IAM User
4. Move to EC2 Console → Launch Instance → Name(zillow-EC2) → Choose Ubuntu Image → t2.medium (Instance Type) → Create New Key Pair (zillow_key_pair), key pair type (RSA), format(.pem) → Create Security Group -> Launch instance
5. Installing Dependencies → python version check, sudo apt update, sudo apt install python3-pip, sudo apt install python3.10-venv → python3 -m venv zillow_venv (Create Virtual Env), source zillow_venv/bin/activate (Activate) → pip install --upgrade awscli → pip install apache-airflow → pip install apache-airflow-providers-amazon → airflow version (Check the version), airflow standalone (Initialize)
6. Copy the public IPv4 of the instance and add port 8080 to access Airflow UI
7. In order to ensure that the UI loads the page, we need to open the ports under Security → Move to Security Groups → Edit Inbound Rules → Add Rule → Custom TCP - 8080 - Anywhere IPv4 → Save Rules → Refresh the Airflow Page → Add credentials
8. {Optional} {Remotely SSH to VS Code} Click on the instance → Connect → SSH Client → Copy command starting with ssh → Open CMD → Paste command and choose yes

9. {Optional} Go to VS Code → Extensions → Remote SSH → Install → Click on the bottom left icon → Connect to Host → Configure SSH Hosts → Select the one with Users → Add the following and save it there → Now reconnect with the icon → Choose the desired host → Platform (Linux) → Verify if the connection is working
10. In the airflow.cfg file, there is a path for dags folder which we have to create → Create the dags folder and a file named zillow_analytics.py → Also set the load_examples to False and Refresh the server
11. RapidAPI → Zillow → Subscribe to Test → Search for Properties (API) → Python Requests Code Snippet → Debug the Code Snippet and modify accordingly → Create config_api.json file under Airflow folder for the keys → Save the file and then refresh the Airflow UI page to see the DAG → Under Graph section, view the DAG path
12. Navigate to S3 to create a bucket (zillow-bucket) which will be used in the Airflow scheduler
13. Move to IAM → Roles → Create Role → AWS Service → EC2 → AmazonS3FullAccess (Role) → Name (Zillow-EC2-Access) → Create Role → Move to EC2 → Check the current instance → Under Actions, select Security -> Modify IAM Role → Choose IAM Role created before (Zillow-EC2-Access) → Update Policy
14. Create Another Role → AWS Service → Lambda → AmazonS3FullAccess, AWSLambdaBasicExecutionRole → Name (Zillow-Lambda-Access)
15. Go to Lambda Function → Create Function → Author from Scratch → copyRawJsonFile-lambda (name) → Runtime (Python 3.10) → Use an existing role (Zillow-Lambda-Access) → Create Function → Add Trigger → Source (S3) → Bucket (zillow-bucket) → All Object Create Events → Check the acknowledgement → Save and deploy the lambda code
16. Create another S3 bucket for copying the raw data into new bucket (zillow-copy-raw-data-bucket) and debug the pipeline for errors using Cloudwatch logs
17. Create another Lambda Function → Author from Scratch → transform-convert-to-csv-lambda (name) → Runtime (Python 3.10) → Use an existing role (Zillow-

Lambda-Access) → Create Function → Add Trigger → Source (S3) → Bucket (zillow-copy-raw-data-bucket) → All Object Create Events → Check the acknowledgement → Save and deploy the lambda code

18. Create another S3 bucket for saving the transformed data into new bucket (zillow-transformed-data-bucket) and debug the pipeline for errors using Cloudwatch logs
19. Need to create a connection between S3 and Airflow, Look for Admin → Connections → Add Connection → Connection id (aws_s3_conn), Connection Type (Amazon Web Services), AWS Access Key and Secret Access Key from earlier → Save the connection
20. {Do this for both lambda functions} In the lambda function console → Configuration → General Configuration → Edit → Set Timer (2-3 mins)
21. For the transformation function, add a layer → AWS Layers → AWSSDKPandas-Python310 → Add the layer → Debug the pipeline to check for errors
22. Navigate to Redshift → Create cluster → redshift-cluster-1 (cluster identifier) → Node Type (ra3.xlplus), No.of nodes (1) → admin username (awsuserzillow), set password→ Create cluster →Move to query editor v2 → click on the cluster → Choose database (dev), username (awsuserzillow) and password → create connection
23. Create a connection between Redshift and Airflow, Look for Admin → Connections → Add Connection → Connection id (conn_id_redshift), Connection Type (Amazon Redshift), Host (Go to Redshift cluster, copy the Endpoint uptil amazonaws.com), database (dev), username (awsuserzillow), password and port (5439) → Save Connection
24. Update the IAM Role (Zillow-EC2-Access) to allow for Redshift access as well → Add Permissions → Attach Policies → AmazonRedshiftFullAccess (Check this policy) → Save the role
25. In the redshift cluster, we need to update the inbound rules → Go to Properties → Network and Security Settings → VPC Security Group → Check for Inbound rules for the 0.0.0.0/0 (If absent, it needs to be added) → Turn on publicly accessible mode

26. Trigger the pipeline and then check if the data is present in the redshift cluster after completion → Try to execute it 2-3 more times to have more data for visualization
27. Navigate to Quicksight → Sign up for Quicksight Standard Edition → In the console, click on Datasets → Create a dataset → Redshift → data source name (zillowdataset), database (dev), username (awsuserzillow) and password → Create Data Source → Choose data from redshift & preview data → Navigate to sheet and create own graphs for analysis