

PART I: SQL with Amazon RDS and Redshift [100 points]

Let's assume you're the seller for your e-commerce website, now designing the SQL tables for your products, customers, and orders, to keep track of the basic sales condition.

Refer to textbook page 244~255 for MySQL, page 269~279 for Amazon Redshift.

1) SQL schema design: draw ER diagram like p244 Figure 6.1 for 3 tables: [10 points]

customer : order = 1 : N

order : product = N : M

Customers (cid, customer_name, gender, phone)

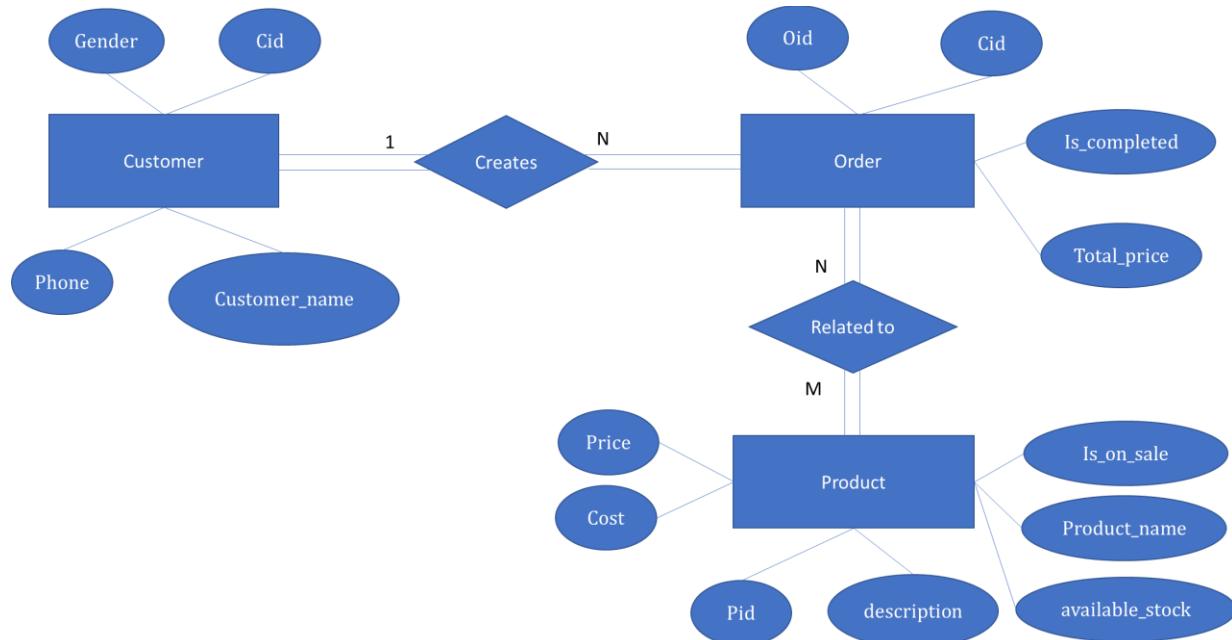
Order (oid, cid, is_completed, total_price)

OrderHasProduct (oid, pid, product_count)

Product (pid, product_name, price, cost, is_on_sale, available_stock, description)

Types: gender boolean, is_completed boolean, total_price int, product_count int, price int, cost int, is_on_sale boolean, available_stock int. all others are string

(if you have your own thought about the schemas, please post on Piazza or discuss with TA ahead.)



2) Locally: create the above tables in your local SQL or MySQL server. [30 points]

Create tables

```
CREATE DATABASE project3;

USE project3;

CREATE TABLE Customer(
cid varchar(50) NOT NULL PRIMARY KEY,
customer_name varchar(100),
gender boolean,
phone varchar(20)
);

CREATE TABLE Orders(
oid varchar(50) NOT NULL PRIMARY KEY,
cid varchar(50) NOT NULL REFERENCES Customer (cid),
is_completed boolean,
total_price int
);

CREATE TABLE Product(
pid varchar(50) NOT NULL PRIMARY KEY,
product_name varchar(100),
price int,
cost int,
is_on_sale boolean,
available_stock int,
description varchar(200)
);

CREATE TABLE OrderHasProduct(
oid varchar(50) REFERENCES Orders (oid),
pid varchar(50) REFERENCES Product (pid),
product_count int
);
```

```
mysql> source /home/aqc/project3/project3.sql
Query OK, 1 row affected (0.02 sec)

Database changed
Query OK, 0 rows affected (0.12 sec)

Query OK, 0 rows affected (0.02 sec)

Query OK, 0 rows affected (0.04 sec)

Query OK, 0 rows affected (0.03 sec)
```

```
mysql> SHOW DATABASES;
+-----+
| Database      |
+-----+
| events        |
| information_schema |
| mysql          |
| performance_schema |
| project3       |
| sys            |
+-----+
6 rows in set (0.00 sec)
```

```
mysql> USE project3;
Database changed
mysql> SHOW TABLES;
+-----+
| Tables_in_project3 |
+-----+
| Customer           |
| OrderHasProduct   |
| Orders              |
| Product             |
+-----+
4 rows in set (0.00 sec)
```

Insert 8 lines of data for each table and do the following queries.

```
aqc@ubuntu: ~/project3
INSERT INTO Customer VALUES ("0", "Alice", TRUE, "6693112564");
INSERT INTO Customer VALUES ("1", "Oliva", TRUE, "6695483265");
INSERT INTO Customer VALUES ("2", "Ava", TRUE, "6682548364");
INSERT INTO Customer VALUES ("3", "Emily", TRUE, "6691254896");
INSERT INTO Customer VALUES ("4", "Liam", FALSE, "6696648124");
INSERT INTO Customer VALUES ("5", "Lucas", FALSE, "6694747223");
INSERT INTO Customer VALUES ("6", "James", FALSE, "6694874359");
INSERT INTO Customer VALUES ("7", "Jason", FALSE, "6693015987");
INSERT INTO Customer VALUES ("8", "Bob", FALSE, "6693015964");

INSERT INTO Orders VALUES ("0", "2", FALSE, 66);
INSERT INTO Orders VALUES ("1", "3", FALSE, 108);
INSERT INTO Orders VALUES ("2", "1", TRUE, 66);
INSERT INTO Orders VALUES ("3", "1", TRUE, 70);
INSERT INTO Orders VALUES ("4", "0", FALSE, 54);
INSERT INTO Orders VALUES ("5", "5", FALSE, 2148);
INSERT INTO Orders VALUES ("6", "7", TRUE, 410);
INSERT INTO Orders VALUES ("7", "4", FALSE, 717);
INSERT INTO Orders VALUES ("8", "6", TRUE, 164);

INSERT INTO Product VALUES ("0", "shoes", 35, 32, TRUE, 3, "NO DESCRIPTION");
INSERT INTO Product VALUES ("1", "water", 41, 25, TRUE, 8, "NO DESCRIPTION");
INSERT INTO Product VALUES ("2", "wine", 66, 41, TRUE, 32, "NO DESCRIPTION");
INSERT INTO Product VALUES ("3", "cookie", 3, 2, TRUE, 678, "NO DESCRIPTION");
INSERT INTO Product VALUES ("4", "pen", 6, 3, TRUE, 452, "NO DESCRIPTION");
INSERT INTO Product VALUES ("5", "apple", 108, 22, FALSE, 0, "NO DESCRIPTION");
INSERT INTO Product VALUES ("6", "juice", 6, 1, TRUE, 365, "NO DESCRIPTION");
INSERT INTO Product VALUES ("7", "macbook pro", 99, 50, TRUE, 8, "NO DESCRIPTION");
;
INSERT INTO Product VALUES ("8", "wallet", 179, 110, TRUE, 90, "NO DESCRIPTION");

INSERT INTO OrderHasProduct VALUES ("0", "2", 1);
INSERT INTO OrderHasProduct VALUES ("4", "4", 3);
INSERT INTO OrderHasProduct VALUES ("6", "1", 10);
INSERT INTO OrderHasProduct VALUES ("7", "2", 9);
```

```
mysql> source /home/aqc/project3/insert.sql
Query OK, 1 row affected (0.10 sec)

Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.01 sec)

Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.01 sec)
    LibreOffice Writer
Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.01 sec)
```

consider generating some data that can print out at least 2 lines of results when doing the queries.

1) Get all the order.

```
mysql> source /home/aqc/project3/q1.sql
+----+----+-----+-----+
| oid | cid | is_completed | total_price |
+----+----+-----+-----+
| 0  | 2  |          0 |      69 |
| 1  | 3  |          0 |      32 |
| 2  | 1  |          1 |       2 |
| 3  | 1  |          1 |      58 |
| 4  | 0  |          0 |     109 |
| 5  | 5  |          0 |      66 |
| 6  | 7  |          1 |      72 |
| 7  | 4  |          0 |     588 |
| 8  | 6  |          1 |       3 |
+----+----+-----+-----+
9 rows in set (0.00 sec)
```

2) Get top 3 orders with highest total_price.

```
mysql> source /home/aqc/project3/q2.sql
+----+----+----+-----+
| oid | cid | is_completed | total_price |
+----+----+----+-----+
| 7   | 4   |           0 |      588 |
| 4   | 0   |           0 |     109 |
| 6   | 7   |           1 |      72  |
+----+----+----+-----+
3 rows in set (0.06 sec)
```

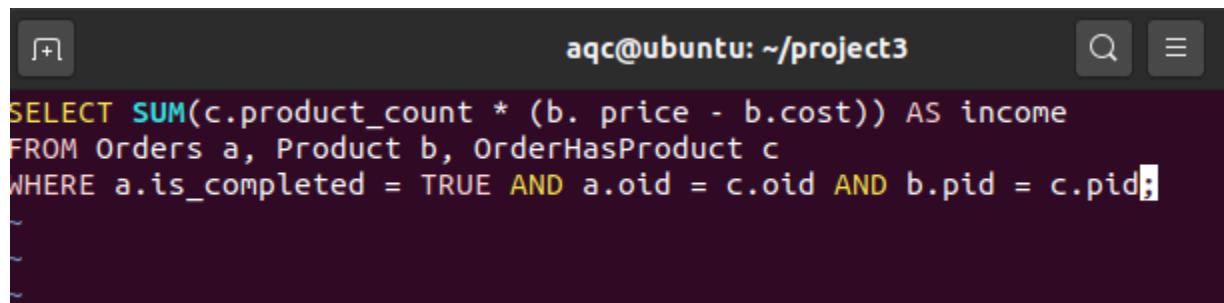
3) Get all the orders belong to customer Bob.

```
mysql> source /home/aqc/project3/q3.sql
+----+----+----+-----+
| cid | customer_name | gender | phone      |
+----+----+----+-----+
| 8   | Bob            |     0 | 6693015964 |
+----+----+----+-----+
1 row in set (0.00 sec)
```

4) Get the number of products that is on sale.

```
mysql> source /home/aqc/project3/q4.sql
+-----+
| COUNT(*) |
+-----+
|       8  |
+-----+
1 row in set (0.17 sec)
```

5) Get the total income for the completed orders. (income = price - cost)



The screenshot shows a terminal window with the following content:

```
aqc@ubuntu: ~/project3
SELECT SUM(c.product_count * (b.price - b.cost)) AS income
FROM Orders a, Product b, OrderHasProduct c
WHERE a.is_completed = TRUE AND a.oid = c.oid AND b.pid = c.pid;
```

```
mysql> source /home/aqc/project3/q5.sql
+-----+
| income |
+-----+
|    255 |
+-----+
1 row in set (0.02 sec)
```

6) Get the product and the number that need to purchase to fulfill the order.
(Available_stock - product_count)

```
aqc@ubuntu: ~/project3
SELECT a.product_name, (b.counts - a.available_stock) AS needToPurchase
FROM Product a, (SELECT pid, SUM(product_count) AS counts FROM OrderHasProduct G
ROUP BY pid WITH ROLLUP) b
WHERE a.pid = b.pid;
```

```
mysql> source /home/aqc/project3/q6.sql
+-----+-----+
| product_name | needToPurchase |
+-----+-----+
| shoes        |          -1 |
| water        |           9 |
| wine         |          -21 |
| cookie       |         -676 |
| pen          |         -444 |
| apple        |            1 |
| wallet       |          -78 |
+-----+-----+
7 rows in set (0.35 sec)
```

3) Amazon RDS: refer to Box 6.4 6.5 in p253 to launch an RDS instance, and create the above MySQL tables, execute the data insertion and doing queries 4~6. You can reuse the code in part 2. [30 points]

Create RDS instance and launch instance

```
(base) aqc@ubuntu:~/project3$ python launchInstance.py
Connecting to RDS
Creating an RDS instance
[{"DBInstance": {"DBInstanceIdentifier": "mysql-db-instance", "DBInstanceClass": "db.t2.micro", "Engine": "mysql", "DBInstanceStatus": "creating", "MasterUsername": "root", "DBName": "mytest", "AllocatedStorage": 5, "PreferredBackupWindow": "07:56-08:26", "BackupRetentionPeriod": 1, "DBSecurityGroups": [], "VpcSecurityGroups": [{"VpcSecurityGroupId": "sg-a62825ec", "Status": "active"}], "DBParameterGroups": [{"DBParameterGroupName": "default.mysql8.0", "ParameterApplyStatus": "in-sync"}], "DBSubnetGroup": {"DBSubnetGroupName": "default", "DBSubnetGroupDescription": "default", "VpcId": "vpc-c57ad0a3", "SubnetGroupStatus": "Complete", "Subnets": [{"SubnetIdentifier": "subnet-9d6ccdc7", "SubnetAvailabilityZone": {"Name": "us-west-1a"}, "SubnetOutpost": {}, "SubnetStatus": "Active"}, {"SubnetIdentifier": "subnet-6ec1a408", "SubnetAvailabilityZone": {"Name": "us-west-1b"}, "SubnetOutpost": {}, "SubnetStatus": "Active"}]}, "PreferredMaintenanceWindow": "sun:1 1:11-sun:11:41", "PendingModifiedValues": {"MasterUserPassword": "****"}, "MultiAZ": False, "EngineVersion": "8.0.23", "AutoMinorVersionUpgrade": True, "ReadReplicaDBInstanceIdentifiers": [], "LicenseModel": "general-public-license", "OptionGroupMemberships": [{"OptionGroupName": "default:mysql-8-0", "Status": "in-sync"}], "PubliclyAccessible": True, "StorageType": "gp2", "DbInstancePort": 0, "StorageEncrypted": False, "DbiResourceId": "db-HV2COMPMSGXRUKXUST20R5YPA", "CACertificateIdentifier": "rds-ca-2019", "DomainMemberships": [], "CopyTagsToSnapshot": False, "MonitoringInterval": 0, "DBInstanceArn": "arn:aws:rds:us-west-1:008889205193:db:mysql-db-instance", "IAMDatabaseAuthenticationEnabled": False, "PerformanceInsightsEnabled": False, "DeletionProtection": False, "AssociatedRoles": [], "TagList": [], "CustomerOwnedEncryptionEnabled": False, "ResponseMetadata": {"RequestId": "2018ee55-e997-44b1-bc9e-613e1e76d4fe", "HTTPStatusCode": 200, "HTTPHeaders": {"x-amzn-requestid": "2018ee55-e997-44b1-bc9e-613e1e76d4fe", "content-type": "text/xml", "content-length": "3809"}, "date": "Mon, 22 Nov 2021 22:13:19 GMT"}, "RetryAttempts": 0}}
Waiting for instance to be up and running
Status: creating
Status: creating
Status: creating

Success: backing-up
Status: backing-up
Status: backing-up
Status: backing-up
Status: backing-up
Status: available

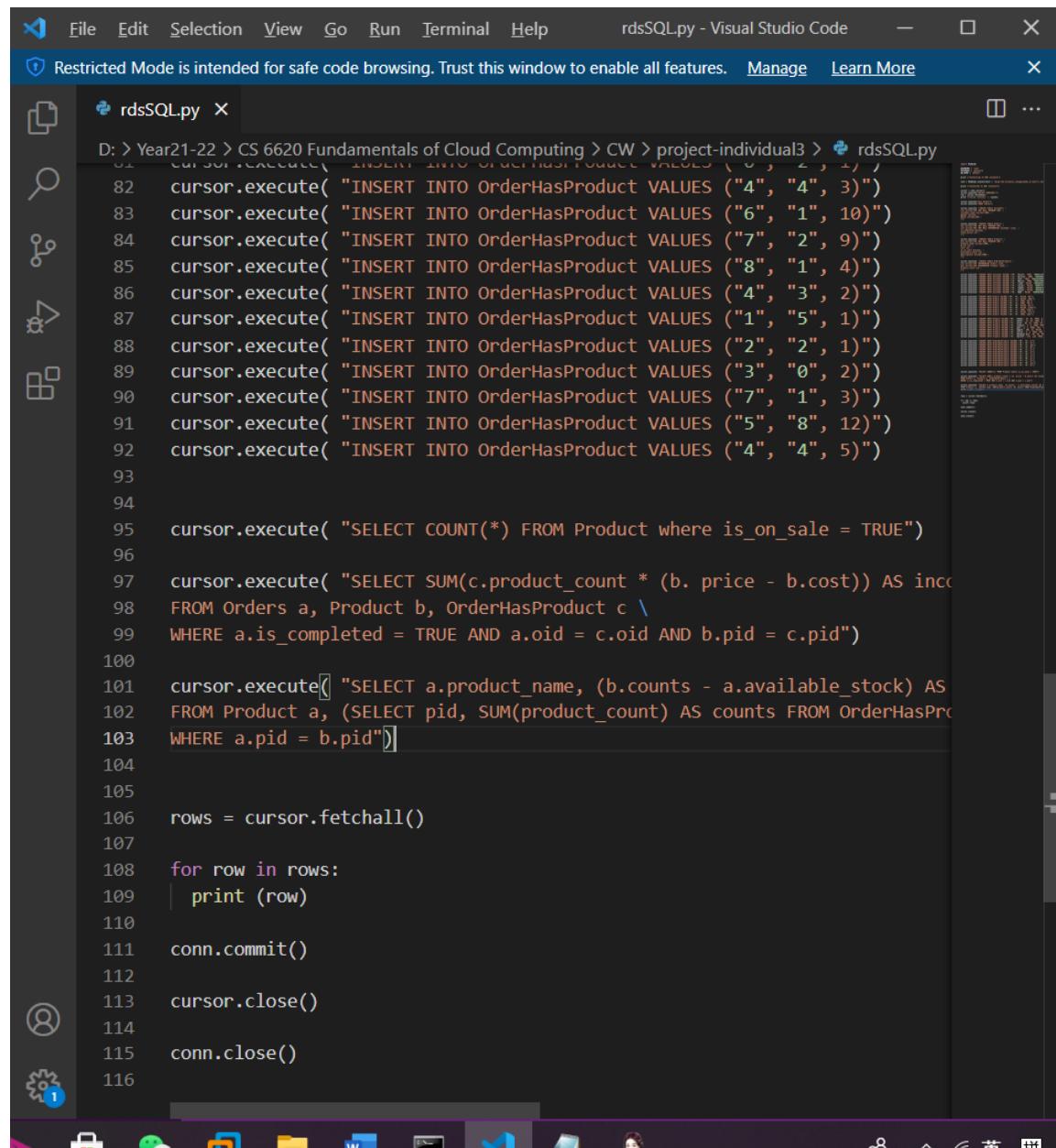
RDS Instance is now running. Instance details are:
Instance ID: mysql-db-instance
Instance Status: available
Instance Type: db.t2.micro
Engine: mysql
Allocated Storage: 5
Endpoint: {'Address': 'mysql-db-instance.cz4jqqocwk5w.us-west-1.rds.amazonaws.com', 'Port': 3306, 'HostedZoneId': 'Z10WI91S55XXQN'}
```

DB identifier	Role	Engine	Region & AZ	Size	Status	CPU	Current activity
mysql-db-instance	Instance	MySQL Community	us-west-1a	db.t2.micro	Available	8.56%	0 Connect

Create a MySQL table, writing and reading from the table

For this part, some libraries didn't work on my Linux VM after many attempts, so that I decide to run it on my Window laptop.

```
C:\Users\Administrator>pip install mysqlclient
Collecting mysqlclient
  Downloading mysqlclient-2.1.0-cp38-cp38-win_amd64.whl (180 kB)
    |████████| 180 kB 1.7 MB/s
Installing collected packages: mysqlclient
Successfully installed mysqlclient-2.1.0
```



```
D: > Year21-22 > CS 6620 Fundamentals of Cloud Computing > CW > project-individual3 > rdsSQL.py
  1 cursor.execute("INSERT INTO OrderHasProduct VALUES ('1', '1', 1)")
  2 cursor.execute("INSERT INTO OrderHasProduct VALUES ('4', '4', 3)")
  3 cursor.execute("INSERT INTO OrderHasProduct VALUES ('6', '1', 10)")
  4 cursor.execute("INSERT INTO OrderHasProduct VALUES ('7', '2', 9)")
  5 cursor.execute("INSERT INTO OrderHasProduct VALUES ('8', '1', 4)")
  6 cursor.execute("INSERT INTO OrderHasProduct VALUES ('4', '3', 2)")
  7 cursor.execute("INSERT INTO OrderHasProduct VALUES ('1', '5', 1)")
  8 cursor.execute("INSERT INTO OrderHasProduct VALUES ('2', '2', 1)")
  9 cursor.execute("INSERT INTO OrderHasProduct VALUES ('3', '0', 2)")
 10 cursor.execute("INSERT INTO OrderHasProduct VALUES ('7', '1', 3)")
 11 cursor.execute("INSERT INTO OrderHasProduct VALUES ('5', '8', 12)")
 12 cursor.execute("INSERT INTO OrderHasProduct VALUES ('4', '4', 5)")

 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30
 31
 32
 33
 34
 35
 36
 37
 38
 39
 40
 41
 42
 43
 44
 45
 46
 47
 48
 49
 50
 51
 52
 53
 54
 55
 56
 57
 58
 59
 60
 61
 62
 63
 64
 65
 66
 67
 68
 69
 70
 71
 72
 73
 74
 75
 76
 77
 78
 79
 80
 81
 82
 83
 84
 85
 86
 87
 88
 89
 90
 91
 92
 93
 94
 95
 96
 97
 98
 99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
```

Query 4

```
97  
98     #cursor.execute( "SELECT COUNT(*) FROM Product where is_on_sale = TRUE")  
99
```

```
D:\Year21-22\CS 6620 Fundamentals of Cloud Computing\CW\project-individual3>python rdsSQL.py  
Connecting to RDS instance  
Connected to RDS instance  
server version: 8.0.23  
(8,)
```

Query 5

```
100   ...  
101   cursor.execute( "SELECT SUM(c.product_count * (b. price - b.cost)) AS income \  
102     FROM Orders a, Product b, OrderHasProduct c \  
103    WHERE a.is_completed = TRUE AND a.oid = c.oid AND b.pid = c.pid")  
104   ...
```

```
D:\Year21-22\CS 6620 Fundamentals of Cloud Computing\CW\project-individual3>python rdsSQL.py  
Connecting to RDS instance  
Connected to RDS instance  
server version: 8.0.23  
(Decimal('255'),)
```

Query 6

```
106   ...  
107   cursor.execute( "SELECT a.product_name, (b.counts - a.available_stock) AS needToPurchase \  
108     FROM Product a, (SELECT pid, SUM(product_count) AS counts FROM OrderHasProduct GROUP BY pid WITH ROLLUP) b \  
109    WHERE a.pid = b.pid")  
110   ...|
```

```
D:\Year21-22\CS 6620 Fundamentals of Cloud Computing\CW\project-individual3>python rdsSQL.py  
Connecting to RDS instance  
Connected to RDS instance  
server version: 8.0.23  
(‘shoes’, Decimal(‘-1’))  
(‘water’, Decimal(‘9’))  
(‘wine’, Decimal(‘-21’))  
(‘cookie’, Decimal(‘-676’))  
(‘pen’, Decimal(‘-444’))  
(‘apple’, Decimal(‘1’))  
(‘wallet’, Decimal(‘-78’))
```

4) Amazon Redshift: refer to Box 6.13 to load your data into with S3 bucket. You can use any input type other than csv. Show the process to add additional nodes into the cluster, remove nodes from the cluster while remaining operational. The nodes can be duplicated data but different names. Do queries 1~3. [30 points]

Create a Redshift cluster

Stack name: redshift-cluster-1

Region: US East (N. Virginia)

Next Step

Specify the node type and cluster type

Stack name: redshift-cluster-1

Region: US East (N. Virginia)

Next Step

Other configurations

Additional configurations Use defaults

These configurations are optional, and default settings have been defined to help you get started with your cluster. Turn off "Use defaults" to modify these settings now.

Network Using default VPC (vpc-c57ad0a3) and default subnet.	Security Using default (sg-a62825ec) cluster security group.
Backup Automated snapshots are created about every eight hours or following every 5 GB per node of data changes, whichever comes first.	Configuration Using default.redshift-1.0 parameter group with no database encryption.
Maintenance Using current maintenance track.	

Results

The screenshot shows the AWS Redshift console interface. On the left, there's a sidebar with various navigation links: QUERIES, EDITOR, DATASHA..., CONFIG, MARKETP..., ADVISOR, ALARMS, EVENTS, and WHAT'S NEW. The main area has three main sections:

- Query data using Redshift query editor:** A section for running SQL queries. It includes a "Query data" button and a "Copy JDBC URL" and "Copy ODBC URL" option.
- Work with your client tools:** Instructions for connecting client tools like SQL clients, BI tools, and ETL tools using JDBC or ODBC drivers. It also includes a "Cluster identifier" dropdown, "Copy JDBC URL" and "Copy ODBC URL" buttons, and a "Download driver" link.
- Clusters (1) Info:** A table showing one cluster named "redshift-cluster-1". The table columns include Cluster, Cluster namespace, Status, Storage capacity us..., CPU utilization, Snapshots, Notifications, and Tags. The cluster status is "Available" with 2% CPU utilization and 3 snapshots.

At the bottom, there are links for Feedback, English (US), and a footer with copyright information: © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

Connect to a database

Connect to database

X

Connection

Select a recent database connection or create a new database connection.

- Use a recent connection
- Create a new connection

Authentication

- Temporary credentials

Use the GetClusterCredentials IAM permission and your database user to generate temporary access credentials. [Learn more](#)

- AWS Secrets Manager

Use a stored secret to authenticate access. [Learn more](#)

Cluster

redshift-cluster-1 (Available)



Database name

dev

Database user

User name authorized to access your database.

awsuser

Cancel

Connect

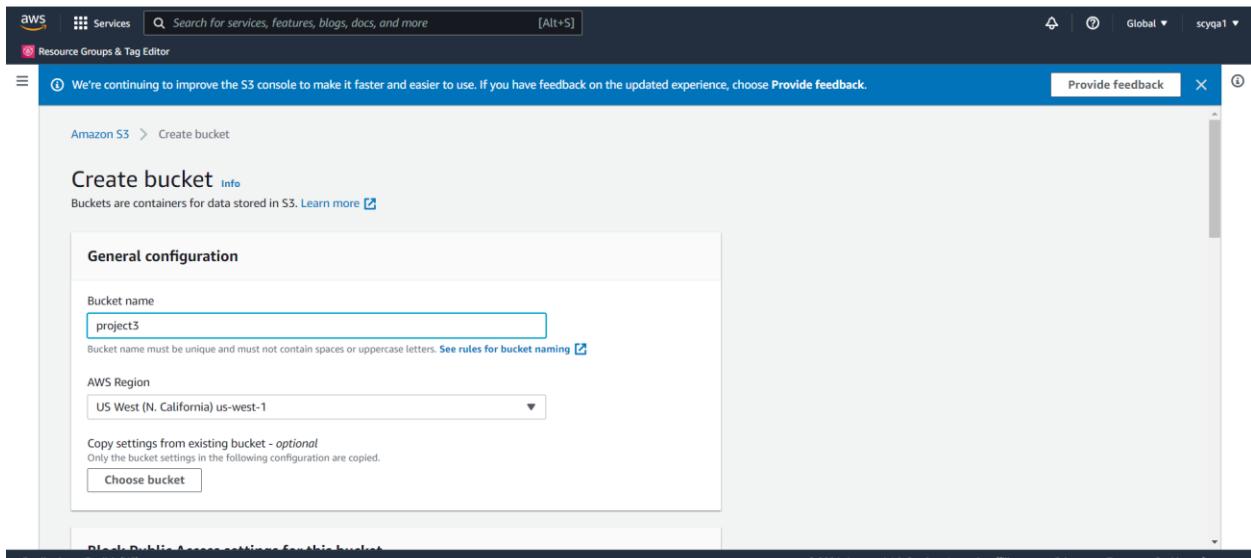
Create Table

The screenshot shows the AWS Redshift Query Editor interface. On the left, the sidebar includes links for Services, Resource Groups & Tag Editor, QUERIES (selected), EDITOR, DATASHA..., CONFIG, MARKETP..., ADVISOR, ALARMS, EVENTS, and WHAT'S N... At the top, there's a search bar and navigation buttons for AWS Regions (N. California) and User (scyqa1). The main area has tabs for Resources and Info. A dropdown for Select database shows 'dev'. A dropdown for Select schema shows 'public'. A 'Query 1' tab is active, containing the following SQL code:

```
1 create table customers (
2     customerNumber char(40) not null distkey sortkey ,
3     customerName varchar(50),
4     phoneNumber varchar(14),
5     postalCode varchar(4),
6     locale varchar(11),
7     dateCreated timestamp,
8     email varchar(20));
```

Below the code are buttons for Run, Save, Schedule, and Clear. The status bar at the bottom indicates the query completed successfully on November 23, 2021, at 16:46:47, with an elapsed time of 00 m 03 s. There are also tabs for Query results, Table details, Execution, Data, and Visualize.

Upload JSON data to S3



The screenshot shows the 'Create bucket' page in the AWS S3 console. The 'General configuration' section is visible, with the 'Bucket name' field set to 'project3'. Other fields include 'AWS Region' (set to 'US West (N. California) us-west-1') and a 'Copy settings from existing bucket - optional' section which is currently empty.

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Name	Type	Last modified	Size	Storage class
data.json	json	November 23, 2021, 16:52:02 (UTC-08:00)	680.0 B	Standard

Resize

The screenshot shows the AWS Amazon Redshift Cluster Details page for a cluster named "redshift-cluster-1". The left sidebar has a "CLUSTERS" tab selected. The main area displays "General information" for the cluster, including its identifier, status (Available), node type (dc2.large), and storage usage. Below this are tabs for "Cluster performance", "Query monitoring", "Schedules", "Maintenance", and "Properties". A "Actions" menu is open, showing options like "Manage cluster", "Resize", "Reboot", etc. The "Resize" option is highlighted. The URL in the browser is <https://console.aws.amazon.com/redshift/v2/cluster/redshift-cluster-1>.

Add nodes

The screenshot shows the "Add nodes" configuration dialog. It starts with a summary row: "dc2.large" node type, "1" node, "160 GB" storage, and "Estimated pricing: \$7.92/day". Below this is a "New configuration" section where the node type is set to "dc2.large" (Storage type: NVMe-SSD). The "Nodes" section shows "2" nodes selected, resulting in "320 GB total storage" and "Estimated pricing: \$15.84/day". A warning message states: "⚠️ Resizing the cluster restarts it, and it stays in read-only mode through the resize operation." At the bottom, there are three options for scheduling the resize: "Resize the cluster now" (selected), "Schedule resize at a later time", and "Schedule recurring resize events". The "Resize cluster now" button is highlighted in orange.

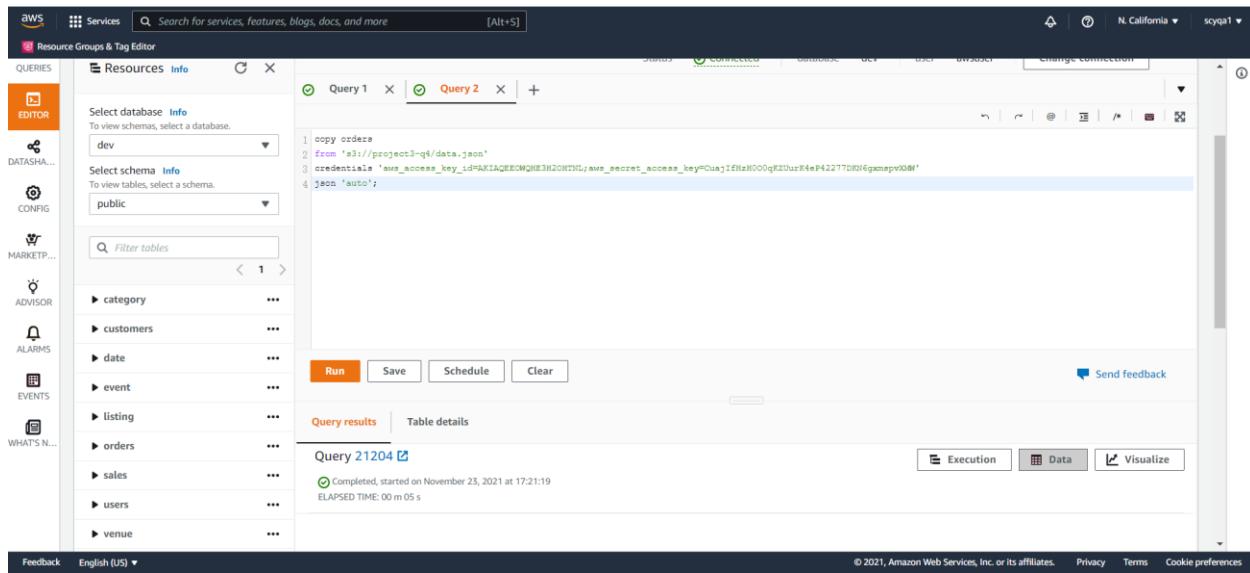
The screenshot shows the AWS Redshift Cluster Overview page for a cluster named 'redshift-cluster-1'. A green banner at the top indicates that the cluster is currently resizing. Below the banner, a message通知关于Amazon Redshift查询编辑器v2的新功能。The main content area displays the 'General information' for the cluster, including its identifier, status (Available), node type (dc2.large), endpoint, and connection details. The sidebar on the left provides navigation links for various Redshift features like Clusters, Queries, and Editor.

Similarly, remove nodes by resize

The screenshot shows the 'New configuration' dialog for resizing a Redshift cluster. It allows selecting a new node type (dc2.large) and specifying the number of compute nodes (1). The estimated price per day is \$7.92. The dialog also includes a note about the cluster being read-only during resize and options to resize now or schedule for later.

The screenshot shows the Redshift Cluster Overview page again, but this time the cluster is in a 'Modifying' state, indicated by a blue banner at the top. The main content area shows the updated 'General information' for the cluster, reflecting the resize operation. The sidebar remains the same, providing access to other cluster management features.

Copy S3 data into Redshift



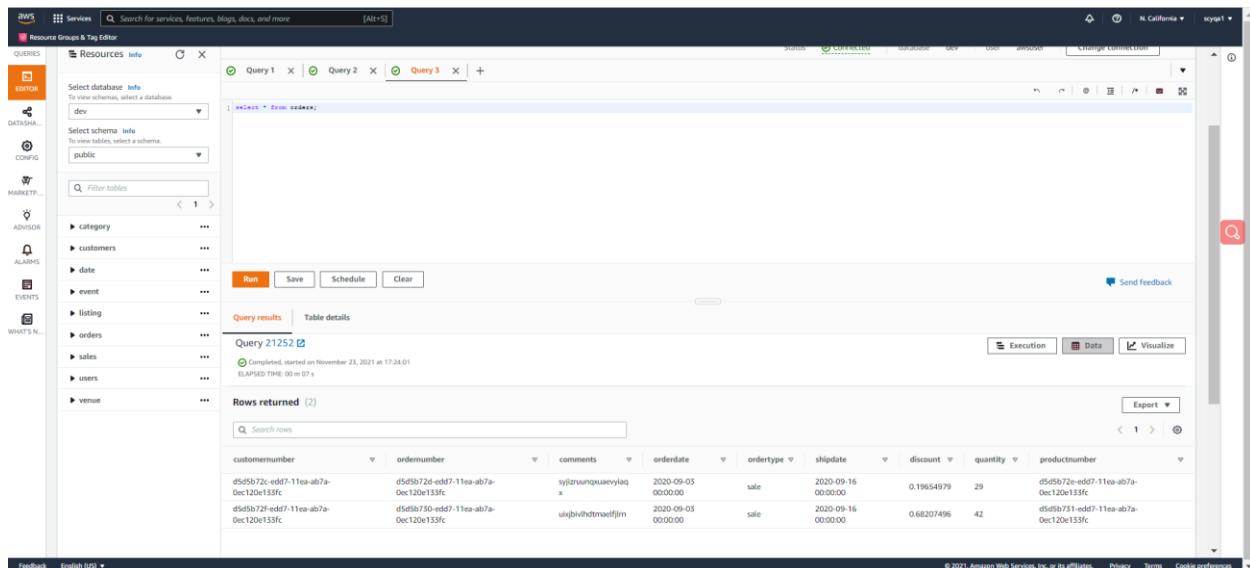
The screenshot shows the AWS Redshift Query Editor interface. On the left, there's a sidebar with various service icons. The main area has two tabs: 'Query 1' and 'Query 2'. The 'Query 2' tab is active, containing the following SQL code:

```
copy orders
from 's3://project3-q4/data.json'
credentials 'aws_access_key_id=AKIAQEONQHE3H2ONTL;aws_secret_access_key=CuejIENh000qKUurK4eP42z77D0N6gxmspv00W'
json 'auto';
```

Below the code, there are buttons for 'Run', 'Save', 'Schedule', and 'Clear'. The 'Run' button is highlighted in orange. To the right of the code, there's a 'Send feedback' link. At the bottom, it says 'Query 21204' with a status of 'Completed, started on November 23, 2021 at 17:21:19' and an 'ELAPSED TIME: 00 m 05 s'. There are tabs for 'Execution', 'Data', and 'Visualize'.

Do queries

Query 1



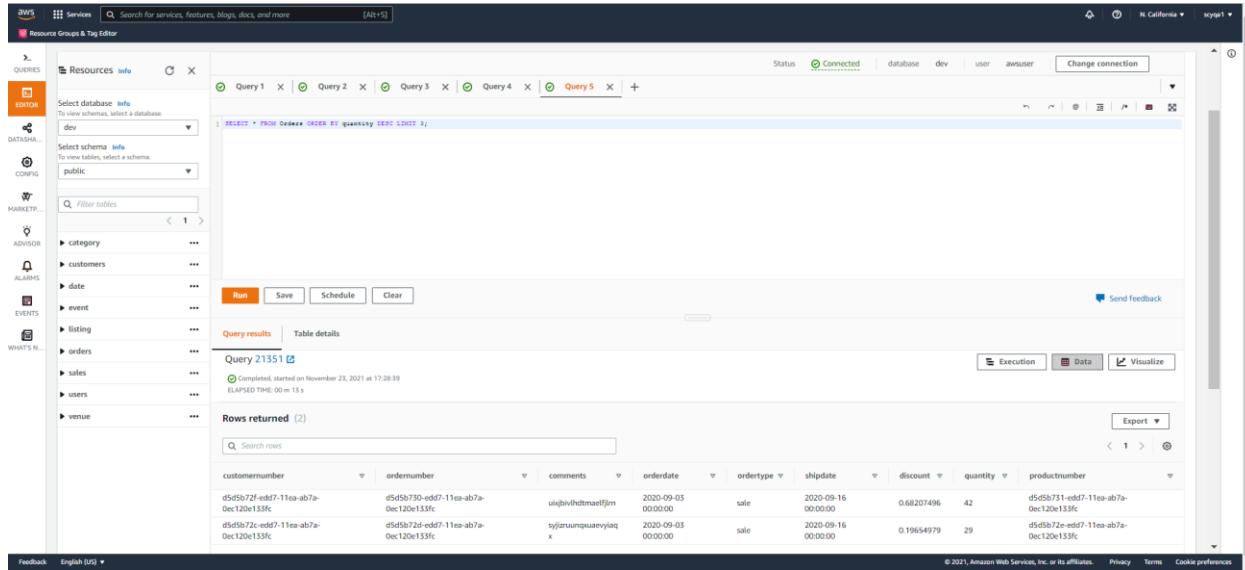
The screenshot shows the AWS Redshift Query Editor interface. The 'Query 3' tab is active, containing the following SQL code:

```
select * from orders;
```

Below the code, there are buttons for 'Run', 'Save', 'Schedule', and 'Clear'. The 'Run' button is highlighted in orange. To the right of the code, there's a 'Send feedback' link. At the bottom, it says 'Query 21252' with a status of 'Completed, started on November 23, 2021 at 17:24:01' and an 'ELAPSED TIME: 00 m 07 s'. There are tabs for 'Execution', 'Data', and 'Visualize'. Under the 'Data' tab, it says 'Rows returned [2]' and shows a table with the following data:

customernumber	ordernumber	comments	orderdate	ordertype	shipdate	discount	quantity	productnumber
d5e5b72e-edd7-11ea-ab7a-0ec120e133fc	d5e5b72e-edd7-11ea-ab7a-0ec120e133fc	syjzrunquuevyiaq x	2020-09-03 00:00:00	sale	2020-09-16 00:00:00	0.19654979	29	d5e5b72e-edd7-11ea-ab7a-0ec120e133fc
d5e5b72e-edd7-11ea-ab7a-0ec120e133fc	d5e5b730-edd7-11ea-ab7a-0ec120e133fc	uajbvihmtmeifjm	2020-09-03 00:00:00	sale	2020-09-16 00:00:00	0.68207496	42	d5e5b711-edd7-11ea-ab7a-0ec120e133fc

Query 2



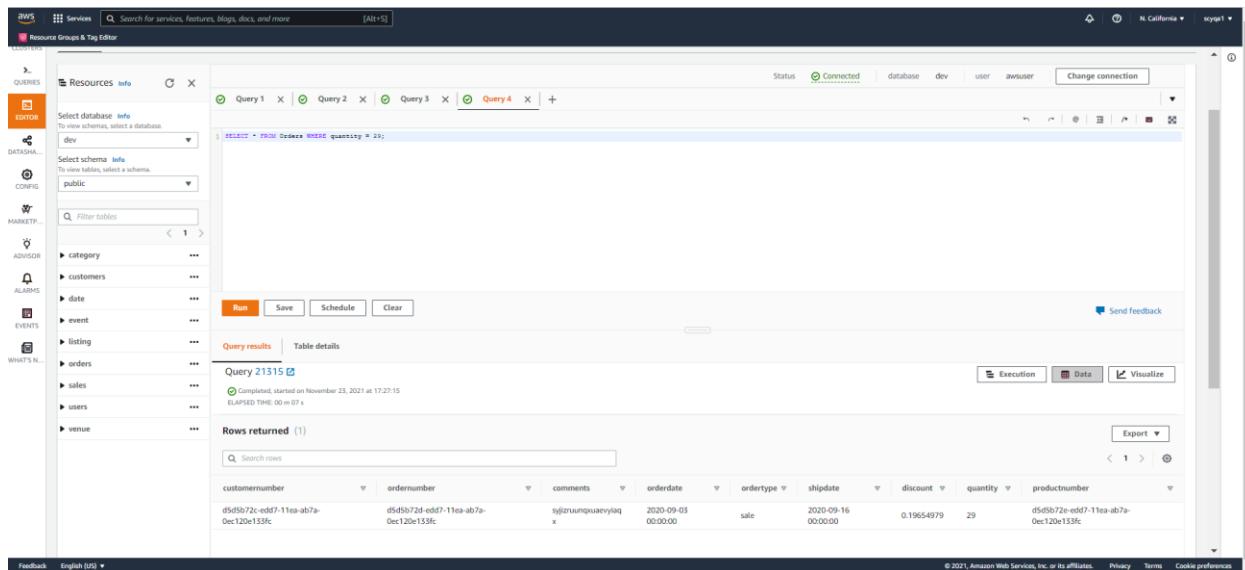
The screenshot shows the AWS Lambda SQL interface. The left sidebar has sections for QUERIES, EDITOR, DATABASES, CONFIG, MARKETPLACE, ADVISOR, ALARMS, EVENTS, and WHAT'S NEW. The main area shows a list of queries: Query 1, Query 2, Query 3, Query 4, and Query 5. Query 5 is selected and contains the following SQL code:

```
1 SELECT * FROM Orders ORDER BY quantity DESC LIMIT 5;
```

The results for Query 5 are displayed in a table:

customername	ordernumber	comments	orderdate	ordertype	shipdate	discount	quantity	productnumber
d5d5b72f-edd7-11ea-ab7a-0ec120e133fc	d5d5b730-edd7-11ea-ab7a-0ec120e133fc	uijblvhdmtafljm	2020-09-05 00:00:00	sale	2020-09-16 00:00:00	0.68207496	42	d5d5b731-edd7-11ea-ab7a-0ec120e133fc
d5d5b72c-edd7-11ea-ab7a-0ec120e133fc	d5d5b732-edd7-11ea-ab7a-0ec120e133fc	syjizunquueyiaq	2020-09-05 00:00:00	sale	2020-09-16 00:00:00	0.19654979	29	d5d5b72e-edd7-11ea-ab7a-0ec120e133fc

Query 3



The screenshot shows the AWS Lambda SQL interface. The left sidebar has sections for QUERIES, EDITOR, DATABASES, CONFIG, MARKETPLACE, ADVISOR, ALARMS, EVENTS, and WHAT'S NEW. The main area shows a list of queries: Query 1, Query 2, Query 3, and Query 4. Query 4 is selected and contains the following SQL code:

```
1 SELECT * FROM Orders WHERE quantity = 29;
```

The results for Query 4 are displayed in a table:

customername	ordernumber	comments	orderdate	ordertype	shipdate	discount	quantity	productnumber
d5d5b72f-edd7-11ea-ab7a-0ec120e133fc	d5d5b732-edd7-11ea-ab7a-0ec120e133fc	syjizunquueyiaq	2020-09-05 00:00:00	sale	2020-09-16 00:00:00	0.19654979	29	d5d5b72e-edd7-11ea-ab7a-0ec120e133fc

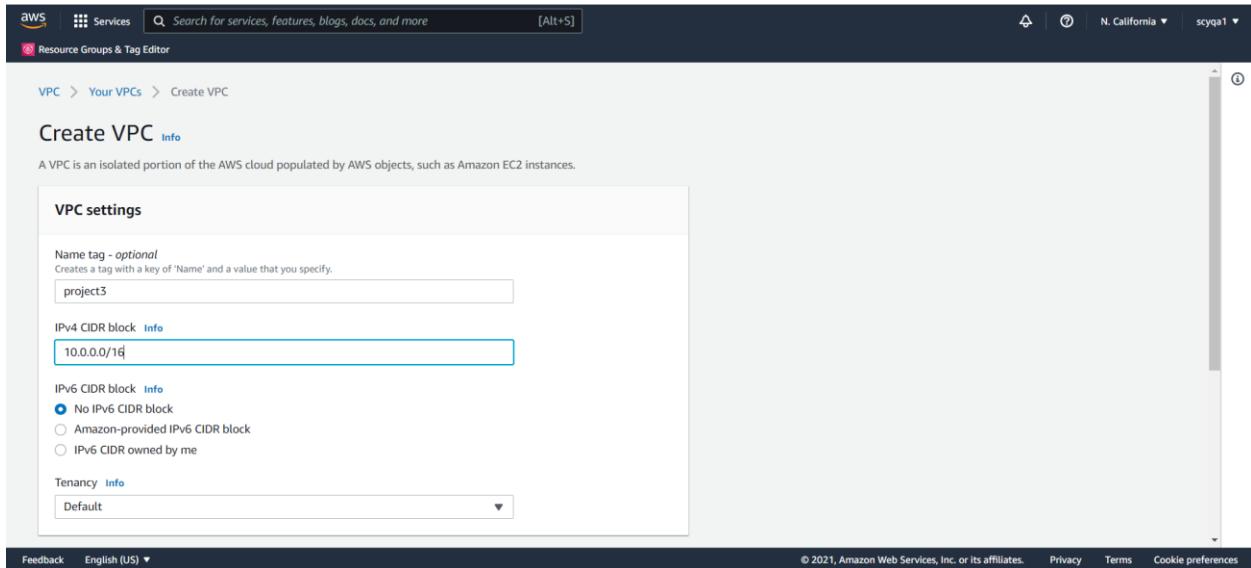
PART II: Build a basic Web App and on AWS [100 points][Coding is required]

Please study this example of building a complete Web App on AWS, using the below infrastructure (IaaS):

<https://medium.com/the-andela-way/designing-a-three-tier-architecture-in-aws-e5c24671f124>

Step 1: Create your IaaS for the Web App: For the IaaS (infrastructure) part, replicate all the steps shown above to create the VMs you need.

Create VPC



Setup the Internet Gateway

The screenshot shows the 'Create internet gateway' page in the AWS VPC service. At the top, there's a navigation bar with the AWS logo, services dropdown, search bar, and account information (N. California, scyqa1). Below the navigation is a breadcrumb trail: VPC > Internet gateways > Create internet gateway.

The main section is titled 'Internet gateway settings'. It includes a 'Name tag' field containing 'project3-IG'. A 'Tags - optional' section shows one tag: 'Name' with value 'project3-IG'. There are buttons for 'Add new tag' and a note about adding up to 49 more tags.

At the bottom of the page, there are links for 'Feedback', 'English (US)', and copyright information: © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

The screenshot shows the 'igw-0857da4096517b41c / project3-IG' page in the AWS VPC service. The left sidebar shows the 'New VPC Experience' is selected, along with 'VPC Dashboard', 'EC2 Global View', 'Filter by VPC', and 'Select a VPC'. Under 'VIRTUAL PRIVATE CLOUD', it lists 'Your VPCs', 'Subnets', 'Route Tables', and 'Internet Gateways'. The 'Internet Gateways' section is expanded, showing 'Egress Only Internet Gateways', 'DHCP Options Sets', 'Elastic IPs', 'Managed Prefix Lists', 'Endpoints', 'Endpoint Services', and 'NAT Gateways'. The right side displays the 'Details' tab for the created internet gateway, showing the Internet gateway ID (igw-0857da4096517b41c), State (Detached), VPC ID (-), and Owner (008889205193). The 'Actions' dropdown menu is open. Below the details, there's a 'Tags' section with a table:

Key	Value
Name	project3-IG

At the bottom, there are links for 'Feedback', 'English (US)', and copyright information: © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

Attach the VPC to the internet gateway

The screenshot shows the AWS VPC Internet Gateways page. A success message at the top states: "The following internet gateway was created: igw-0857da4096517b41c . You can now attach to a VPC to enable the VPC to communicate with the internet." Below this, the internet gateway details are shown: Internet gateway ID is igw-0857da4096517b41c, State is Detached, VPC ID is -, and Owner is 008889205193. The "Actions" menu includes options like Attach to VPC, Detach from VPC, Manage tags, and Delete. On the left sidebar, under the "Internet Gateways" section, the "Attach to VPC" button is highlighted.

The screenshot shows the "Attach to VPC" dialog box. It asks to attach an internet gateway to a VPC to enable communication with the internet. The "Available VPCs" section lists "Select a VPC" and "vpc-0ec7532b335f184d1 - project3". At the bottom are "Cancel" and "Attach internet gateway" buttons. The URL in the browser bar is "VPC > Internet gateways > Attach to VPC (igw-0857da4096517b41c)".

Create 4 Subnets

The screenshot shows the AWS VPC Subnet creation interface. In the 'VPC ID' section, 'vpc-0ec7532b335f184d1 (project3)' is selected. Under 'Associated VPC CIDRs', the CIDR '10.0.0.0/16' is listed. In the 'Subnet settings' section, 'Subnet 1 of 1' is being configured. The 'Subnet name' field contains 'project3-subnet1'. The 'Availability Zone' dropdown is set to 'US West (N. California) / us-west-1b'. The 'IPv4 CIDR block' dropdown shows 'Q 10.0.0.0/16'. A note indicates that the name can be up to 256 characters long. The interface includes standard AWS navigation and footer elements.

This screenshot shows the continuation of subnet creation. The 'Associated VPC CIDRs' section still lists '10.0.0.0/16'. In the 'Subnet settings' section for 'Subnet 1 of 1', the 'Subnet name' is now 'project3-subnet2'. The 'Availability Zone' is set to 'US West (N. California) / us-west-1b'. The 'IPv4 CIDR block' dropdown shows 'Q 10.0.0.0/24'. A 'Tags - optional' section is visible, showing a key-value pair: 'Name' and 'project3-subnet2'. The interface includes standard AWS navigation and footer elements.

The screenshot shows the AWS VPC Subnet list. A green banner at the top right says 'You have successfully created 1 subnet: subnet-off7b670d1d6d9665'. The main table displays seven subnets:

ID	Name	CIDR Block	Status	IP Range	Actions
subnet-054582c8f23f9f25f	project3-subnet1	10.0.1.0/24	Available	vpc-0ec7532b335f184d1 pro...	...
subnet-09184029a82bab400	project3-subnet2	10.0.2.0/24	Available	vpc-0ec7532b335f184d1 pro...	...
subnet-0dc3b2c570430b11d	project3-subnet3	10.0.3.0/24	Available	vpc-0ec7532b335f184d1 pro...	...
subnet-off7b670d1d6d9665	project3-subnet4	10.0.4.0/24	Available	vpc-0ec7532b335f184d1 pro...	...
subnet-0d446f056ffe859c2	Public subnet	10.0.0.0/24	Available	vpc-0ef7df9a77d00bbb2	...

The sidebar on the left shows the VPC navigation menu, and the bottom includes standard AWS footer links.

Create Two Route Tables

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

N. California scyqa1

VPC > Route tables > Create route table

Create route table [Info](#)

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

VPC
The VPC to use for this route table.

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key Value - optional Remove

Feedback English (US) © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

N. California scyqa1

VPC > Route tables > Create route table [Info](#)

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

VPC
The VPC to use for this route table.

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key Value - optional Remove

Add new tag

Feedback English (US) © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

N. California scyqa1

New VPC Experience Tell us what you think

Select a VPC

Route tables (7) [Info](#)

Route tables (7) [Actions](#) [Create route table](#)

Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC	Ow...
project3-public-RT	rtb-0f516c7b0ce1dce14	-	-	No	vpc-0ec7532b335f184d1 pro...	00888.
project3-private-RT	rtb-0d4203de948327b7d	-	-	No	vpc-0ec7532b335f184d1 pro...	00888.
Project1_routeTable	rtb-02fedf20b348fbfe2	-	-	No	vpc-c57ad0a3	00888.
-	rtb-09d040d2318302c0	-	-	Yes	vpc-0ec7532b335f184d1 nrm...	00888.

Select a route table

Feedback English (US) © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Subnet Association

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

N. California scyqa1

Route tables (1/7) Info

Filter route tables

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Ow...
<input checked="" type="checkbox"/> project3-public-RT	rtb-0f516c7b0ce1dce14	-	-	No	vpc-0ec7532b335f184d1 pro...	00888.
<input type="checkbox"/> project3-private-RT	rtb-0d4203de948327b7d	-	-	No	vpc-0ec7532b335f184d1 pro...	00888.
<input type="checkbox"/> Project1_routeTable	rtb-02fedf29b348fbfe2	-	-	No	vpc-c57ad0a3	00888.

rtb-0f516c7b0ce1dce14 / project3-public-RT

Details Routes Subnet associations Edge associations Route propagation Tags

Explicit subnet associations (0)

Find subnet association

Edit subnet associations

Subnet ID IPv4 CIDR IPv6 CIDR

No subnet associations

© 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

N. California scyqa1

VPC > Route tables > rtb-0d4203de948327b7d > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (2/4)

Filter subnet associations

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input type="checkbox"/> project3-subnet2	subnet-09184029a82bab400	10.0.2.0/24	-	rtb-0f516c7b0ce1dce14 / project3-public-RT
<input checked="" type="checkbox"/> project3-subnet3	subnet-0dc3b2c570430b11d	10.0.3.0/24	-	Main (rtb-08d0d30d2318302c0)
<input checked="" type="checkbox"/> project3-subnet4	subnet-0ff7b670d1d6d9665	10.0.4.0/24	-	Main (rtb-08d0d30d2318302c0)
<input type="checkbox"/> project3-subnet1	subnet-054582c8f23f9f25f	10.0.1.0/24	-	rtb-0f516c7b0ce1dce14 / project3-public-RT

Selected subnets

subnet-0ff7b670d1d6d9665 / project3-subnet4 subnet-0dc3b2c570430b11d / project3-subnet3

https://us-west-1.console.aws.amazon.com/vpc/home?region=us-west-1#subnets:SubnetId=...

© 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

N. California scyqa1

You have successfully updated subnet associations for rtb-0d4203de948327b7d / project3-private-RT.

Route tables (7) Info

Filter route tables

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Ow...
<input type="checkbox"/> project3-public-RT	rtb-0f516c7b0ce1dce14	2 subnets	-	No	vpc-0ec7532b335f184d1 pro...	00888.
<input type="checkbox"/> project3-private-RT	rtb-0d4203de948327b7d	2 subnets	-	No	vpc-0ec7532b335f184d1 pro...	00888.

Select a route table

© 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Route the traffic to the internet through the internet gateway

Screenshot of the AWS VPC Route Tables page showing two route tables: project3-public-RT and project3-private-RT. A success message indicates subnet associations were updated.

Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC
project3-public-RT	rtb-0f516c7b0ce1dce14	2 subnets	—	No	vpc-0ec7532b335f184d1 project3
project3-private-RT	rtb-0d4203de948327b7d	2 subnets	—	No	vpc-0ec7532b335f184d1 project3

Screenshot of the 'Edit routes' dialog for the project3-public-RT route table. It shows a route from 10.0.0.0/16 to 'local' and another to 'igw-0857da4096517b41c'. Both routes are active and not propagated.

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	igw-0857da4096517b41c	—	No

Screenshot of the AWS VPC Route Tables page showing the updated routes for the project3-public-RT route table. The route to 'igw-0857da4096517b41c' is now listed under 'Routes'.

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	igw-0857da4096517b41c	Active	No

Create the NAT Gateway

The screenshot shows the AWS VPC service interface for creating a NAT gateway. In the top navigation bar, 'Services' is selected. The search bar contains 'Search for services, features, blogs, docs, and more'. The location is set to 'N. California' and the user is 'scyqa1'. A progress bar at the bottom indicates '正在等待 us-west-1.console.aws.amazon.com 的响应...'.

Create NAT gateway [Info](#)

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

Subnet
Select a subnet in which to create the NAT gateway.

Connectivity type
Select a connectivity type for the NAT gateway.
 Public
 Private

Elastic IP allocation ID [Info](#)
Assign an Elastic IP address to the NAT gateway.
 [Allocate Elastic IP](#)

Feedback English (US) © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

NAT gateway nat-063510f5161999298 | project3-NAT was created successfully.

VPC > NAT gateways > nat-063510f5161999298 / project3-NAT

Details [Info](#)

NAT gateway ID	Connectivity type	State	State message Info
nat-063510f5161999298	Public	Pending	-
Elastic IP address	Private IP address	Network interface ID	VPC
-	-	-	vpc-0ec7532b335f184d1 / project3
Subnet	Created	Deleted	
subnet-09184029a82bab400 / project3-subnet2	2021/11/23 18:24 GMT-8	-	

[Monitoring](#) [Tags](#)

© 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Edit the private route table to make use of the NAT gateway to access the internet

The screenshot shows the AWS VPC Route Tables page. On the left sidebar, under 'Route Tables New', 'Route Tables' is selected. In the main content area, the 'Route tables (1/7)' section is displayed with a table. One row is selected: 'project3-private-RT' (rtb-0d4203de948327b7d). The table columns include Name, Route table ID, Explicit subnet associations, Edge associations, Main, VPC, and Owner. Below the table, the 'Routes (1)' section shows a single route: Destination 10.0.0.0/16, Target local, Status Active, and Propagated No.

The screenshot shows the 'Edit routes' dialog for the selected route table. It lists two routes: one to 'local' and another to a NAT gateway with ID 'nat-063510f5161999298'. The 'Add route' button is visible at the bottom left, and 'Save changes' is at the bottom right.

aws Services Search for services, features, blogs, docs, and more [Alt+S]

Resource Groups & Tag Editor

New VPC Experience Tell us what you think

VPC Dashboard

EC2 Global View New

Filter by VPC:

Select a VPC

VIRTUAL PRIVATE CLOUD

Your VPCs

Subnets

Route Tables New

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Managed Prefix Lists

Endpoints

Endpoint Services

NAT Gateways

Updated routes for rtb-0d4203de948327b7d / project3-private-RT successfully

Details

VPC > Route tables > rtb-0d4203de948327b7d

rtb-0d4203de948327b7d / project3-private-RT

Actions

You can now check network connectivity with Reachability Analyzer

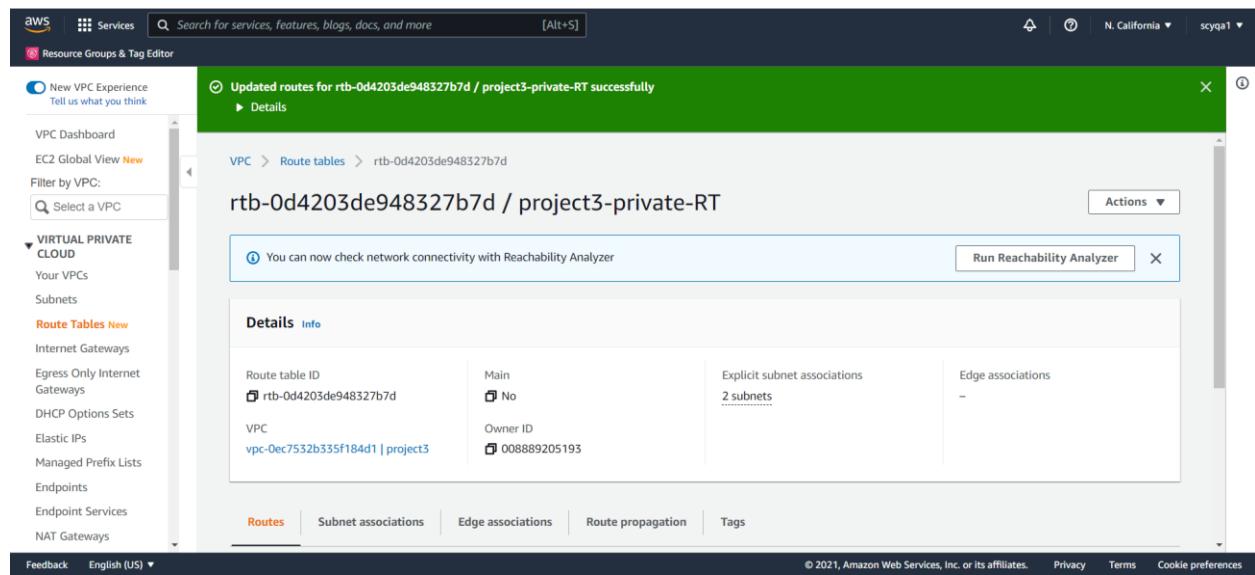
Run Reachability Analyzer

Details Info

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-0d4203de948327b7d	No	2 subnets	-
VPC	Owner ID	008889205193	

Routes Subnet associations Edge associations Route propagation Tags

Feedback English (US) © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



Create Elastic Load Balancer

Screenshot of the AWS Services search results for "Load balancer type".

The search bar shows "Search for services, features, blogs, docs, and more" with the keyboard shortcut "[Alt+S]".

A message at the top says "Select load balancer type" and "A complete feature-by-feature comparison along with detailed highlights is also available. Learn more".

The "Load balancer types" section displays three options:

- Application Load Balancer**: Info. Diagram shows traffic from a client through an ALB (HTTP to HTTPS) to Lambda targets.
- Network Load Balancer**: Info. Diagram shows traffic from a client through a VPC, NLB (TCP, UDP, TLS), and ALB to Lambda, Container, and IP targets.
- Gateway Load Balancer**: Info. Diagram shows traffic from a client through a GWLB to Lambda, Container, and IP targets, including security features like TLS and Firewall.

At the bottom, there are links for "Feedback", "English (US)", "Privacy", "Terms", and "Cookie preferences".

Screenshot of the "How Application Load Balancers work" configuration page.

The search bar shows "Search for services, features, blogs, docs, and more" with the keyboard shortcut "[Alt+S]".

A message at the top says "► How Application Load Balancers work".

The "Basic configuration" section includes:

- Load balancer name**: project3-Frontend-LB. A note says "Name must be unique within your AWS account and cannot be changed after the load balancer is created."
- Scheme**: Info. A note says "Scheme cannot be changed after the load balancer is created." Options:
 - Internet-facing**: An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#).
 - Internal**: An internal load balancer routes requests from clients to targets using private IP addresses.
- IP address type**: Info. A note says "Select the type of IP addresses that your subnets use." Options:
 - IPv4**: Recommended for internal load balancers.
 - Dualstack**: Includes IPv4 and IPv6 addresses.

At the bottom, there are links for "Feedback", "English (US)", "Privacy", "Terms", and "Cookie preferences".

Successfully created load balancer: project3-Frontend-LB

Note: It might take a few minutes for your load balancer to be fully set up and ready to route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks.

EC2 > Load balancers

Suggested next steps

- Review, customize, or enable attributes for your load balancer and listeners using the **Description** and **Listeners** tabs within **project3-Frontend-LB**.
- Discover other services that you can integrate with your load balancer. Visit the **Integrated services** tab within **project3-Frontend-LB**.

View load balancer

Basic configuration

Load balancer name
Name must be unique within your AWS account and cannot be changed after the load balancer is created.
project3-backend-LB
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme info
Scheme cannot be changed after the load balancer is created.
 Internet-facing
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)
 Internal
An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type [Info](#)
Select the type of IP addresses that your subnets use.
 IPv4
Recommended for internal load balancers.
 Dualstack
Includes IPv4 and IPv6 addresses.

Network mapping [Info](#)

Name	DNS name	State	VPC ID	Availability Zones	Type	Created
project3-backend-LB	internal-project3-backend-LB...	Provisioning	vpc-0ec7532b335f184d1	us-west-1a, us-west-1b	application	November
project3-Frontend-LB	project3-Frontend-LB-13475...	Provisioning	vpc-0ec7532b335f184d1	us-west-1a, us-west-1b	application	November

Select a load balancer

Bastion Host

Screenshot of the AWS CloudFormation console showing the 'Step 7: Review Instance Launch' step. The page displays the configuration for launching an instance, including the AMI, instance type, and security group.

Step 7: Review Instance Launch

AMI Details: Amazon Linux 2 AMI (HVM) - Kernel 4.14, SSD Volume Type - ami-0074ef78ecb07948c

Instance Type: t2.micro

Security Groups: Select an instance above

Launch button

Instances (1) Info:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
-	i-07852bc81d0ca5ef0	Pending	t2.micro	-	No alarms	us-west-1a	-

Step 2: deploy a Web App Then, deploy a basic Web app (SaaS) on this IaaS. The App can be the same as the Java eCommerce app you previously built, OR any other very simple web app you can adopt. Goal here is to deploy a basic web app on the IaaS you build in Step 1.

Before deployment, I would like to describe the design of the 3 tiers of the web application first.

For the Presentation tier, I would use Django as the eCommerce app I previously built.

For the App tier, I would use Zappa and AWS Lambda to deploy serverless Django web application. Also, I would use API calls to communicate with the data tier for this tier.

For the Data tier, I would use Amazon RDS.

For this SaaS web application, I encountered many difficulties - debugging is a great obstacle for me, even beyond my ability to address. There isn't any reference for this part. Finally, there are still some parts I can't deploy successfully.

Django eCommerce web app:

Before coding, create a virtual environment to manage dependencies.

Shell

```
$ python3 -m venv venv
```

Shell

```
$ python3 -m venv venv
```

Then activate the virtual environment.

Windows Console

```
C:\> venv\Scripts\activate.bat
```

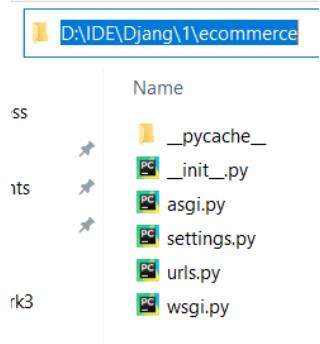
Install Django

Shell

```
(venv) $ pip install Django
```

Then, create a Django Project with command below:

```
$ django-admin startproject ecommerce
```



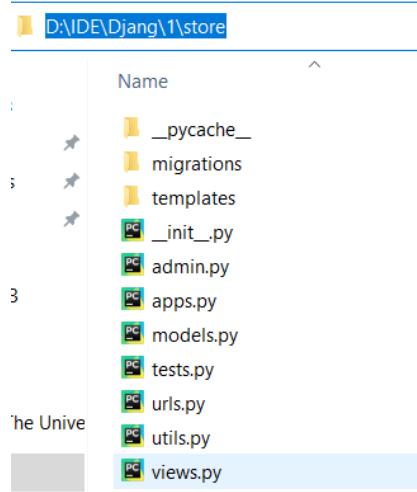
migrate admin database with command below:

```
$ python manage.py migrate
```

```
$ python manage.py makemigrations
```

Create a Django Application with command below:

```
$ python manage.py startapp
```



Running the development server with command below:

```
$ python manage.py runserver
```

manage.py

```
D:\IDE\DJANGO\1> manage.py
 1  #!/usr/bin/env python
 2  """Django's command-line utility for administrative tasks."""
 3  import os
 4  import sys
 5
 6
 7  def main():
 8      os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'ecommerce.settings')
 9      try:
10          from django.core.management import execute_from_command_line
11      except ImportError as exc:
12          raise ImportError(
13              "Couldn't import Django. Are you sure it's installed and "
14              "available on your PYTHONPATH environment variable? Did you "
15              "forget to activate a virtual environment?"
16          ) from exc
17      execute_from_command_line(sys.argv)
18
19
20  if __name__ == '__main__':
21      main()
22
23
```

Templates

D:\IDE\DJANGO\1\store\templates\store	
	Name
;	cart.html
5	main.html
	store.html

Use Zappa and AWS Lambda to deploy serverless Django web application

Activate the virtual environment

```
C:\D:\>cd D:\IDE\DJang\DJango  
D:\IDE\DJang\DJango>venv\Scripts\activate.bat
```

```
(venv) D:\IDE\DJang\DJango>
```

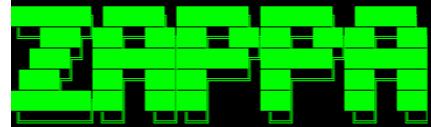
Install kappa (0.6.0)

```
(venv) D:\IDE\DJang\DJango>cd D:\IDE\DJang\DJango\kappa-0.6.0\kappa-0.6.0  
(venv) D:\IDE\DJang\DJango\kappa-0.6.0\kappa-0.6.0>python setup.py install  
Warning: 'classifiers' should be a list, got type 'tuple'
```

Install zappa (0.53.0)

```
(venv) D:\IDE\DJang\DJango\kappa-0.6.0\kappa-0.6.0>cd D:\IDE\DJang\DJango  
(venv) D:\IDE\DJang\DJango>pip install zappa==0.53.0  
Collecting zappa==0.53.0
```

Zappa initialization

```
(venv) D:\IDE\DJang\DJango>zappa init  
d:\ide\pycharm\anaconda\envs\site-packages\setuptools\distutils_patch.py:25: UserWarning: Distutils was imported before Setuptools. This usage is discouraged and may exhibit undesirable behaviors or errors. Please use Setuptools' object directly or at least import Setuptools first.  
    warnings.warn(  
  
Welcome to Zappa!
```

Zappa deployment

```
(venv) D:\IDE\DJang\DJango>zappa deploy dev  
Important! A new version of Zappa is available!
```

Then, configure the api gateway to the 'settings.py'

```
ALLOWED_HOSTS = ['127.0.0.1', 'oakh71wulb.execute-api.us-west-1.amazonaws.com', ]
```

Zappa update

```
D:\IDE\ Django\ Django\ ecommerce>zappa update
```

Some other configurations

1. change add_description to set_description

```
    self.cf_template = troposphere.Template()
    self.cf_template.set_description("Automatically generated with Zappa")
    self.cf_template.add_resource(...)
```

2. zappa_settings.json

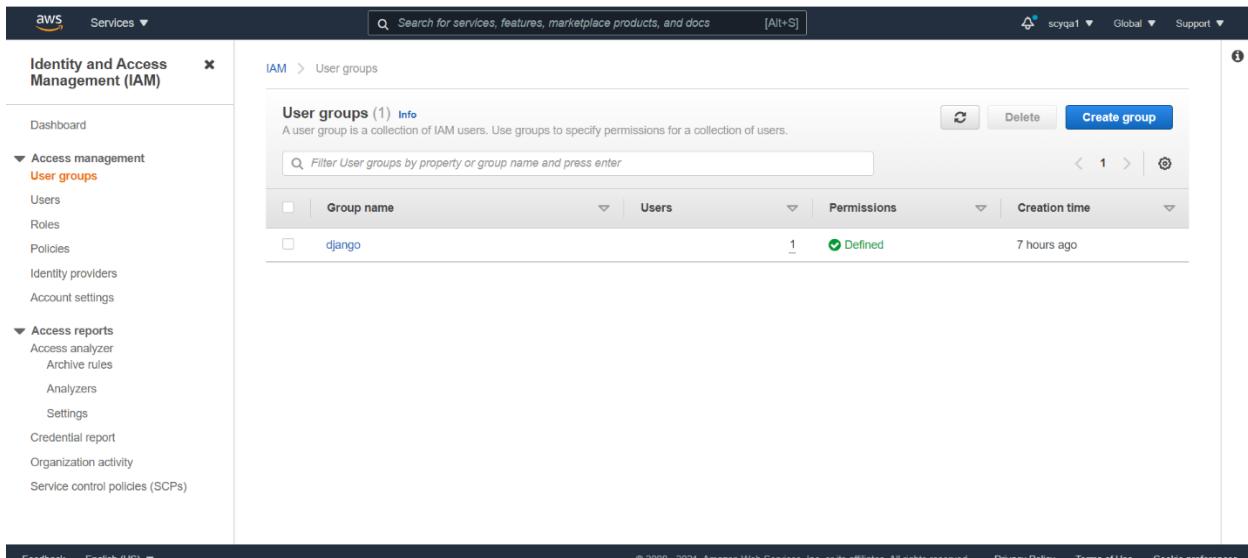
```
{
  "dev": {
    "django_settings": "ecommerce.settings",
    "profile_name": "default",
    "project_name": "django",
    "aws_region": "us-west-1",
    "runtime": "python3.8",
    "s3_bucket": "zappa-tn4u6j73t"
  }
}
```

3. credentials

This PC > Local Disk (C:) > Users > Administrator > .aws				
Name	Date modified	Type	Size	
credentials	2021/10/15 23:55	File	1 KB	

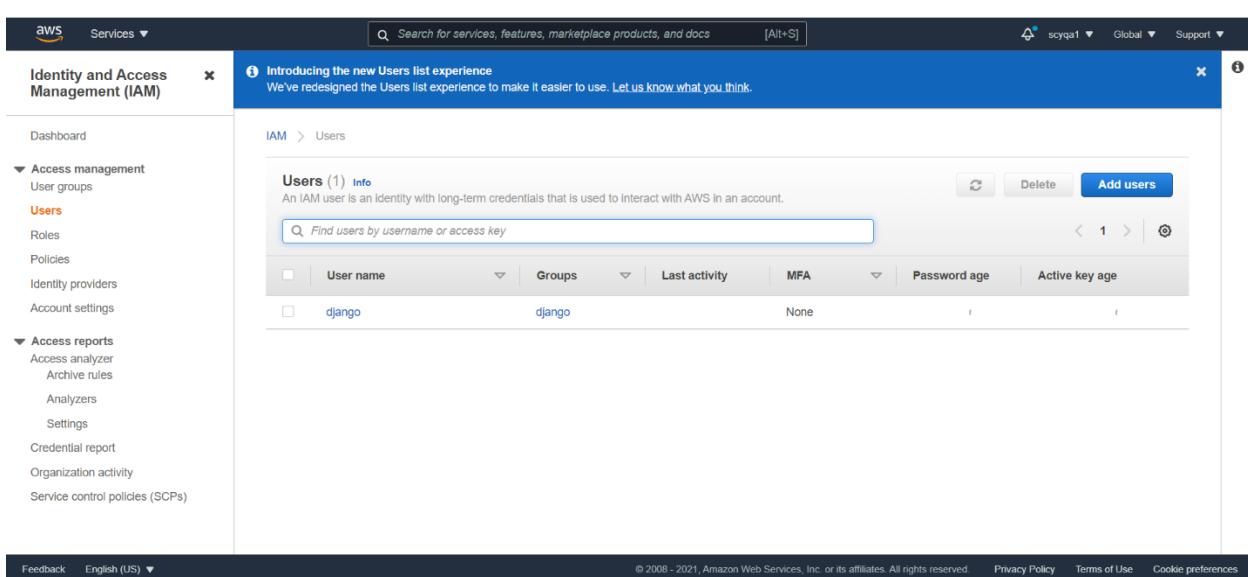
```
[[default]
aws_access_key_id = AKIAQEEOWQHERVWM3N7U
aws_secret_access_key = lmo/XX7HXs/Rd8iuXb1GemXSAmLeKuTVczodv5D]
```

4. IAM



The screenshot shows the AWS IAM User Groups page. The left sidebar has sections for Dashboard, Access management (User groups, Roles, Policies, Identity providers, Account settings), and Access reports (Access analyzer, Archive rules, Analyzers, Settings, Credential report, Organization activity, Service control policies (SCPs)). The main content area shows 'User groups (1) Info'. A user group is a collection of IAM users. Use groups to specify permissions for a collection of users. There is a search bar 'Filter User groups by property or group name and press enter'. A table lists one group: 'django' (Group name), '1' (Users), 'Defined' (Permissions), and '7 hours ago' (Creation time). Action buttons include 'Create group', 'Delete', and a refresh icon.

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences



The screenshot shows the AWS IAM Users page. The left sidebar has sections for Dashboard, Access management (User groups, Users, Roles, Policies, Identity providers, Account settings), and Access reports (Access analyzer, Archive rules, Analyzers, Settings, Credential report, Organization activity, Service control policies (SCPs)). A blue banner at the top says 'Introducing the new Users list experience. We've redesigned the Users list experience to make it easier to use. Let us know what you think.' The main content area shows 'Users (1) Info'. An IAM user is an identity with long-term credentials that is used to interact with AWS in an account. There is a search bar 'Find users by username or access key'. A table lists one user: 'django' (User name), 'django' (Groups), 'None' (MFA), and 'None' (Password age, Active key age). Action buttons include 'Delete', 'Add users', and a refresh icon.

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Final Result:

oakh71wulb.execute-api.us-west-1.amazonaws.com/cart

Ecom Store Hi, Could6620_Qichen 1

The screenshot shows a grid of nine items arranged in three rows of three. Each item has a thumbnail, a name, a price, and two buttons: 'Add to Cart' and 'View'. The items are:

Item	Price
Headphones	\$179.99
Mount of Olive's Book	\$14.99
Project Source Code	\$19.99
Watch	\$259.0
Shoes	\$89.99
T-Shirt	\$25.99

Ecom Store Hi, Could6620_Qichen 1

← Continue Shopping

Items: 1 Total: \$19.99 Checkout

The screenshot shows a summary of the cart with one item: 'Project Source Code' at \$19.99. The quantity is set to 1.

Item	Price	Quantity	Total
	\$19.99	1	\$19.99

Item	Price	Quantity	Total
	\$19.99	3	\$59.97

Communicates with the data tier using API

Settings:

```
165 print("">>>> ENVIRONMENT VARIABLES" + str(os.environ))
166 if 'RDS_HOSTNAME' in os.environ:
167     DATABASES = {
168         'default': {
169             'ENGINE': 'django.db.backends.mysql',
170             'NAME': os.environ['RDS_DB_NAME'],
171             'USER': os.environ['RDS_USERNAME'],
172             'PASSWORD': os.environ['RDS_PASSWORD'],
173             'HOST': os.environ['RDS_HOSTNAME'],
174             'PORT': os.environ['RDS_PORT'],
175             # 'OPTIONS': {
176             #     'ssl': {
177             #         'ca': os.getcwd() + '/rds-ca-2019-root.pem'
178             #     }
179             # }
180         }
181     }
182 else:
183     try:
184         DATABASES = {
185             'default': {
186                 'ENGINE': 'django.db.backends.mysql',
187                 'NAME': rds_configs['db-name'],
188                 'USER': os.getenv('RDS_USER'),
189                 'PASSWORD': os.getenv('RDS_PASSWORD'),
190                 'HOST': os.getenv('RDS_HOST'),
191                 'PORT': rds_configs['rds-port'],
192                 # 'OPTIONS': {
193                 #     'ssl': {
194                 #         'ca': os.getcwd() + '/rds-ca-2019-root.pem'
195                 #     }
196                 # }
197             }
198         }
199     
```

Store data to RDS

Example (Create a new user)

```
....  
....  
  
Ensure we can create a new user object.  
....  
  
url = reverse('user-list')  
data = {  
    "firstName" : "John",  
    "lastName" : "Wick",  
    "partnerCode" : "01377224",  
}  
response = self.client.post(url, data, format='json')  
self.assertEqual(response.status_code, status.HTTP_201_CREATED)  
self.assertEqual(User.objects.count(), 1)  
self.assertEqual(User.objects.get().firstName, 'John')
```