Introduction for DynamoDB

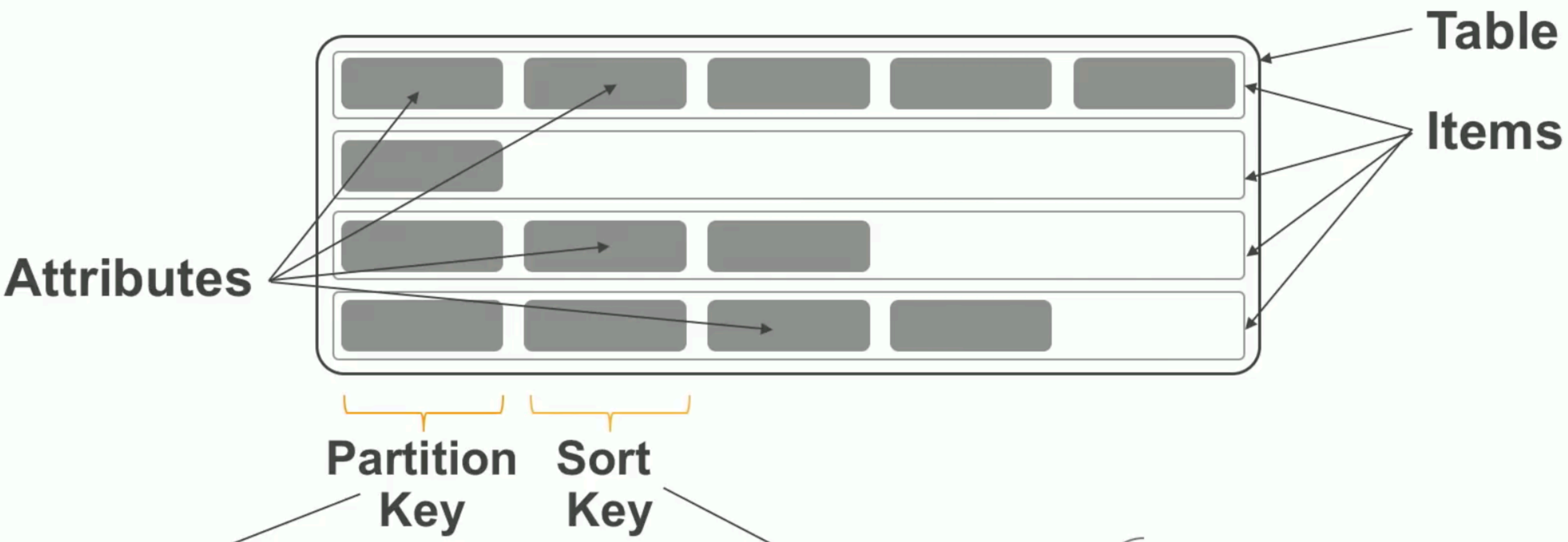
Yina Bao

DynamoDB is a scalable and faster performing Non-relational database service that is completely hosted by Amazon Web Service.

COMPONENTS

- Tables – It’s like the other database tables which contain all the data row-wies.
- Items – Each row in the table represents an Item that containing all the attribute data.
- Attributes – Columns in each Item containing various data of DynamoDB supported data types.

Table



Mandatory
Key-value access pattern
Determines data distribution

Optional
Model 1:N relationships
Enables rich query capabilities

All items for key
==, <, >, >=, <=
“begins with”
“between”
sorted results
top/bottom N values

PRIMARY KEY

- DynamoDB represents its primary key with the combination of a **Partition key** and an optional **Sort key**.
- A primary key can hold only **scalar** values. Supported data types for a primary key are **string, number, and binary**. Other **non-key** attributes are free from this restriction.

CONT.

PRIMARY KEY

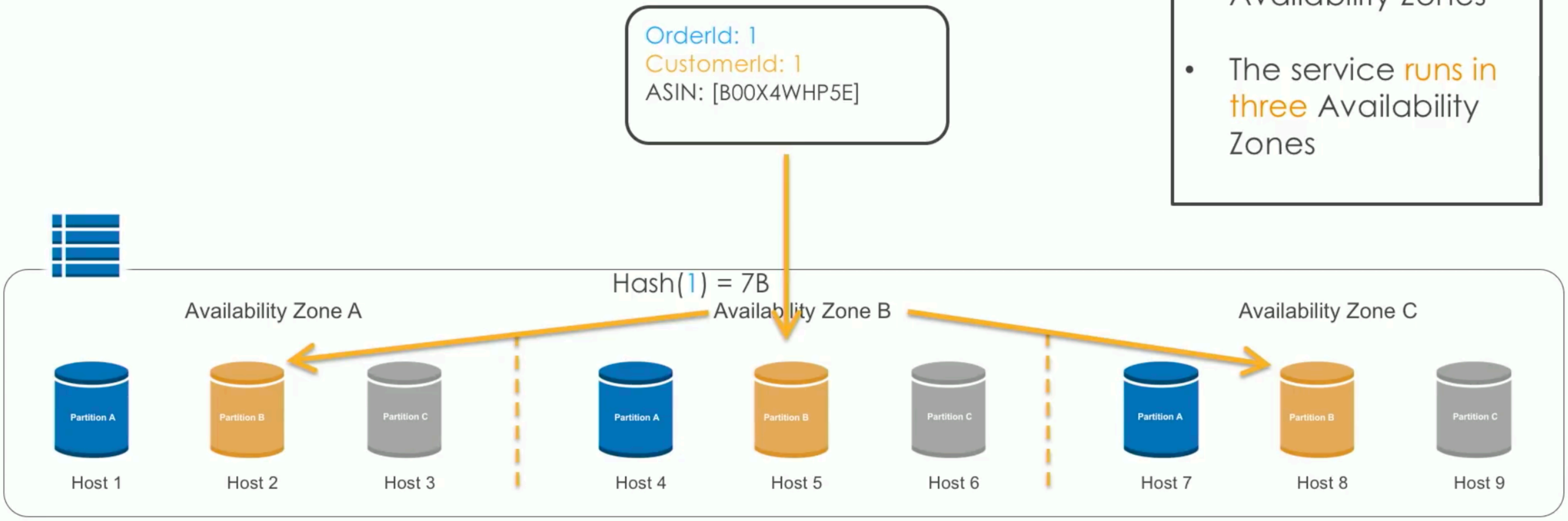
- Partition key – It's an internal hash to point physical storage in the database where the data is stored. Two Items can't hold **identical** partition key If the **primary key** is formed with only the **partition key**.

- Sort key – Sort key is optional to form the primary key but adds some additional advantages in need. The table containing partition key and sort key can have **more than one item** holding the **same partition key**. In that case, the sort key must be **non-identical**. Partition key helps to stores all the items within some **particular shards** and sort key helps to keep those data inside those shards in **sorted order**. By this, the performance of query operations becomes very efficient. Also, sort key is useful to perform range queries with operators like **=, <, >, <=, >=, between and begins_with**.

SUPPORTED DATA TYPES

- Scalar Types – Number, String, Binary, Boolean, Null.
- Document Types – Complex structure with nested attributes like in JSON data format. Usually list and map.
- Set Types – Contains more than one and different scalar types

High availability and durability in DynamoDB

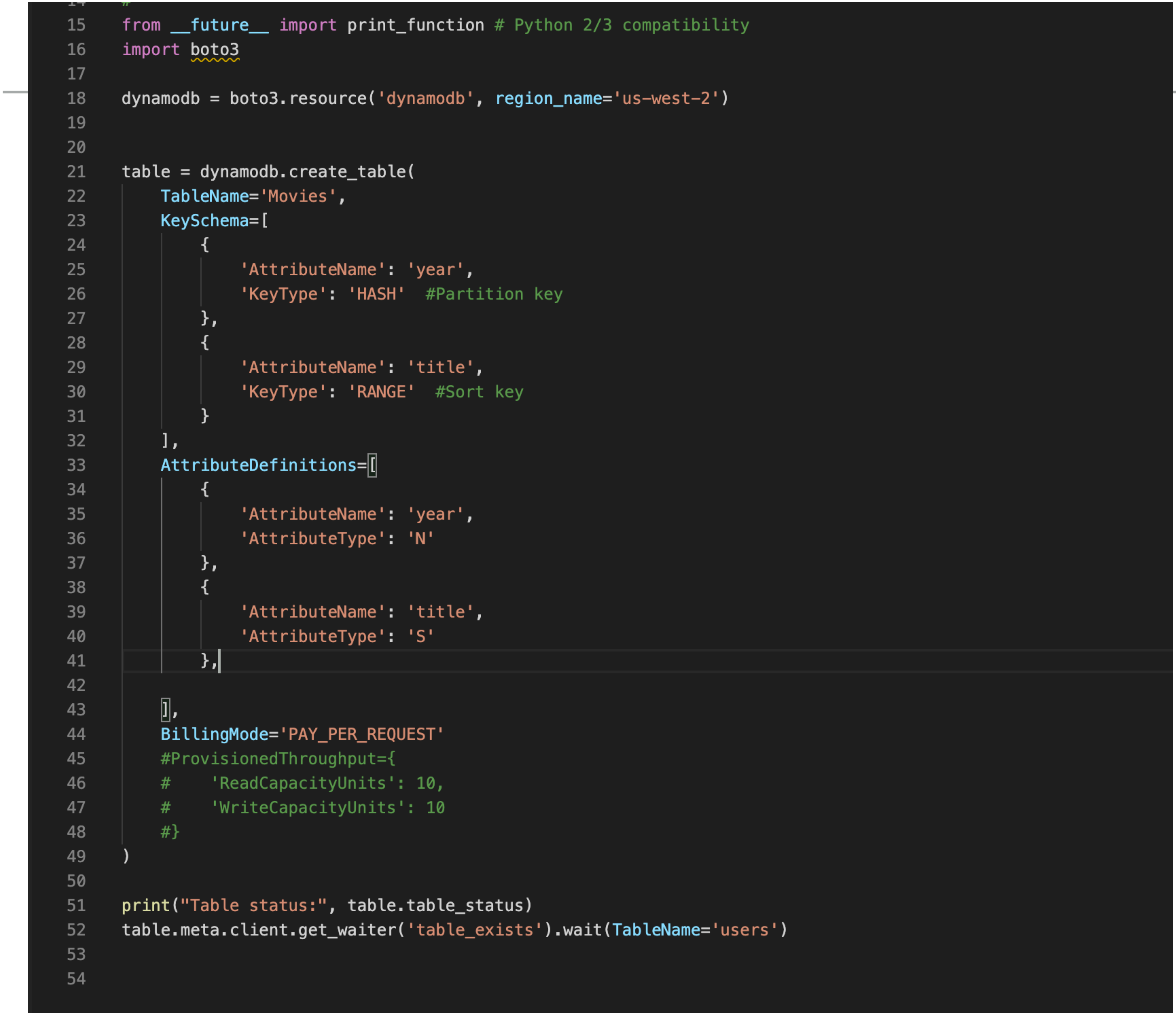


OrderId: 1
CustomerId: 1
ASIN: [B00X4WHP5E]

Hash(1) = 7B

3-way replication

- Data is always replicated to three Availability Zones
- The service runs in three Availability Zones



```
#
from __future__ import print_function # Python 2/3 compatibility
import boto3

dynamodb = boto3.resource('dynamodb', region_name='us-west-2')


table = dynamodb.create_table(
    TableName='Movies',
    KeySchema=[
        {
            'AttributeName': 'year',
            'KeyType': 'HASH'  #Partition key
        },
        {
            'AttributeName': 'title',
            'KeyType': 'RANGE'  #Sort key
        }
    ],
    AttributeDefinitions=[
        {
            'AttributeName': 'year',
            'AttributeType': 'N'
        },
        {
            'AttributeName': 'title',
            'AttributeType': 'S'
        },

    ],
    BillingMode='PAY_PER_REQUEST'
    #ProvisionedThroughput={
    #    'ReadCapacityUnits': 10,
    #    'WriteCapacityUnits': 10
    #}
)

print("Table status:", table.table_status)
table.meta.client.get_waiter('table_exists').wait(TableName='users')


```

- AWS Lambda & DynamoDB
  https://www.youtube.com/watch?v=ijyeE-pXFk0
- Python and DynamoDB
  https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/GettingStarted.Python.html
- AWS Official YouTube Channel
  https://www.youtube.com/user/AmazonWebServices/featured