**Databases Intro:**

* Storing Data on disk (EFS, EBS, EC2, Instance Store, S3) can have its  
  limits
* Storing these data in a database is recommended.
* You can structure the data
* You build indexes to efficiently query/search through the data
* You can define relationships between your datasets

1. **Relationship Databases:**

* Looks just like Excel spreadsheets, with link between the tables.



* Can use the SQL language to perform queries/lookups

1. **NoSQL Databases:**

* NoSQL = no SQL = non-relational databases
* These are modern databases and are purpose built for specific data models and have flexible schemas (schemas are basically the shape of the data) for building modern applications.
* Benefits:
* Flexibility: easy to evolve data model
* Scalability: designed to scale-out by using distributed clusters
* High performance: optimized for specific data model
* Highly functional: types optimized for the data model
* Examples: Key-value, document graph, in-memory, search databases

NoSQL data examples: JSON:

* JSON: JavaScript Object Notation
* JSON is a common form of data that fits into a NoSQL model
* Data can be nested
* Fields can change over time
* Support for new types: arrays, etc...

**DATABASES AND SHARED RESPONSIBILITY ON AWS**

* AWS offers use to manage different databases
* Benefits include:
* Quick provisioning, High Availability, Vertical and Horizontal Scaling
* Automated Backup and Restore, Operations, Upgrades
* Operating system patching is handled by AWS
* Monitoring and Alerting is also integrated

**NOTE:** Many databases’ technologies could be run on EC2, but you must handle yourself the resiliency, backup, patching, high availability, fault tolerance, scaling…