AWS CSA-Pro Reviews ACloudGuru Section 2 Data Stores Part 2

In what scenarios would you want to run a database on an EC2 instance? What extra do you need to do to manage that database?

In what scenarios would you use an RDS database? What are some benefits of it?

For MySQL what engine do you need for a read replica?

How are multi-az databases updated? What about read replicas?

If the master DB dies of a Multi-AZ deployment what happens?

If a whole region fails what do you need to do in order to use a Read Replica as the new master?

When would you use DynamoDB?

What is the difference between relational and NoSQL?

In DynamoDB what are the 3 elements? What does each represent?

What are the 2 ways you can set up your partitions in DynamoDB?

What are the 2 secondary index types? What are their differences? When you should you use each?

If you need to access just a few attributes in the fastest way possible which secondary index should you use?

If you need to frequently access some non-key attributes which secondary index should you use?

If you need to frequently access most non-key attributes which secondary index should you use?

If you rarely query, but you write or update frequently which secondary index should you use?

What are limitations of both secondary indexes?

What is Redshift?

What is a data lake? How would you use redshift as a data lake?

What is Neptune? When would you use it?

What is Elasticache?

What benefits does the use of Elasticache with web session store have? Database caching? Leaderboards? Streaming data dashboards? Which engine should you use for each of these?

What are the differences of both engines?

When should you use a database on EC2?

When should you use RDS?

When should you use DynamoDB?

When should you use Redshift?

When should you use Neptune?

When should you use Elasticache?

If you need lots of large binary objects (BLOBS) what should you use?

If you need automated scalability what should you use?

If you need name/value data structure what should you use?

If you have data that is not well structured or unpredictable what should you use?

If the database platform is not supported by RDS what should you use?

If you need complete control over the database what should you use?