Hello Cloud Gurus, and welcome to this lesson

on increasing scalability using RDS Proxy.

We'll begin with how RDS Proxy works.

We'll take a look at RDS Proxy fault tolerance,

other use cases, and my exam tips as well.

So how does RDS Proxy work?

Well, if you think of a typical application

that is connecting to an RDS database,

the database receives information

from the application through a direct connection.

Well, when we use RDS Proxy,

your application is pointed towards RDS Proxy.

RDS Proxy sits between the user application

and the database, and the way it works is

that it pulls and shares database connections to assist

with application stability and database efficiency,

and then the database receives information

from the application through RDS Proxy.

So when we talk about connection pooling,

this is simply the process of having a pool

of active connections to the backend database,

and by maintaining this active connection,

the application is going to run more efficiently.

Now, in terms of fault tolerance, RDS Proxy is serverless

and its scales automatically.

So it's gonna scale with your workload

and provide connection pooling,

and sharing the database connections,

and because it scales automatically,

it's gonna stop the database from being overloaded

by pooling already established database connections instead

of creating new ones each time,

and it's particularly useful for serverless applications,

which could be growing all the time.

It also preserves application connections during failover.

So if your primary database fails, your application

connection is going to be preserved

during the failover process.

So it's gonna detect failover and route the requests

to your standby database really quickly.

And it's deployable over multiple availability zones

for protection from infrastructure failure,

and with RDS Proxy, you get up to 66% faster failover times

compared with not using it.

Pretty impressive, eh?

Other use cases include applications

with unpredictable workloads, applications that open

and close their database connections infrequently,

that require availability through transient failures,

and applications which have open but idle connections,

and as you can imagine, all of these types of application

are going to benefit from having a pool

of active connections available to use.

So for the exam, just remember that RDS Proxy sits

between your client applications and your RDS database.

It performs connection pooling, so it maintains a pool

of active connections to the backend database,

and that's gonna help you with application scalability

and database efficiency as well.

It's serverless and it scales automatically

to your workload.

Preserves application connections, detects failover,

and route requests to your standby.

It's deployable over multiple AZs

for protection against infrastructure failure,

and with RDS Proxy,

you get up to 66% faster failover times.

So that is it for this lesson.

Any questions, let me know.

Otherwise, I will see you in the next one.

Thank you.