[ENGLISH]

Hello, Cloud Gurus, and welcome to this lesson,

which will introduce MemoryDB for Redis.

And we'll begin with, what is MemoryDB for Redis?

What is it suitable for?

We'll take a look at the differences

between MemoryDB and ElastiCache, and my exam tips as well.

So what is MemoryDB for Redis?

Well, it's a massively scalable in-memory database

that scales from gigabytes to over 100 terabytes.

It's highly available, so multi-AZ

and includes a transaction log

for recovery and durability of your transactions.

It's so scalable that it can be used as a primary database

so the entire data set

of an application can actually be stored

in memory instead of using a traditional database

by itself or a traditional database

plus a cache for the frequently accessed queries.

And with MemoryDB for Redis, you get ultra fast performance

and it supports over 160 million requests per second

and you get microsecond read

and single digit millisecond write latency.

And I know what you might be thinking,

microseconds and milliseconds,

they're both really tiny.

But what's the difference?

Well, the way that I remember it is

that 1 millisecond is 1 thousandth of a second.

So it's really, really small just like a millipede.

Whereas a microsecond is 1 millionth of a second.

So think about something that is so small

that you might need a microscope to see it.

And with MemoryDB for Redis,

you are getting microsecond read latency

and single digit millisecond latency for writes.

So it is really, really fast for both reads and writes.

So with that in mind, what is it actually suitable for?

While it's great for workloads

requiring an ultra fast Redis compatible primary database,

high performance applications

that need an in-memory database to handle millions

of requests per second,

application that are data intensive

or low latency applications

that also require high scalability

and highly scalable microservices based architectures.

And an example use case is an online gaming company

that needs a way for millions of concurrent users

to generate interactive 3D digital worlds

and share digital assets.

Now, you might be thinking, what's the difference

between ElastiCache for Redis and MemoryDB for Redis?

And I don't blame you,

I was confused myself

because they are both in memory data stores

that help applications to access data faster

than reading from disk,

but there are some key differences.

So ElastiCache for Redis is a database cache

so it's an in-memory database cash service

and it sits in front of a traditional database like RDS.

It's fast, but it is not ultra fast.

So you get millisecond read latency.

And a common use case would be a website

that needs to store session data for its customers.

Whereas MemoryDB for Redis

can be used as your primary database,

so you can reduce complexity by removing the need

of a traditional RDS database plus a cache.

You get Ultrafast performance,

so you get microsecond read

and single digit millisecond, write latency;

and a common use case is the online gaming company

with millions of users sharing digital assets.

So for the exam,

memoryDB for Redis is an in-memory database

with ultra fast performance, microsecond read

and single digit millisecond writes.

It's also massively scalable

to over 100 terabytes, use cases include high performance,

large scale microservices applications.

For instance, an online game with millions of users,

sharing digital assets.

And if you have a choice

between MemoryDB and ElastiCache,

just remember that MemoryDB is so scalable

it can store your entire dataset in memory,

no database required.

Whereas ElastiCache is an in-memory cache for databases

and it's gonna sit in front of an RDS database.

So that's it for this lesson.

Any questions, let me know,

otherwise, I'll see you in the next one, thank you.

[SPANISH]

Hola, Cloud Gurus, y bienvenidos a esta lección.

que presentará MemoryDB para Redis .

Y comenzamos con, ¿qué es MemoryDB para Redis ?

¿Para qué es adecuado?

Vamos a echar un vistazo a las diferencias.

entre MemoryDB y ElastiCache , y también mis consejos para el examen.

Entonces, ¿qué es MemoryDB para Redis ?

Bueno, es una base de datos en memoria escalable masivamente

que escala de gigabytes a más de 100 terabytes.

Tiene alta disponibilidad, así que multi-AZ

e incluye un registro de transacciones

para la recuperación y durabilidad de sus transacciones.

Es tan escalable que se puede utilizar como base de datos primaria.

por lo que todo el conjunto de datos

de una aplicación se puede almacenar

en memoria en lugar de usar una base de datos tradicional

por sí mismo o una base de datos tradicional

además de un caché para las consultas de acceso frecuente.

Y con MemoryDB para Redis , obtiene un rendimiento ultrarrápido

y admite más de 160 millones de solicitudes por segundo

y obtienes una lectura de microsegundos

y latencia de escritura de milisegundos de un solo dígito.

Y sé lo que podrías estar pensando,

microsegundos y milisegundos,

son muy pequeños.

Pero ¿cuál es la diferencia?

Bueno, la forma en que lo recuerdo es

que 1 milisegundo es 1 milésima de segundo.

Así que es muy, muy pequeño como un milpiés.

Mientras que un microsegundo es 1 millonésima de segundo.

Así que piensa en algo que es tan pequeño

que podría necesitar un microscopio para verlo.

Y con MemoryDB para Redis ,

obtienes una latencia de lectura de microsegundos

y latencia de milisegundos de un solo dígito para escrituras.

Por lo tanto, es muy, muy rápido tanto para lecturas como para escrituras.

Entonces, con eso en mente, ¿para qué es realmente adecuado?

Si bien es excelente para cargas de trabajo

que requiere un ultrarrápido Base de datos principal compatible con Redis ,

aplicaciones de alto rendimiento

que necesitan una base de datos en memoria para manejar millones

de solicitudes por segundo,

aplicaciones que son intensivas en datos

o aplicaciones de baja latencia

que también requieren alta escalabilidad

y arquitecturas basadas en microservicios altamente escalables .

Y un caso de uso de ejemplo es una empresa de juegos en línea

que necesita una forma para millones de usuarios simultáneos

para generar mundos digitales 3D interactivos

y compartir activos digitales.

Ahora, usted podría estar pensando, ¿cuál es la diferencia

entre ElastiCache para Redis y MemoryDB para Redis ?

Y no te culpo,

yo mismo estaba confundido

porque ambos están en almacenes de datos de memoria

que ayudan a las aplicaciones a acceder a los datos más rápido

que leer del disco,

pero hay algunas diferencias clave.

Entonces ElastiCache para Redis es un caché de base de datos

por lo que es un servicio de caja de base de datos en memoria

y se encuentra frente a una base de datos tradicional como RDS.

Es rápido, pero no ultrarrápido .

Entonces obtienes una latencia de lectura de milisegundos.

Y un caso de uso común sería un sitio web

que necesita almacenar datos de sesión para sus clientes.

Mientras que MemoryDB para Redis

puede utilizar como su base de datos principal,

para que pueda reducir la complejidad eliminando la necesidad

de una base de datos RDS tradicional más un caché.

Obtiene un rendimiento ultrarrápido ,

entonces obtienes una lectura de microsegundos

y milisegundos de un solo dígito, latencia de escritura;

y un caso de uso común es la empresa de juegos en línea

con millones de usuarios compartiendo activos digitales.

Así que para el examen,

memoryDB para Redis es una base de datos en memoria

con rendimiento ultrarrápido , lectura de microsegundos

y escrituras de milisegundos de un solo dígito.

También es enormemente escalable

a más de 100 terabytes, los casos de uso incluyen alto rendimiento,

de microservicios a gran escala .

Por ejemplo, un juego en línea con millones de usuarios,

compartir activos digitales.

Y si tienes una opción

entre MemoryDB y ElastiCache ,

solo recuerde que MemoryDB es tan escalable

puede almacenar todo su conjunto de datos en la memoria,

no requiere base de datos.

Mientras que ElastiCache es un caché en memoria para bases de datos

y se sentará frente a una base de datos RDS.

Eso es todo por esta lección.

Cualquier pregunta, hágamelo saber,

De lo contrario , te veré en el próximo, gracias.