



AWS Simple Storage Service (S3)

—TOMORROW'S HERE—

Topics to be covered

- Introduction of S3
- Object & Bucket
- Features of S3
- Types of S3 Classes
- Versioning
- Use Cases

Introduction to Simple Storage Service (S3)

- S3 is one of the **oldest and most fundamental** AWS services.
- This service allows you to **store and retrieve any amount of data** from **anywhere in the world**.

Anytime

Any
amount of
data

Anywhere

- Company need an ability to:
 - Simply & Securely collect data
 - Store data
 - Analyze data

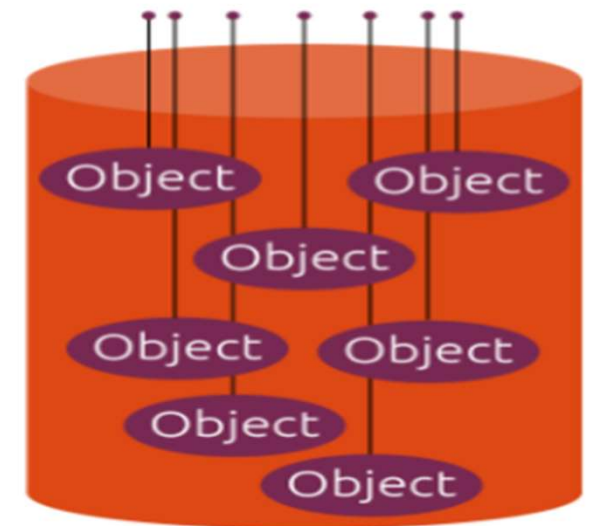
On
Massive
Scale



S3

S3 Overview

- Amazon S3 is **object storage**.
- **Object Storage**
 - Store any file as a **whole** and **does not divide** them.
 - If you want to change a **part of a file**, you must make the change and then **re-upload the entire modified file**.
 - Object May be
 - The File / data itself
 - Metadata
 - Object Global Unique ID (Number), always Unique
- Example:
 - DropBox, AWS S3, Google Drive.



Amazon S3 - Buckets

- Amazon S3 allows people to store objects (files) in “buckets” (directories)
- Buckets must have a globally unique name (across all regions all accounts)
- Buckets are defined at the region level
- S3 looks like a global service but buckets are created in a region
- Naming convention
 - No uppercase, No underscore
 - 3-63 characters long
 - Not an IP
 - Must start with lowercase letter or number
 - Must NOT start with the prefix xn--
 - Must NOT end with the suffix -s3alias



S3 Bucket

Buckets

- It is a **containers** for store Objects.
- All data stored as an object.
- Virtually unlimited storage
 - Single object limited to 5TB.
- Allowed **100 buckets** per account default.
- Provide **11 9s durability**.
- **Granular access** to bucket and objects.
- **Low-latency** access to the data over the internet by **Hypertext Transfer Protocol (HTTP) or Secure HTTP (HTTPS)**



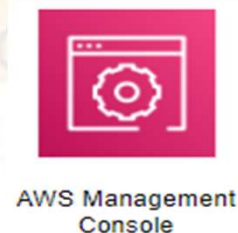
Buckets

- You can not create nested buckets.
- Bucket ownership is non-transferrable.
- Bucket is always Region specific.



Features provided

- By default, **none of your data is shared publicly**.
- You can also **encrypt your data** in transit and choose to enable **server-side encryption** on your **objects**.
 - The encryption of data at **its destination by the application or service** that receives it.
- Access Amazon S3 through
 - Web-based AWS Management Console
 - AWS Command Line Interface (AWS CLI)
 - Programmatically through the **API and SDKs**



AWS Management Console



AWS Command Line Interface



SDK

– Amazon S3 - Objects

- Objects (files) have a Key
- The **key** is the FULL path:
 - s3://my-bucket/my_file.txt
 - s3://my-bucket/my_folder1/another_folder/my_file.txt
- The key is composed of **prefix** + **object name**
 - s3://my-bucket/my_folder1/another_folder/my_file.txt
- There's no concept of “directories” within buckets (although the UI will trick you to think otherwise)



Object



S3 Bucket
with Objects

Amazon S3 – Objects (cont.)



- Object values are the content of the body:
 - Max. Object Size is 5TB (5000GB)
 - If uploading more than 5GB, must use “multi-part upload”
- Metadata (list of text key / value pairs – system or user metadata)
- Tags (Unicode key / value pair – up to 10) – useful for security / lifecycle
- Version ID (if versioning is enabled)

Bucket URLs

- In Amazon S3, **path-style URLs** use the following format:
 - `https://s3.Region.amazonaws.com/bucket-name/key-name`
- Example:
 - create a bucket named **sagebucket** in the **US West (Oregon) Region**, and you want to access the **sage.jpg** object in that bucket, you can use the following path-style URL:
 - `https://s3.us-west-2.amazonaws.com/sagebucket/sage.jpg`
- In a **virtual-hosted-style URI**, the **bucket name** is part of the **domain name in the URL**.
 - `https://bucket-name.s3.Region.amazonaws.com/key-name`
- `https://sagebucket.s3.us-west-2.amazonaws.com/sage.jpg`

S3 Sub-Resources

- **Lifecycle:**
 - To decide on object life cycle management.
- **Website:**
 - To hold configuration related to static website in S3 bucket.
- **Versioning:**
 - Keep object versions as it changes.
 - Prevent accidentally data / object **deletion or overwrite**
 - only **enable** or **suspend**
- **Access Control List:**
 - To provide security – Bucket Policy

Bucket Permissions

- You can grant S3 bucket / object permission to:
 - Individual User
 - AWS account
 - Make the resource public
 - To all authentic Users.

Amazon S3 Object Ownership

- It is **enables** bucket owners to **automatically assume ownership of objects** that are uploaded to their buckets by other AWS Accounts.
- Object Ownership has three settings:
 - **ACLs disabled:**
 - **Bucket owner enforced (recommended)** – ACLs are disabled, and the **bucket owner automatically owns and has full control over** every object in the bucket.
 - **ACLs enabled:**
 - **Bucket owner preferred** – The bucket owner **owns and has full control over** new objects that **other accounts write to the bucket** with the bucket-owner-full-control canned ACL.
 - **Object writer (default)** – The AWS account that uploads an **object owns the object**, has full control over it, and **can grant other users access** to it through ACLs.

Object Lock

- Amazon S3 Object Lock is an Amazon S3 feature that allows you to store objects using a **write once, read many (WORM) model**.
- Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely.
- You can use **WORM protection** for scenarios where it is imperative that data is **not changed or deleted** after it **has been written**.

Amazon S3 Use cases

- Backup and storage
- Disaster Recovery
- Archive
- Hybrid Cloud storage
- Application hosting
- Media hosting
- Data lakes & big data analytics
- Software delivery
- Static website



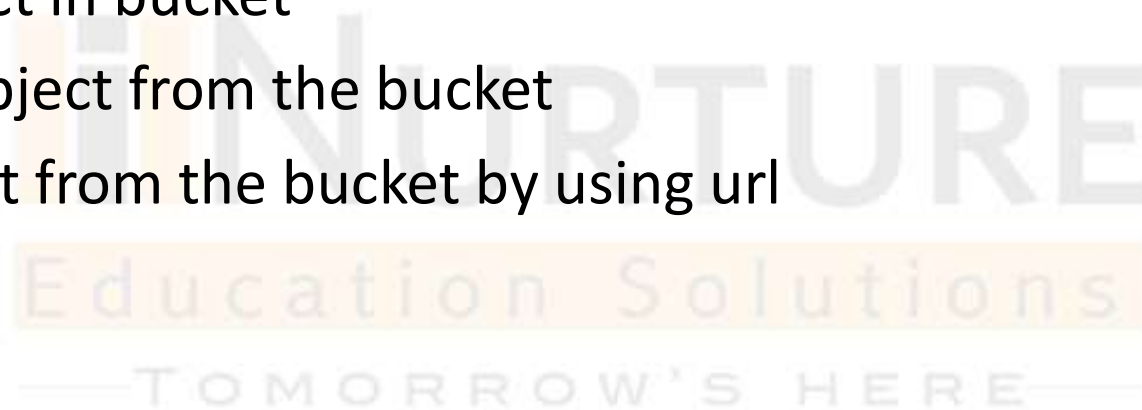
Nasdaq stores 7 years of data into S3 Glacier



Sysco runs analytics on its data and gain business insights

Demo

- Create the bucket
- Upload object in bucket
- Download object from the bucket
- Access object from the bucket by using url



Amazon S3 Storage Classes

- S3 offers a range of storage classes that you can choose from based on the **data access, resiliency, and cost** requirements of your workloads.
- S3 storage classes are purpose-built to provide the **lowest cost storage** for different **access patterns**, and virtually any use case.
 - S3 Standard
 - S3 Intelligent-Tiering
 - S3 Standard-Infrequent Access (S3 Standard-IA)
 - S3 Glacier(Instant Retrieval / Flexible Retrieval / Deep Archive)
 - S3 One Zone-Infrequent Access (S3 One Zone-IA)
 - S3 on Outposts

S3 Standard

- Its **high durability, availability, and performance** object storage for **frequently accessed data**.
- It provide wide variety of use cases, including
 - Cloud applications
 - Dynamic websites
 - Content distribution
 - Mobile and gaming applications
 - Big data analytics.

S3 Standard Key Features

- ❖ Low latency and high throughput performance
- ❖ Designed for durability of 99.9999999999%.
- ❖ Resilient against events that impact an entire Availability Zone
- ❖ Designed for 99.99% availability over a given year
- ❖ Backed with the Amazon S3 Service Level Agreement for availability
- ❖ Supports SSL for data in transit and encryption of data at rest



S3 Standard-Infrequent Access (S3 Standard-IA)

- S3 Standard-IA is for data that is accessed less frequently but requires rapid access when needed.
- It is less expensive than S3 Standard storage, but you will be charged for a retrieval fee.
- S3 Standard-IA ideal for
 - Long-term storage
 - Backups
 - Data store for disaster recovery files

S3 Intelligent-Tiering

- Data with unknown or changing access patterns.
- Designed to optimize costs by automatically moving data to the most cost-effective access tier, without performance impact or operational overhead.
- S3 Intelligent-Tiering automatically stores objects in three access tiers:
 - One tier that is optimized for frequent access
 - A 40% lower-cost tier that is optimized for infrequent access
 - A 68% lower-cost tier optimized for rarely accessed data.

S3 Intelligent-Tiering

- S3 Intelligent-Tiering **monitors** access patterns and moves objects that **have not been accessed for 30 consecutive days** to the **Infrequent Access tier**.
- After **90 days** of no access to the **Archive Instant Access tier**.
- For data that **does not require immediate retrieval**, you can set up **S3 Intelligent-Tiering to monitor** and automatically move objects that **aren't accessed for 180 days or more** to the **Deep Archive Access tier** to realize up to **95% in storage cost savings**.
- There are **no retrieval charges** in S3 Intelligent-Tiering.

S3 Intelligent-Tiering Features

- Frequent, Infrequent, and Archive Instant Access tiers have the same low-latency and high-throughput performance of S3 Standard.
- Objects smaller than 128KB can be stored in S3 Intelligent-Tiering but will always be charged at the Frequent Access tier rates and are not charged the monitoring and automation charge.
- It is the ideal storage class for long-lived data with access patterns that are unknown or unpredictable.

S3 One Zone-Infrequent Access (S3 One Zone-IA)

- Unlike other S3 Storage Classes which store data in a minimum of three Availability Zones (AZs), S3 One Zone-IA stores data in a single AZ and costs 20% less than S3 Standard-IA.
- S3 One Zone-IA is ideal for customers who want a lower-cost option for infrequently accessed data but do not require the availability and resilience of S3 Standard or S3 Standard-IA.
- Designed for 99.5% availability over a given year.
- It's a good choice for storing secondary backup copies of on-premises data or easily re-creatable data if AZ fails.

Amazon S3 Glacier

- Purpose
 - built for data archiving
- Designed to provide
 - Highest performance
 - Most retrieval flexibility
 - Lowest cost archive storage in the cloud.
- Amazon S3 Glacier Instant Retrieval
- Amazon S3 Glacier Flexible Retrieval (Formerly S3 Glacier)
- Amazon S3 Glacier Deep Archive

Amazon S3 Glacier Instant Retrieval

- Lowest-cost storage for long-lived data that is **rarely accessed** and requires retrieval in milliseconds.
- Ideal for archive data that **needs immediate access**, such as
 - Medical Images,
 - News media assets, or user-generated content archives.
- **128 KB** minimum object size.
- **S3 PUT API for direct uploads** to **S3 Glacier Instant Retrieval**, and S3 Lifecycle management for automatic migration of objects

Amazon S3 Glacier Flexible Retrieval

- Its low-cost storage, **up to 10% lower cost** (than S3 Glacier Instant Retrieval), for archive data that is accessed **1—2 times per year** and is retrieved **asynchronously**.
- Ideal for **backup and disaster recovery** use cases when **large sets of data** occasionally need to be **retrieved in minutes**, without concern for **costs**.
- Configurable **retrieval times**, from **minutes to hours**, with free bulk retrievals.

Amazon S3 Glacier Deep Archive

- Lowest-cost storage class and supports long-term retention & digital preservation for data that will be retained for 7–10 years and may be accessed once or twice in a year.
- Ideal alternative to magnetic tape libraries
- Retrieval time within 12 hours and It has a minimum storage duration period of 180 days.
- S3 PUT API for direct uploads to S3 Glacier Deep Archive, and S3 Lifecycle management for automatic migration of objects.
- Retrieval costs can be reduced by using bulk retrieval, which returns data within 48 hours.

AWS Outposts

- AWS Outposts is a family of **fully managed solutions** delivering **AWS infrastructure and services** to virtually any on-premises or edge location for a truly consistent hybrid experience.
- Amazon S3 on **Outposts** delivers object storage to **your on-premises AWS Outposts environment**.
- S3 on Outposts provides **a single Amazon S3 storage class**, named S3 Outposts, which uses the **S3 APIs**, and is designed to **durably and redundantly store data** across multiple devices and servers on your Outposts.

Key Features

- S3 Object compatibility and bucket management through the [S3 SDK](#)
- Designed to [durably and redundantly](#) store data on your Outposts
- Encryption Provided
- Authentication and authorization using [IAM](#), and [S3 Access Points](#)
- Transfer data to AWS Regions using [AWS DataSync](#)

Summary

Particulars	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier Instant Retrieval	S3 Glacier Flexible Retrieval	S3 Glacier Deep Archive
Durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Availability	99.99%	99.9%	99.9%	99.5%	99.9%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99%	99.9%	99.9%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128 KB	128 KB	128 KB	40 KB	40 KB

Summary

Particulars	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier Instant Retrieval	S3 Glacier Flexible Retrieval	S3 Glacier Deep Archive
Minimum storage duration charge	N/A	N/A	30 days	30 days	90 days	90 days	180 days
Retrieval charge	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	milliseconds	milliseconds	milliseconds	milliseconds	minutes or hours	hours
Lifecycle transitions	Yes	Yes	Yes	Yes	Yes	Yes	Yes

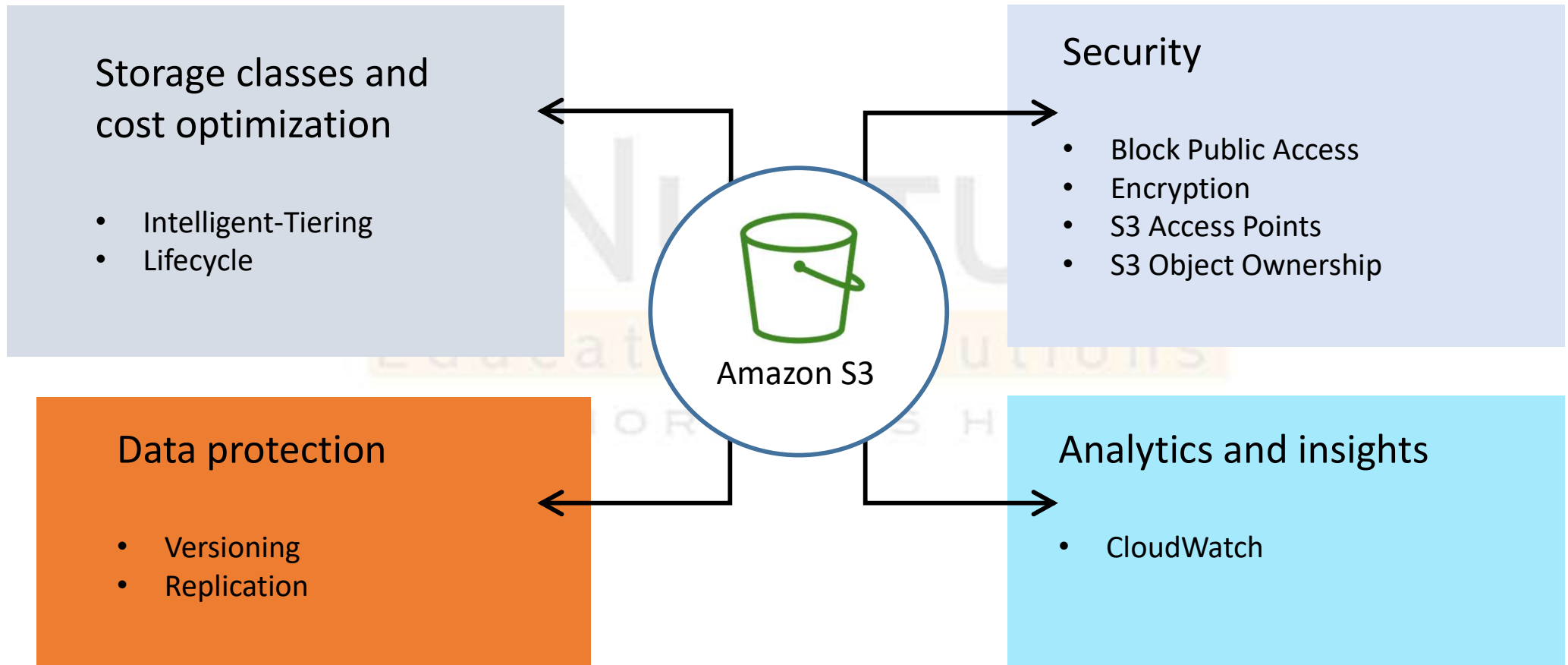
S3 Lifecycle Policies

- With S3 Lifecycle policies, you can delete or move objects based on age.
- Using S3 Lifecycle policies, you can have data cycled at regular intervals between different Amazon S3 storage types.
- This reduces your overall cost because you are paying less for data as it becomes less important with time.
- In addition to being able to set lifecycle rules per object, you can also set lifecycle rules per bucket.
- Lifecycle configuration automatically changes data storage tiers.

S3 Lifecycle Policies

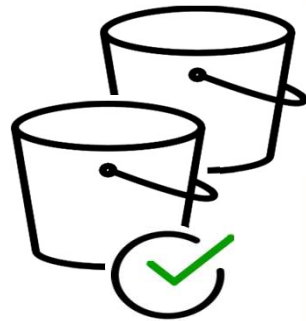
- Lifecycle functionality is located on the **bucket level**.
- A lifecycle policy can be applied to:
 - The **entire bucket**
 - One **specific folder** within a bucket
 - One **specific object** within a bucket
- When an **S3 object is deleted** by the end user/application, Amazon S3 will insert a **delete marker**.
- By default, an **object on S3 doesn't have an expiration date**.
- If you don't specify a **lifecycle rule**, it will be there **forever**.

Best Practices



Amazon S3 Use Cases

- When you need to write once, read many times
- Large number of users and diverse amounts of content
- Growing data sets
- Spiky access patterns (S3 Standard and S3 Intelligent-Tiering)
- Long-term archival storage (S3 Glacier and S3 Glacier Deep Archive)



AWS DynamoDB

- Amazon DynamoDB is a fully managed, serverless, key-value NoSQL database designed to run high-performance applications at any scale. DynamoDB offers built-in security, continuous backups, automated multi-Region replication, in-memory caching, and data import and export tools.
- Amazon DynamoDB is a fully managed NoSQL database service that allows to create database tables that can store and retrieve any amount of data.
- It automatically manages the data traffic of tables over multiple servers and maintains performance.
- It also relieves the customers from the burden of operating and scaling a distributed database. Hence, hardware provisioning, setup, configuration, replication, software patching, cluster scaling, etc. is managed by Amazon.

Amazon RDS

- Amazon Relational Database Service (Amazon RDS) is a collection of managed services that makes it simple to set up, operate, and scale databases in the cloud.
- Amazon RDS is a Relational Database Cloud Service
- Amazon RDS minimizes relational database management by automation
- Amazon RDS creates multiple instances for high availability and failovers
- Amazon RDS supports PostgreSQL, MySQL, Maria DB, Oracle, SQL Server, and Amazon Aurora

Amazon RDS



Amazon RDS

Advantage over using RDS versus deploying DB on EC2

- RDS is a managed service:
 - Automated provisioning, OS patching
 - Continuous backups and restore to specific timestamp (Point in Time Restore)!
 - Monitoring dashboards
 - Read replicas for improved read performance
 - Multi AZ setup for DR (Disaster Recovery)
 - Maintenance windows for upgrades
 - Scaling capability (vertical and horizontal)
 - Storage backed by EBS (gp2 or io1)
- BUT you can't SSH into your instances

Relational Databases

- Amazon RDS stores data as tables, records, and fields
- Values from one table can have a relationship to values in other tables. Relationships are a key part of relational databases.
- Relational databases are often used for storing transactional and analytical data
- Relational databases provide stability and reliability for transactional databases

RDS Pricing - Pay as You Go

- Amazon RDS is pay as you go. It is comprised of 3 parts:
- Hosting. You can choose from different types of hosting depending on your need
- Storage and Operations. Storage is billed per gigabyte per month, and I/O is billed per million-request
- Data transferred