# RDS

# Aurora

* An Amazon Aurora *DB cluster* consists of one or more DB instances and a cluster volume that manages the data for those DB instances. C*luster volume*  spans multiple Availability Zones  with each Availability Zone having a copy of the DB cluster data.
* 6 copies of your data across 3 Availability Zones
* DB instance types:
  + Primary DB instance – Supports read and write operations
  + Aurora Replica
* Snapshots:
  + Auto : can’t be disabled and retention period is 1.
  + mannual

# DynamoDB

* key-value and document database
* Only requires a table name and primary key [partition key (Hash attribute) + sort key].
* Secondary indexes allow you to perform queries on attributes that are not part of the table's primary key. DynamoDB maintains indexes automatically.
* DynamoDB Streams + Global tables: Used to capture data modification events (events + lambda🡪SES). Life time is 24 hr only.
* Encryption at rest using KMS: Will encrypt data in streams
* Recovery:
  + Point-in-time
  + on-demand backups
* TTL: Helps to reduce cost
* Item size:400 kb.
* Read/Write Capacity Modes:
  + On-demand: Used when un-predictable traffic/workloads. Will not support auto scaling.
  + Provisioned.
* Consistency
  + - Strongly:
      * Read capacity: ( (Item size/4) \* Item read/sec)
      * Write capacity: (Item size \* Item write/sec)
    - Eventual
    - Transactional
* Put Item /BatchWriteItem (Put/delete)
* Get Item.batch Get item
* For composite PK, use query.

# Redshift

* Cluster (1 Leader node + multiple compute node ) can have multiple DB (Base DB + other DB)
* Master user have access to all DB.
* parameter groups: date presentation style and floating-point precision
* Snapshots
  + Automated (1 day) and manual.
  + Stored in s3
  + For restore snapshot, redshift will create new cluster and import data from s3.
* Supports both the EC2-VPC and EC2-Classic platforms to launch a cluster.
* 1 VPC+ 1 subnet + 1 AZ+ multi security GP +(VPC endpoints )+ (Elastic IP )
* Security:
  + Infra-structure level: IAM policies
  + N/W layer: SG
  + Database level: Data encryption(KMS) and use SSL encrypted connection
* Monitoring :
  + Audit logs: authentication attempts, connections, disconnections, changes to database user definitions, and queries run in the database.
  + Cloud watch: cpu utilization + health check etc.
  + Events and Notification.
* Connecting using JDBC/ODBC URL
* Redshift Spectrum:Using Amazon Redshift Spectrum, you can efficiently query and retrieve structured and semistructured data from files in Amazon S3 without having to load the data into Amazon Redshift tables.

# Route 53

* Functions
  + Domain Name Registration
  + DNS routing
  + Health Checking
* Hosted Zone
* Records
* Policies (TTL)
  + Simple routing policy  (1 Record + Multiple IP)
  + Multivalue answer routing policy (Simple RP + HealthCheck)
  + Weighted routing policy (Health Check + 20%)
  + Latency routing policy (HC)
  + Failover routing policy (Active-passive+HC)
  + Geolocation routing policy (HC)
  + Geoproximity routing policy (Traffic Policies)

# ELB

* Types:
  + Application LB :Intelligent Routing
  + N/W LB : High performance
  + Classic ( Layer 7 + 4) : Cost down
* 504 Error & X-forwarded-for
* Cross-Zone LB & Connection-draining
* Response Timeout + Interval + Unhealthy threshold + Healthy threshold
* Subnets per Availability Zone per load balancer: 1
* Rules
  + Forward
  + Redirect
  + Fixed response
  + Authentication
* <https://aws.amazon.com/elasticloadbalancing/features/#compare>
* Custom security policies can only define in classic LB
* ALB supports Lambda functions as targets
* With SNI support you can associate multiple certificates with a listener and each secure application behind a load balancer can use its own certificate.