# AWSome day online session

# Load Balancer

* Types:
  + Application Load Balancer
    - Flexible application management
    - Advanced load balancing of TTP
    - EG: Can Fork request to different EC2 instance, different containers by Resources
* Auto Scaling:
  + Operates with few constraints MODE: **max, min, and desired** based on RESOURCE
  + Based on Resource requirements. EG: high CPU, low usage
  + Select a load metric for your application
  + Conditional and/or
  + DESIRED mode, between max and min.
  + Based on past history of USAGES, we can set the autoscaling accordingly. EG: high demand on Fridays
* Amazon cloudwatch
  + Monitors aws resources
  + It allows the Auto Scaling to choose different MODE, that can be high, low or desired
  + Cloudwatch collects and track metrics: standard and custom metrics.
  + Cloudwatch alarms; sends notifications
* Coudwatch works as:
  + Free five minute measurements for EC2 metrics
  + BENFITIS:
    - Visibility across your applications, infrastructure, and services
    - Reduce mean time to resolution and improve total cost of ownership
    - Drive insights for the usage of application
* Route 53
  + Register domain names, EG: we can assign domain name to a specific Load Balancer and works to detect any health checks,
  + Sends traffic to LEAST busy Load Balancer
  + Route 53 routing policies:
    - Simple
    - Weighted
    - Latency
    - Failover
    - Geolocation
    - Geoproximity

# AWS database services and automation

* + Doing-it yourself v. Aws database services

1. Databases on Amazon Ec2
2. Operating system access
3. Need feature of specific application
   * AWS Database service
     + Automated
   * Without the management overhead
   * Amazon Neptune:
     + Graph database
     + Full-managed, suggested based automated
   * Highly manageable
   * Amazon Aurora
     + Cheaper than rest
     + High performance and durable
     + 5X faster than MySQL databases
     + Upto 3X faster than standard PostgreSQL databases

# Amazon RDS Benefits

* easy to administrator

# AWS LAMBDA

* automation, where **source** s3 bucket loads files to **target** s3 bucket by lambda function
* usage: to trigger functions,

# Automated Deployment

* Elastic Beanstalk:
  + Upload your application doe
  + EG: for stores that don’t have IT departments
  + The service handles:
    - Resource providsioning
    - Wide selection of application platforms
    - Variety of application

# AWS cloud formation

* Caches the content
* Stacks up codes: YAML or JSON and fires up into WEB server
* Like ANSIBLE, TERRAFORM
* Step wise instructions
* We can use the TEMPLATE in different region to perform similar actions

#