AWS Solution Architect Associate (SAA-C03)
Course Curriculum

Module-1: Introduction to Cloud Computing

Objective
To help learners understand the fundamentals of cloud computing, its benefits, key characteristics, deployment and service models, and the importance of cloud adoption in the modern it landscape, establishing a strong foundation for AWS learning.

Topics

- What is Cloud Computing
- Key Characteristics of Cloud Computing
- Benefits of Cloud Computing
- Cloud Deployment Models Public Private Hybrid
- Cloud Service Models IaaS PaaS SaaS
- Cloud Computing vs Traditional Computing
- Why Cloud Computing is Important ToModule
- Real-World Use Cases of Cloud Computing
- Introduction to AWS as a Leading Cloud Provider

Hands-On Lab

- Identify real-life examples of cloud computing in daily services and enterprise environments
- Classify sample workloads into appropriate deployment models public private or hybrid
- Classify services like EC2 S3 Gmail and Salesforce under IaaS PaaS SaaS categories
- Navigate the AWS global infrastructure map and identify regions and availability zones relevant for learners

Module-2: Getting started with AWS

Objective
To help learners understand AWS as a cloud platform, its global infrastructure, set up and navigate their AWS accounts, explore the AWS Free Tier, and gain hands-on familiarity with the AWS Management Console for practical use.

Topics

- Introduction to AWS and its core services.
- Understanding AWS global infrastructure: regions, availability zones, and edge locations.
- Overview of AWS Free Tier and its limitations.
- Creating and verifying an AWS Free Tier account.
- Navigating the AWS Management Console effectively.
- Understanding the AWS shared responsibility model.
- Introduction to AWS service categories: compute, storage, networking, and database.
- AWS billing console, cost management basics, and budget alert setup.
- Best practices for managing AWS accounts safely and cost-effectively.

Hands-On Lab

- Create and verify an AWS Free Tier account with billing alerts to monitor usage.
- Navigate the AWS Management Console and locate core services like EC2, S3, and RDS.
- Select and explore AWS regions and availability zones for deploying services.
- View the AWS billing dashboard and set up a Free Tier budget alert.
- Launch a basic EC2 instance and stop it to understand cost implications.
- Explore the AWS Global Infrastructure Map and identify regions relevant for your use.

Module-3: IAM & AWS CLI

Objective
To help learners understand the fundamentals of Identity and Access Management IAM for secure access control within AWS and to enable them to use the AWS Command Line Interface AWS CLI for managing AWS resources efficiently through practical hands-on activities.

Topics
- Introduction to Identity and Access Management IAM
- Key concepts users, groups, roles, and policies
- Understanding IAM permissions and least privilege principle
- Enabling Multi Factor Authentication MFA for AWS accounts
- IAM best practices for account security
- Introduction to AWS CLI and its benefits
- Installing and configuring AWS CLI on local machines
- Basic AWS CLI commands for managing AWS resources
- Using AWS CLI for managing EC2 S3 and other services
- Best practices for using AWS CLI securely

Hands-On Lab
- Create IAM users and groups with specific permissions
- Enable Multi Factor Authentication for IAM users
- Attach policies to users and groups for controlled access
- Install and configure AWS CLI using access keys and default region
- Use AWS CLI to list and manage S3 buckets
- Use AWS CLI to describe and manage EC2 instances
- Practice retrieving AWS resource information using CLI commands

Module-4: Amazon EC2 & EBS

Objective
To help learners understand Amazon Elastic Compute Cloud EC2 and Elastic Block Store EBS, including how to launch, configure, manage, and secure virtual servers and storage volumes within AWS, enabling them to confidently build scalable and cost-effective compute environments with hands-on practice.

Topics
- Introduction to Amazon EC2 and its use cases
- EC2 instance types families and pricing models
- Key components AMI instance types key pairs and security groups
- Launching and configuring EC2 instances
- Connecting to EC2 instances using SSH and EC2 Instance Connect
- Understanding Elastic Block Store EBS
- Types of EBS volumes and their use cases
- Attaching detaching and resizing EBS volumes
- Creating and managing EBS snapshots for backup
- EC2 instance lifecycle and stopping starting and terminating instances
- Best practices for managing EC2 and EBS resources efficiently

Hands-On Lab
- Launch an Amazon EC2 instance using the AWS Management Console
- Connect to the EC2 instance using SSH or EC2 Instance Connect
- Configure security groups to allow specific inbound traffic

- Create and attach an EBS volume to the EC2 instance
- Format and mount the EBS volume on the instance
- Create a snapshot of the EBS volume for backup
- Stop start and terminate EC2 instances to understand lifecycle management

Module-5: EC2 & More

Objective
To help learners deepen their understanding of Amazon EC2 by exploring advanced networking, IP management, storage options, and related services, enabling them to architect and manage compute resources effectively for various workload requirements.

Topics
- Understanding private public and elastic IP addresses in AWS
- Differences and use cases for private IPs public IPs and elastic Ips
- Introduction to EC2 placement groups for workload distribution and high performance
- Types of placement groups cluster spread and partition
- Elastic Network Interfaces ENIs and their use cases for network management
- Attaching detaching and managing ENIs with EC2 instances
- Understanding EC2 instance store ephemeral storage and its use cases
- Differences between instance store and EBS volumes
- Introduction to Amazon Elastic File System EFS and its features
- Using EFS for shared storage across multiple EC2 instances
- EFS performance modes and lifecycle management
- Best practices for managing networking and storage for EC2 instances

Hands-On Lab
- Launch EC2 instances with private and public IPs and observe access differences
- Allocate and associate an elastic IP address with an EC2 instance
- Create and attach an elastic network interface to an existing EC2 instance
- Launch instances using cluster placement groups and observe placement behaviour
- Launch an EC2 instance with instance store volumes and test ephemeral storage
- Create and mount an Amazon EFS file system on multiple EC2 instances
- Test file sharing across EC2 instances using EFS

Module-6: Amazon S3 & Storage Classes

Objective
To help learners understand Amazon Simple Storage Service S3 and its storage classes, including how to store, manage, and secure data efficiently on AWS, while gaining hands-on experience in creating buckets, managing objects, configuring permissions, and implementing lifecycle policies for cost-effective storage management.

Topics
- Introduction to Amazon S3 and its use cases
- Key concepts buckets objects and keys in S3
- Durability availability and scalability features of S3
- S3 storage classes and their use cases
- Standard Intelligent Tiering Standard-IA One Zone-IA
- Glacier and Glacier Deep Archive
- Uploading downloading and managing objects in S3
- S3 versioning and its benefits

- Bucket policies and access control lists ACLs for permissions management
- Enabling and configuring S3 encryption for data at rest
- S3 lifecycle policies for automated transitions and deletions
- Hosting static websites using Amazon S3
- Monitoring and logging S3 access using AWS CloudTrail and server access logging
- Best practices for managing data securely and cost-effectively in S3

Hands-On Lab

- Create an S3 bucket and upload objects to it
- Configure bucket permissions and test public access to objects
- Enable versioning on the S3 bucket and observe object version management
- Create and apply lifecycle policies to transition objects between storage classes
- Enable server-side encryption for objects stored in S3
- Configure the bucket for static website hosting and test accessing the site
- Enable server access logging on the S3 bucket for monitoring purposes

Module-7: Auto Scaling and Load Balancers

Objective

To help learners understand how to design and implement scalable and highly available applications using AWS Auto Scaling and Load Balancers, enabling them to efficiently manage variable workloads, improve fault tolerance, and optimize costs within AWS environments.

Topics

- Introduction to scalability and high availability concepts in AWS
- What is AWS Auto Scaling and its benefits
- Understanding Auto Scaling Groups ASG and their components
- Launch configurations and launch templates
- Scaling policies target tracking step scaling and scheduled scaling
- Health checks and automatic instance replacement
- Introduction to Elastic Load Balancing ELB
- Types of load balancers Application Load Balancer Network Load Balancer and Gateway Load Balancer
- Configuring and using Application Load Balancer with EC2 instances
- Listener rules target groups and health checks in load balancers
- Best practices for designing scalable and fault tolerant architectures with ASG and ELB

Hands-On Lab

- Create and configure an Auto Scaling Group with launch configurations
- Define scaling policies to automatically add or remove instances based on CPU utilization
- Create and configure an Application Load Balancer and attach it to EC2 instances
- Test load balancing by accessing the public DNS of the load balancer
- Simulate high CPU load to trigger Auto Scaling actions
- Observe scaling activities in the AWS Management Console
- Terminate instances manually to observe automatic replacement by the Auto Scaling Group

Module-8: AWS Networking & Content Delivery

Objective
To help learners understand and design secure, scalable, and highly available network architectures using AWS networking and content delivery services, including VPC, CloudFront, API Gateway, and Route 53, enabling them to optimize performance and connectivity for applications on AWS.

Topics

- Introduction to AWS Virtual Private Cloud VPC and its benefits
- Understanding CIDR blocks and IP addressing in VPC
- Creating and managing subnets public and private subnets
- Route tables and routing in VPC
- Internet Gateway and NAT Gateway differences and use cases
- Configuring and using Internet Gateway for internet access
- Using NAT Gateway and NAT Instances for outbound internet access from private subnets
- Security Groups and Network Access Control Lists NACLs and their differences
- VPC Peering concepts and limitations
- Understanding and using VPC endpoints
- Elastic IPs within VPC and their use cases
- Introduction to AWS Transit Gateway and its use cases for complex architectures
- Monitoring VPC using VPC Flow Logs
- Best practices for designing secure and scalable VPC architectures
- Introduction to Amazon CloudFront and its use cases for content delivery
- Configuring CloudFront distributions with S3 and EC2 as origins
- Understanding caching, invalidation, and edge locations in CloudFront
- Introduction to Amazon API Gateway and its use cases
- Creating and managing REST APIs with API Gateway
- Securing APIs using throttling, authentication, and authorization
- Introduction to Amazon Route 53 as a scalable DNS service
- Configuring domain registration, DNS records, and routing policies in Route 53
- Health checks and failover routing in Route 53

Hands-On Lab

- Create a custom VPC with a defined CIDR block
- Create public and private subnets within the VPC
- Attach an Internet Gateway to the VPC and configure routing
- Launch EC2 instances in public and private subnets and test connectivity
- Create and configure a NAT Gateway for internet access from private subnets
- Configure security groups and NACLs to control traffic flow
- Set up VPC peering between two VPCs and test connectivity
- Create a VPC endpoint for S3 and test private connectivity
- Enable and view VPC Flow Logs to monitor traffic
- Create a CloudFront distribution with an S3 bucket as the origin and test content delivery
- Create and deploy a simple REST API using Amazon API Gateway with Lambda integration
- Configure a custom domain in Route 53 and set up routing to an S3 static website or EC2 instance

Module-9: AWS Lambda

Objective
To help learners deeply understand AWS Lambda for building serverless architectures, including function creation, event-driven processing, permissions, monitoring, and integration with other

AWS services, preparing them to design scalable, cost-effective, and maintainable serverless solutions on AWS.

Topics

- Introduction to AWS Lambda and serverless computing
- Core concepts of Lambda functions including handler, runtime, and memory allocation
- Event-driven architecture and supported event sources
- Understanding the Lambda execution environment and lifecycle
- Lambda permissions and execution roles
- Creating, deploying, and updating Lambda functions
- Lambda triggers including S3, API Gateway, DynamoDB Streams, and EventBridge
- Environment variables and Lambda layers for code and dependency management
- Using AWS SDK within Lambda functions
- Error handling, retries, and DLQs in Lambda workflows
- Monitoring and logging Lambda functions using CloudWatch
- Concurrency and scaling behaviour of Lambda functions
- Pricing model and cost optimization strategies for Lambda
- Best practices for building and managing Lambda functions

Hands-On Lab

- Create a Lambda function using the AWS Management Console
- Invoke the Lambda function manually and test with sample payloads
- Configure an S3 bucket trigger to invoke the Lambda function on object upload
- Create an API using API Gateway to invoke the Lambda function over HTTP
- Set environment variables and access them inside the Lambda function
- View Lambda logs in CloudWatch and analyse invocation details
- Test Lambda error handling and retry mechanisms
- Explore Lambda concurrency settings and test scaling by invoking in parallel

Module-10: Database

Objective

To help learners understand AWS database services, including relational and non-relational options, enabling them to design, deploy, and manage secure, scalable, and highly available databases on AWS for various workload requirements.

Topics

- Introduction to AWS database services and their categories
- Amazon RDS overview, supported engines, and use cases
- RDS Multi-AZ deployments and read replicas for high availability and scalability
- Automated backups, snapshots, and point-in-time recovery in RDS
- Monitoring and maintenance of RDS instances
- Amazon Aurora overview, benefits, and use cases
- Aurora Serverless and scaling options
- Introduction to Amazon DynamoDB as a managed NoSQL database
- Key concepts in DynamoDB including tables, items, and attributes
- Capacity modes provisioned and on-demand in DynamoDB
- Global tables, streams, and DynamoDB triggers with Lambda
- Introduction to Amazon ElastiCache for Redis and Memcached
- Caching strategies with ElastiCache for improving application performance
- Introduction to Amazon Redshift for data warehousing workloads
- Best practices for selecting and managing database services on AWS

Hands-On Lab

- Launch an Amazon RDS instance with automated backups enabled
- Connect to the RDS instance using a SQL client and perform basic operations
- Create a read replica for the RDS instance and observe replication
- Create and configure a DynamoDB table with specified read and write capacity
- Insert and retrieve items from the DynamoDB table using the console
- Enable DynamoDB Streams and configure a Lambda trigger
- Launch an ElastiCache Redis cluster and connect to it for caching data
- Explore the AWS Console for Redshift and review cluster creation steps

Module-11: Containers

Objective

To help learners understand containerization on AWS, including deploying and managing containers using Amazon ECS, Amazon EKS, and AWS Fargate, enabling them to design scalable, portable, and efficient container-based architectures on AWS.

Topics

- Introduction to containerization and its benefits
- Overview of Docker containers and container images
- Introduction to Amazon Elastic Container Service ECS and its components
- Task definitions, services, and clusters in ECS
- ECS launch types including EC2 and AWS Fargate
- Introduction to Amazon Elastic Kubernetes Service EKS and its use cases
- Comparison between ECS and EKS for container orchestration
- ECR (Elastic Container Registry) for storing and managing container images
- IAM roles and permissions for ECS and EKS
- Networking for containers using VPC, security groups, and load balancers
- Monitoring containers with CloudWatch and ECS/EKS integrations
- Best practices for designing container-based workloads on AWS

Hands-On Lab

- Push a Docker image to Amazon ECR
- Create an ECS cluster and run a container using the Fargate launch type
- Deploy a sample containerized application on ECS with a load balancer
- Create an EKS cluster and deploy a simple Kubernetes application
- Test container scaling in ECS by adjusting desired task counts
- Monitor running containers and logs using CloudWatch
- Explore IAM permissions required for ECS tasks and ECR access

Module-12: Management & Governance

Objective

To help learners understand AWS Management and Governance services, enabling them to monitor, automate, audit, and manage AWS resources efficiently while maintaining operational best practices for scalable and secure environments.

Topics

- Introduction to AWS Management and Governance services
- Overview of AWS Organizations for multi-account management
- Creating and managing organizational units and service control policies SCPs
- AWS CloudWatch for monitoring, metrics, and alarms

- Using CloudWatch Logs and CloudWatch Events
- Introduction to AWS CloudTrail for API activity tracking and auditing
- Viewing and analyzing CloudTrail logs for governance
- Overview of AWS Config for resource configuration tracking and compliance
- Creating AWS Config rules for compliance monitoring
- Introduction to AWS Systems Manager and its components
- Using Systems Manager Session Manager for secure instance access
- Patch management and automation with Systems Manager
- Introduction to AWS CloudFormation for infrastructure as code
- Creating and managing CloudFormation stacks and templates
- Introduction to AWS Service Quotas and monitoring service limits
- Best practices for managing AWS resources efficiently using governance services

Hands-On Lab

- Create an AWS Organization and add an AWS account to the organization
- Create an organizational unit and apply a service control policy SCP
- Create CloudWatch alarms to monitor CPU utilization on an EC2 instance
- Send CloudWatch alarm notifications using SNS
- View and analyse CloudWatch Logs for an EC2 instance
- Enable CloudTrail in your AWS account and analyse logged events
- Enable AWS Config and review the configuration history of resources
- Create and evaluate AWS Config rules for compliance
- Use AWS Systems Manager Session Manager to connect to an EC2 instance without SSH
- Create a maintenance window and run patching using Systems Manager
- Create a basic CloudFormation stack to launch an EC2 instance with security groups
- Delete the CloudFormation stack to clean up resources and observe automated rollback
- View and check AWS Service Quotas for EC2 and configure quota alerts

Module-13: AWS CloudFormation

Objective

To help learners understand AWS CloudFormation for infrastructure as code, enabling them to design, deploy, and manage AWS resources automatically and consistently using templates for scalable and repeatable infrastructure management.

Topics

- Introduction to Infrastructure as Code and its benefits
- Overview of AWS CloudFormation and its use cases
- Core concepts: stacks, templates, resources, parameters, and outputs
- Understanding CloudFormation template structure in JSON and YAML
- Intrinsic functions and pseudo parameters in CloudFormation
- Stack creation, updating, and deletion workflows
- Nested stacks for modular template design
- Handling stack failures and rollback protection
- Integration with other AWS services for automated deployments
- Best practices for writing and managing CloudFormation templates
- Cost management considerations when using CloudFormation

Hands-On Lab

- Launch a CloudFormation stack using a sample template to create an EC2 instance and security group
- Explore the stack resources, outputs, and events in the AWS Management Console
- Update the CloudFormation stack to add an S3 bucket and observe changes
- Delete the CloudFormation stack and observe automated resource cleanup

- Create a nested stack to modularize infrastructure deployment
- Use CloudFormation to deploy a VPC with subnets and an Internet Gateway
- Explore using YAML/JSON syntax to create readable CloudFormation templates

Module-14: Application Integration

Objective
To help learners understand AWS Application Integration services, including Step Functions, SNS, SQS, and EventBridge, enabling them to build decoupled, event-driven, and scalable architectures on AWS for various application workflows.

Topics
- Introduction to application integration and event-driven architecture on AWS
- Overview of AWS Step Functions for workflow orchestration
- Creating state machines and managing workflows using Step Functions
- Integration of Step Functions with Lambda and other AWS services
- Monitoring workflows and handling errors in Step Functions
- Introduction to Amazon Simple Notification Service SNS for pub/sub messaging
- Topics, subscriptions, and message publishing in SNS
- SNS delivery protocols and use cases for application notifications
- Introduction to Amazon Simple Queue Service SQS for decoupling applications
- Standard queues vs FIFO queues in SQS and their use cases
- Sending, receiving, and deleting messages in SQS
- Dead-letter queues and visibility timeouts in SQS
- Introduction to Amazon EventBridge for event-driven application design
- Concept of event buses, rules, and targets in EventBridge
- Using EventBridge with AWS services and custom event sources
- Best practices for building scalable and loosely coupled applications using integration services

Hands-On Lab
- Create an SNS topic and configure email subscriptions to receive notifications
- Publish messages to the SNS topic and verify email delivery
- Create a standard SQS queue and send and receive messages using the console
- Configure a dead-letter queue for the SQS queue and test failure scenarios
- Create a Lambda function to process messages from the SQS queue
- Create a Step Functions state machine to orchestrate a workflow with Lambda integration
- Execute the Step Functions workflow and monitor execution and error handling
- Create an EventBridge rule to trigger a Lambda function based on an event pattern
- Test sending custom events to EventBridge and verify Lambda invocation

Module-15: Developer Tools

Objective
To help learners understand AWS Developer Tools, including CodeBuild, CodeDeploy, CodePipeline, and CodeArtifact, enabling them to implement CI/CD pipelines, automate deployments, and manage artifacts efficiently on AWS for modern application development workflows.

Topics
- Introduction to AWS Developer Tools and their role in CI/CD
- Overview of AWS CodeCommit (brief) as a managed Git repository

- Introduction to AWS CodeBuild for building and testing code
- Buildspec files and environment configurations in CodeBuild
- Viewing build history and logs in CodeBuild
- Introduction to AWS CodeDeploy for automated application deployments
- Deployment strategies: in-place, blue/green, and canary deployments
- Using CodeDeploy with EC2 instances and Lambda functions
- Monitoring deployments and rollback mechanisms in CodeDeploy
- Introduction to AWS CodePipeline for automating build, test, and deploy workflows
- Creating pipelines and integrating CodeCommit, CodeBuild, and CodeDeploy
- Pipeline stages, actions, and approvals in CodePipeline
- Introduction to AWS CodeArtifact for artifact storage and management
- Creating repositories and managing packages in CodeArtifact
- Integrating CodeArtifact with build tools like Maven, npm, and pip
- Best practices for implementing CI/CD on AWS

Hands-On Lab

- Create a CodeBuild project and build a sample application using a buildspec file
- Explore build logs and artifacts generated by CodeBuild
- Deploy a sample application to EC2 instances using AWS CodeDeploy
- Test deployment rollback and monitor deployment status in CodeDeploy
- Create an end-to-end CI/CD pipeline using AWS CodePipeline
- Integrate CodeCommit, CodeBuild, and CodeDeploy within the pipeline
- Create a CodeArtifact repository and upload a sample package
- Configure npm or Maven to pull packages from CodeArtifact and test retrieval

You will have a solid foundation across AWS services, architecture best practices, and hands-on experience to design, deploy, and manage scalable, highly available, and cost-effective applications on AWS.

Continue practicing in your AWS Free Tier environment to solidify your learning. Explore advanced architecture patterns, review the AWS Well-Architected Framework, and work on real-world projects to prepare for your certification exam confidently.

Remember, cloud learning is a continuous journey. Stay updated with AWS services, new features, and architecture best practices to strengthen your expertise and deliver value to your projects and clients.

We wish you all the best in your AWS journey and your path toward becoming a skilled, confident AWS Solution Architect.