AWS Cloud Practitioner Essentials: Project Ideas

Project Ideas to Practice AWS Cloud Practitioner Essentials

Practicing with a hands-on project is a great way to reinforce your knowledge from the AWS Cloud Practitioner Essentials course. Here are some project ideas that cover various AWS services and concepts:

1. **Deploy a Static Website on Amazon S3**

   Skills Practiced:

   - Amazon S3
   - IAM (Identity and Access Management)
   - CloudFront (CDN)
   - Route 53 (DNS)

   Project Outline:

   (a) Create an S3 bucket and configure it to host a static website. Upload your HTML, CSS, and JavaScript files.

   (b) Set the correct bucket policies to allow public read access to the files.

   (c) Optionally, set up a custom domain using Amazon Route 53 and connect it to your S3 bucket.

   (d) Enable CloudFront as a content delivery network (CDN) to speed up the website's performance and provide HTTPS support.

2. **Set Up a Simple Web Application with EC2 and RDS**

   Skills Practiced:

   - EC2 (Elastic Compute Cloud)
   - RDS (Relational Database Service)
   - VPC (Virtual Private Cloud)
   - IAM
   - Security Groups

   Project Outline:

(a) Launch an EC2 instance running a web server (e.g., Apache or Nginx) and deploy a basic web application (like a simple PHP or Python web app).

(b) Set up a MySQL or PostgreSQL database using Amazon RDS and connect your web application to the database.

(c) Configure security groups and IAM roles to secure access between the EC2 instance and the RDS database.

(d) Optionally, use VPC to set up a private subnet for your database to enhance security.

3. **Build a Serverless Application with AWS Lambda and API Gateway**

   Skills Practiced:

   - AWS Lambda
   - API Gateway
   - DynamoDB
   - IAM

   Project Outline:

   (a) Create a simple API using AWS API Gateway that invokes a Lambda function when an endpoint is accessed.

   (b) Write a Lambda function in Python, Node.js, or another supported language that processes a request and returns a response.

   (c) Use Amazon DynamoDB to store data that the Lambda function retrieves or modifies.

   (d) Set up IAM roles and policies to grant the necessary permissions for Lambda to interact with DynamoDB and API Gateway.

4. **Deploy a Scalable Web Application using Elastic Beanstalk**

   Skills Practiced:

   - Elastic Beanstalk
   - EC2
   - RDS
   - Auto Scaling
   - Load Balancing

   Project Outline:

   (a) Use Elastic Beanstalk to deploy a web application that automatically handles scaling and load balancing.

(b) Integrate an RDS database with your Elastic Beanstalk environment.

(c) Configure Auto Scaling to handle changes in traffic and a load balancer to distribute traffic evenly across EC2 instances.

(d) Monitor the application using Elastic Beanstalk's built-in monitoring tools.

5. Implement a Data Processing Workflow with AWS S3, Lambda, and SQS

Skills Practiced:

- S3
- Lambda
- SQS (Simple Queue Service)
- IAM

Project Outline:

(a) Set up an S3 bucket to store incoming data files (e.g., CSV, JSON).

(b) Create an SQS queue to handle file processing requests.

(c) Write a Lambda function that is triggered when new files are uploaded to the S3 bucket, which then sends a message to the SQS queue.

(d) Create another Lambda function that processes messages from the SQS queue, performs data transformations or analysis, and stores the results back to S3.

6. Automate Infrastructure Provisioning with AWS CloudFormation

Skills Practiced:

- CloudFormation
- IAM
- EC2
- S3
- RDS

Project Outline:

(a) Write a CloudFormation template to automate the provisioning of a complete stack (e.g., a web server, database, and S3 bucket).

(b) Deploy the stack using the CloudFormation console or CLI.

(c) Modify the template to add more resources or configurations, such as Auto Scaling groups or IAM roles.

(d) Update the stack with the new template and manage resources using CloudFormation.

Tips for All Projects

- **Document Your Work:** Keep notes on the steps you take and any challenges you encounter. This will help reinforce your learning and serve as a valuable reference in the future.
- **Use IAM Roles and Policies:** Practice creating and applying IAM roles and policies to control access to your resources securely.
- **Monitor Costs:** Always keep an eye on your AWS account's billing dashboard to understand the costs associated with your projects.
- **Security Best Practices:** Implement AWS security best practices, such as using multi-factor authentication (MFA), least privilege, and VPCs for network security.

These projects will give you practical experience with various AWS services and help reinforce the concepts you've learned in the AWS Cloud Practitioner Essentials course.