Manage Internet Cloud Resources Using AWS

(Note: If you are not using Free Tier resources in AWS, charges will be incurred for the following tasks. Delete all objects immediately upon completing the project and its assessment.)

Now that you have explored possible database and source control options, you will be migrating to using cloud resources with your development team. You will need virtual machines, object storage and a database. Create them using AWS Management Console, AWS CLI or CloudShell and test connectivity from your local computer. You will also need to ensure that you have good version control procedures in place.

1. Login to the AWS Management Console and create an EC2 instance (Free Tier). Login to the virtual machine (VM) using SSH credentials. Configure the computer to run a web server and test your connectivity to it in a web browser.

2. Using the Management Console, create an S3 bucket and configure it to be a static web server. Test your connectivity to it using a web browser. Duplicate this task using AWS CLI from the CloudShell or your local computer. Create a "data" folder in the bucket and upload CSV or TXT files to it that contain a few customer records.

3. Create a Managed SQL Server database (Free Tier). Upload the records from your S3 bucket data file into a new customer table. Use Management Studio or another tool to test connectivity to the database from your computer.

4. **Optional:** Create a CloudFront distribution that performs CDN (content delivery network) services for one of the websites you have just created. It should improve content delivery speeds for your customers in North America. Delete the CloudFront distribution after testing it and verifying that it works.

5. **Optional:** Create an Auto Scaling Group for your EC2 website. It should use a Load Balancer and have a maximum of 4 servers in the group.

6. As a group, discuss and write down your answers to these questions:
    - When would an EC2 instance be preferable to an S3 bucket for hosting a website and visa-versa?
    - What advantages are gained by using an auto-scaling group instead of a single server for hosting an application?
    - What options do you have for converting the ASG to a serverless deployment?

Optional Exercise:

If you have a Microsoft Azure account with available funds, these steps can be done on that platform using their resources instead of AWS.