PROG8830: Practical Lab 8 (Group Assignment)

Terraform Practical Assignment: Advanced Features in Action

Objective

In this assignment, you will apply Terraform's advanced features such as loops, functions, expressions, and dynamic infrastructure management to a real-world app scenario. You will gain hands-on experience by building and modifying Terraform configurations while observing how these features work in practice.

Prerequisites

- Basic understanding of Terraform concepts and commands.
- Access to an environment with the Terraform CLI installed.
- Familiarity with the app we are building, which is a web application that includes a load balancer, a database, and multiple compute instances. (If needed, refer to the project documentation for more details.)

Assignment Tasks

Task 1: Using Terraform Loops

a**. Create a Terraform configuration that demonstrates the use of both the count and for_each loop constructs.**

1. Use count to create multiple identical compute instances.
2. Use for_each to provision resources with unique configurations, such as different
3. AWS security groups based on a map of security group names and ports.

Task 2: Applying Functions in Terraform

a. Develop a set of Terraform configurations that make use of various built-in functions.

- Choose at least one function from each of the following categories:
    - String functions (e.g., upper, lower)
    - Numeric functions (e.g., min, max)
    - Collection functions (e.g., concat, length)
    - Date/time functions (if applicable)
    - IP address/networking functions (e.g., cidrsubnet)

Task 3: Enhancing Terraform Configurations

a. **Improve your Terraform configurations by incorporating advanced expressions and built-in functions.**

1. Refactor one of your existing configurations to make it more modular and dynamic using advanced expressions.

Submission Requirements

- Submit a Git repository link containing your Terraform configurations, modules, and documentation.
  - Provide a detailed README.md that explains:
  - How to run your configurations.
  - The purpose and functionality of each task.
  - The lessons learned regarding the practical differences between using count and for each, and how functions enhance your IaC practices.

Include comments in your code to explain your logic and any assumptions you made during implementation.