Lab 05: Create a Virtual Machine with Full Resource Dependency and Referencing

Objective

Apply all the core Terraform concepts learned so far to provision a **Virtual Machine** in Azure. This includes:

- Using variables
- Referencing resources (`resource.<type>.<name>.attribute`)
- String interpolation
- Explicit dependencies using `depends_on`
- Creating all resources through Terraform

Folder Naming Convention

Create and work in this folder:

```
terraform-lab-05-virtual-machine
```

📝 What You'll Build

1. Resource Group
2. Virtual Network (VNet)
3. Subnet
4. Network Interface
5. Public IP Address
6. Network Security Group
7. Virtual Machine

All these will be created through Terraform using variable references and interpolation.

Key Terraform Concepts Practiced

- `depends_on` for explicit resource ordering
- Referencing outputs from other resources
- Using `${}` for dynamic values
- Defining and consuming variables from `.tfvars`, `default`, and interactive prompts

---

📝Steps

1. **Create Folder and Files in VS Code**

   ```
   terraform-lab-05-virtual-machine/
   ├── main.tf
   ├── variables.tf
   ├── terraform.tfvars
   ```

2. **Define Variables for:**
   - Resource group name
   - Location
   - Admin username/password
   - VM size
   - VM name (default)
3. **Use Interpolation for Names**
   Example:

   ```
   name = "${var.vm_name}-nic"
   ```

4. **Use Variables from All Sources**
   - `terraform.tfvars` for sensitive info like admin credentials
   - `default` for location
   - Prompt for things like `vm_size`
5. **Deploy Everything Using:**

   ```
   terraform init
   terraform plan
   terraform apply
   ```

📦 Deliverables

- A fully working `main.tf`, `variables.tf`, and `terraform.tfvars`
- All resources deployed with proper referencing and dependency management
- VM accessible via public IP