1. **Manage Subscriptions and Governance (15-20%):** Manage Azure Active Directory (Azure AD) objects
Manage role-based access control (RBAC)
Manage subscriptions and governance
2. **Manage Azure Active Directory (Azure AD) objects:**
3. **Create users and groups:** To view the Azure AD users, access the all users page. You can define users in three ways.
Cloud identities : These users exist only in azure AD, When these users are removed from the primary directory, they are deleted.
Directory-Synchronized identities : These users exist on an on-premises Active directory.
Guest Users : these users exist outside azure. Examples are accounts from other cloud providers. Their source is invited user. Useful when external vendors or contractors need access to your Azure resources.

Azure Portal can be used to add users, Along with Azure with Microsoft 365 Admin center, Intune admin console, and the CLI
4. **Manage user and group properties:** Azure Portal can be used to manage users, Along with Azure with Microsoft 365 Admin center, Intune admin console, and the CLI. You can define two types of groups,
security groups for computer access to shared resources.
Microsoft 365 groups to give access to shared office resources.
You can assign users to groups, configure dynamic user assignment based on the members attributes, or dynamic device based on a devices attributes
5. **Manage device settings:** You can configure devices with dynamic device based on a devices attributes to remove or add it to security groups
6. **Perform bulk user updates:** Azure AD supports bulk user create and delete operations, and supports downloading lists of users. You just need to fill out a CSV template. Consider naming conventions to standardize usernames. Password convention for initial passwords for new users, figure out a secure way to deliver a users password, like randomly generating it and emailing it to a new user or their manager. PowerShell can also be used
7. **Manage guest accounts:**
8. **Configure Azure AD Join:** To get a device under the control of Azure AD, you have two options.
Registering a device to Azure AD enables you to manage a devices identity. Azure AD device registration provides the device with an identity that is used to authenticate the device when a user signs-in to Azure AD. You can use the identity to enable or disable a device.

Joining a device is an extension to registering a device. Joining provides the benefits of registering and changes the local state of a device. Changing the local state enables your users to sign in to a device using an organizational work or school account instead of a personal account.

9. **Configure Self-Service Password Reset:** From your Azure AD tenant, on the Azure portal under Azure Active Directory (Users) select Password Reset
You can set it to None, Selected, or All.
Selected allows specific groups who have self service password reset enabled.
Authentication methods
You can pick the number of methods required to reset a password. This can be a notification, a test, a code sent to user's mobile or office phones, or a set of security questions

Implement Azure AD self-service password reset - Learn | Microsoft Docs

10. **Manage role-based access control (RBAC):** Create a custom role
Provide access to Azure resources by assigning roles at different scopes
Interpret access assignments

11. **Create a custom role:** You can define specifically what you want a role to perform using a JSON file. Use the Actions and NotActions property to define what they can do, and AssignableScopes to assign where they can do it

12. **Provide access to Azure resources by assigning roles at different scopes-:** You can use the AssignableScopes property to determine what scope a user can perform their role at.

13. **Interpret Access assignments:** You can assign a role to a user, group, service principal, or managed identity. This is to grant access. Built in roles are owner, contributor, and reader.

14. **Manage subscriptions and governance:** configure Azure policies
configure resource locks
apply and manage tags on resources
manage resource groups
manage subscriptions
manage costs
configure management groups

15. **Configure Azure Policies:** Follow these steps, Browse Policy definitions, Create initiative Definitions, Scope the Initiative Definitions, View Policy evaluation Results.
When there isn't an applicable policy definition built-in you can load policies from GitHub, Policy definitions have a specific JSON format.
Once you have determined what policy definitions you need you create an Initiative

definition to group those policies and apply them to a scope
You can assign it to a subscription and then a resource group optionally
Use the compliance blade to review non-compliant initiatives, policies, or resources.
This occurs about once an hour.
16. **Configure Resource Locks:** Created to prevent accidental deletion of resources, great for vital systems.
Can be
Read-Only locks, which prevent any changes to the resource
Delete locks, which prevents deletion

These locks can be applied to a subscription, resource group, or resource.
Can only be created by owner and user access administrator roles
17. **Apply and manage tags on resources:** You can apply tags to your Azure resources to logically organize them by categories. Each tag has a name and value. After creating your tags, you associate them with appropriate resources. This is a great way to group billing data
18. **Manage Resource Groups:** Resources can be deployed to any new or existing resource group. Resource Groups are at their simplest a logical collection of resources.
There are a couple of small rules for resource groups.
Resources can only exist in one resource group.
Resource Groups cannot be renamed.
Resource Groups can have resources of many different types (services).
Resource Groups can have resources from many different regions.

All the resources in your group should share the same lifecycle
You can use locks to prevent accidental deletion
You can move resources to new groups or subscriptions
You can define resource limits with Usage + Quotas
19. **Manage subscriptions:** An Azure subscription is a logical unit of Azure services that is linked to an Azure account.
You can obtain a subscription through Enterprise agreements, Resellers, Partners, or a Personal free account.
20. **Configure management groups:** If your organization has many subscriptions, you may need a way to efficiently manage access, policies, and compliance for those subscriptions. Azure management groups provide a level of scope above subscriptions.
Using the portal, PowerShell, or azure CLI you can create management groups and instill restrictions or track spending over multiple subscriptions

21. **Implement and manage storage (15-20%):** Secure Storage
Manage Storage
Configure Azure files and Azure Blob Storage
22. **Secure Storage:** configure network access to storage accounts
create and configure storage accounts
generate shared access signature (SAS) tokens
manage access keys
configure Azure AD authentication for a storage account
configure access to Azure Files
23. **Generate shared access signature (SAS) tokens:** You can give users SAS tokens that can be configured with scope of access, ip you can access from, duration of access, and permissions in access
24. **Manage access keys:** Using Azure Key Vault you can manage your encryption keys or generate your own.
25. **Configure Azure AD authentication for a storage account:** You can use Azure AD to control who has access to a storage account as you would in AD or through RBAC
26. **Create And Configure storage accounts:** you can create and configure the following options Azure Containers (Blobs), Azure Files, Azure Queues, Azure Tables
27. **Configure network access to storage accounts:** Using the Firewalls and virtual networks blade to add the virtual network that will have access.
For file storage you need the url and a SAS token
28. **Manage Storage:** export from Azure job
import into Azure job
install and use Azure Storage Explorer
copy data by using AZCopy
implement Azure Storage replication
configure blob object replication
29. **export from Azure job:** You can identify the data in azure blobs you intend to export, identify the number of disks needed, in the azure portal create an export job referencing the azure storage account, ship the disks to the azure region hosting the account, receive the disks back from azure and decrypt the bitlocker using the keys provided in the portal
30. **import into Azure job:** Create an azure storage account, identify the number of disks you need to accomodate all the data, identify the computer that will perform the data copy, attach the disks, and use the WAImportExport tool to copy the data and encrypt it with bitlocker, Use the azure portal to create an import job referencing

your storage account, ship the disks to the destination, they will import the data for you, your disks will be mailed back to you.

31. **install and use Azure Storage Explorer:** Its a free tool you can use on your device to manage accounts in your azure subscriptions, accounts and services shared from others, and manage local storage using the azure storage emulator. if you want access to other peoples accounts and services you need a account name and account key to access the data

32. **copy data by using AZCopy:** A command line utility to copy files to and from azure blob and file storage the format is azcopy copy source destination flags

33. **Configure Azure files and Azure Blob storage:** create an Azure file share
create and configure Azure File Sync service
configure Azure Blob Storage
configure storage tiers
configure blob lifecycle management

34. **Create An Azure File Share:** Create a Storage account, Provide the file share name and quota, Quota is total size of files on the share. Select connect from your file share page to connect windows or windows server. Ensure port 445 for SMB communications. the CIFS kernel client can be used to mount in Linux distributions. Secure transfer is required so use HTTPs or it wont work.

35. **Create and configure Azure File Sync Service:** Deploy the Storage Sync service from the azure portal, Prepare windows server to use the Azure File Sync (latest powershell version and disabling ie security for a moment), Installing the Azure File Sync Agent, Registering Windows Server with the Storage Sync Service

36. **Configure Azure Blob Storage:** You can configure blob storage to hold as many containers or blobs as you wish. You can upload different types of blobs with different strengths.

37. **Configure Storage tiers:** You can use hot, cool or the archive tier. Switching between hot to cool is instant. It becomes more expensive to access the data as you go from hot to archive

38. **Configure Blob Lifecycle management:** You can configure blobs to move between tiers based on its lifecycle or automatically delete itself after a certain time.

39. **Deploy and manage Azure compute resources (20-25%):** Automate deployment of virtual machines (VMs) by using Azure Resource Manager templates
Configure VMs
Create and configure containers
Create and configure Azure App Service

40. **Automate deployment of virtual machines (VMs) by using Azure Resource Manager templates:** modify an Azure Resource Manager template
configure a virtual hard disk (VHD) template

deploy from a template
save a deployment as an Azure Resource Manager template
deploy virtual machine extensions

41. **Modify an Azure Resource Manager template.:** A template is a JSON text file that can be edited in any of the below fields.

$schema
contentVersion
parameters
variables
functions
resources
outputs

Primarily you can update parameters to define configurable attributes with each deployment in a separate file

You can customize your template from the Template deployment tool in the portal or for production you can use Visual Studio Code

42. **Deploy from a template.:** In the search box type deploy and select deploy a custom template, you can load a JSON file from your computer, an azure storage account, or git repository. You can then edit it from there if its not to your liking before deploying it. Each azure service has a unique name so you need to add a variable to create a unique string for deployments, or it will fail.

43. **Save a deployment as an Azure Resource Manager template.:** You can use QuickStart templates to get community generated templates, or select add a resource and configure it to your liking, and before you hit create select download a template for automation

44. **Deploy virtual machine extensions:** These are Small applications that provide post-deployment configuration and automation tasks on Azure VMs. Custom script extensions can be created in PowerShell and implemented on VMs by by accessing the virtual machines Extensions blade.

45. **Create and Configure containers:** Configure sizing and scaling for Azure container instances
Configure container groups for azure container instances

46. **Configure sizing and scaling for Azure container instances:**

47. **Configure container groups for azure container instances:**

48. **Create and Configure Azure App Service:** Create an App Service.
Secure an App Service.

Configure custom domain names.
Configure backup for an App Service.
Configure networking settings.
Configure deployment settings.
Create an App Service plan.
Configure scaling settings in an App Service plan

49. **Create an App service:** You need to specify a resource group and a service plan along with some other information.
Name - that's unique and can be used to locate your app.
Publish - with code or a Docker Container
Runtime stack - the software stack that runs the app, including the language and SDK versions.
Operating System - Linux or Windows
Region - for availability purpose

50. **Secure an App Service:** Additional configuration options include
Always on - Keep the app loaded even when there is no traffic.
ARR affinity - In a multi-instance deployment, ensure the client is routed to the same instance for the life of the session
Connection strings - Encrypted at rest and transmitted over an encrypted channel
Built in support for authentication and authorization, so you can sign in users and access data with minimal or no code in your web app, api, and mobile back end, and also Azure Functions. It runs in the same sandbox as your application code. When enabled, every incoming http request passes through it before being handled by your application code.
Allow Anonymous requests - defers authorization of unauthenticated traffic to your application code
Allow only authenticated requests - Redirects all anonymous requests for the provider you choose.

51. **Configure Custom Domain Names:** Reserve your domain name / Create DNS records that map the domain to your Azure web app / Enable the custom domain

52. **Configure backup for an App Service.:** If your app is the standard or premium tier you can create app backups manually or on a schedule, you configure how long the backups are retained, you can restore the app to a snapshot of a previous state. Can back up App configuration, file content, and databases connected to your app.

53. **Configure networking settings:** You can configure DNS records to map your apps domain name to an IP address, CNAME records can be used to assign a custom domain name to your apps domain name.

54. **Configure deployment settings:** You can set up a CI CD pipeline by using azure DevOps, GitHub, or Bitbucket and automate a push test review and deploy process, in addition to that you can use manual code to update apps.
55. **Create an App Service plan:** App service plans determine location, VM version, and number of vm instances. Can be spread over pricing tiers.
56. **Configure scaling settings in an App Service plan:** Can be based on metrics or time.
57. **Configure VMs:** configure Azure Disk Encryption
move VMs from one resource group to another
manage VM sizes
add data disks
configure networking
redeploy VMs
configure high availability
deploy and configure scale sets
58. **Move VMs from one resource group to another.:** Select the resource group containing this resources and select the move button, acknowledge that you need to update scripts.
59. **Manage VM sizes:** VM sizes scale based on your use, A being the entry level option, down to specialized workloads for high compute, gpu attachments, or storage optimized
60. **Add Data disks:** Decide if it will be managed or Unmanaged. Can be configured during the virtual machine creation process,
61. **Configure Networking:** Can be managed during the virtual machine creation process, Connection methods include RDP, SSH, or Bastion
62. **Redeploy VMs:**
63. **Configure High availability:** High availability can be controlled with Availability Sets and Availability zones. There are various SLAs you can count on, Two instances across two AZs give you 99.99%, Two instances deployed in the same availability set give you 99.95% connectivity, One instance using premium OSdisks and Data disks give you 99.9%. You can use Availability sets to control the region the instances are deployed in, how many fault and update domains they have, and if they use managed disks, option changes can increase the cost. Availability zones offer resilience past faults and updates, and for disasters and unaccounted for events like datacenter failures.
64. **Deploy and configure scale sets:** Scale sets let you deploy a managed set of identical VMs, You can implement automatic scaling (autoscale) up to 1000 vm instances, or 600 vm instances with a custom vm image. You decide how many to deploy, if its a custom image, if there is the ability to use autoscale for elastic scaling.

65. **Monitor and back up Azure resources (10-15%):** Monitor resources by using Azure Monitor
66. **Monitor resources by using Azure Monitor:** Configure Application Insights.
67. **Configure Application Insights.:** A feature of azure monitor, monitors your live applications. It detects performance anomalies and includes analytics tools to help you diagnose issues and understand user actions in your app. Can be included in a CI CD pipeline