Access the AWS Management Console

1. After creating your account, log in to the AWS Management Console.
2. Familiarize yourself with the dashboard, which lists all AWS services.

SaaS on AWS

Example Service: Amazon WorkSpaces (Virtual Desktops)

1. Go to the AWS Management Console and search for **Amazon WorkSpaces**.
2. Click on **Launch WorkSpaces**.
3. Select the **WorkSpace bundle**:
    1. Standard (Windows or Linux).
    2. Choose the desired configuration (CPU, memory, storage).
4. Provide user details for WorkSpaces access.
5. Launch the WorkSpace and access it via the AWS-provided client application.
6. **Use Case**: A virtual desktop as a service is a classic SaaS example.

PaaS on AWS

Example Service: AWS Elastic Beanstalk

1. In the AWS Management Console, search for **Elastic Beanstalk**.
2. Click on **Create Application**.
3. Provide the following details:
    1. **Application Name**: e.g., "MyApp".
    2. **Platform**: Choose a development platform (e.g., Python, Node.js, Java).
4. Upload your application code as a ZIP file.
5. Select a deployment configuration (default is usually sufficient for beginners).
6. Click **Create Application**, and Elastic Beanstalk will handle the deployment.
7. Access your deployed application via the URL provided.
8. **Use Case**: Simplifies application deployment without managing infrastructure.

IaaS on AWS

Example Service: Amazon EC2 (Elastic Compute Cloud)

1. In the AWS Management Console, search for **EC2**.
2. Click on **Launch Instance**.
3. Choose the following:
    1. **AMI (Amazon Machine Image)**: Select an OS (e.g., Ubuntu, Windows Server).
    2. **Instance Type**: Select the required configuration (e.g., t2.micro for free tier).
4. Configure instance details (number of instances, network settings).
5. Add storage if required.
6. Review and launch the instance.
7. **Key Pair**: Download the key pair for SSH access (for Linux instances).
8. Use SSH (Linux) or RDP (Windows) to connect to your instance.
9. **Use Case**: Provides complete control over virtualized servers and networks.

Step 3: Manage Resources

- Use the AWS Management Console to monitor and manage the resources created.
- Ensure to **stop or terminate unused resources** to avoid incurring unnecessary charges.