# Getting Started with Managed Services

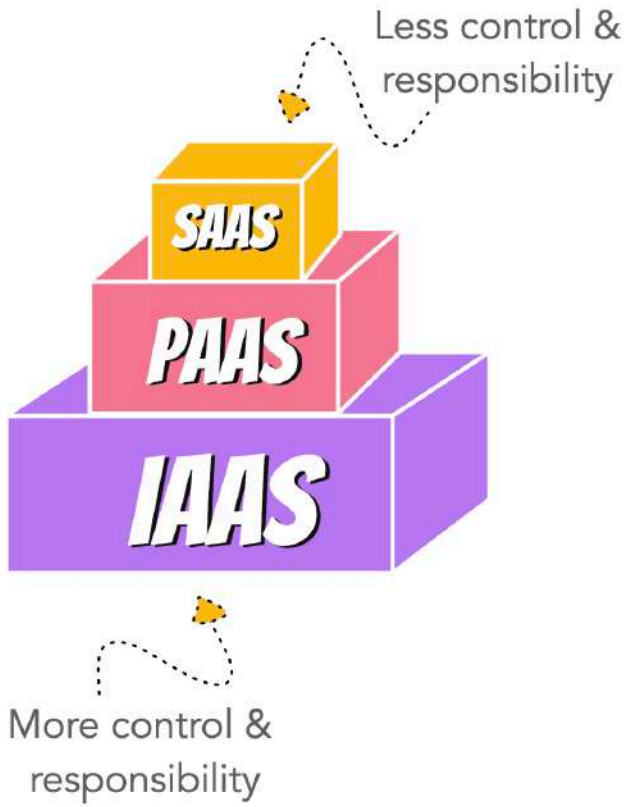
**Question**: Do you want to run apps in the cloud like you do in a data center?

- **Alternative**: Or are there better cloud-native approaches?

You should **understand some terminology**:

- **IaaS** (Infrastructure as a Service)
- **PaaS** (Platform as a Service)
- **SaaS** (Software as a Service)

Ready for a journey to understand these?



# Example 1A: Setting up an App - IaaS Approach

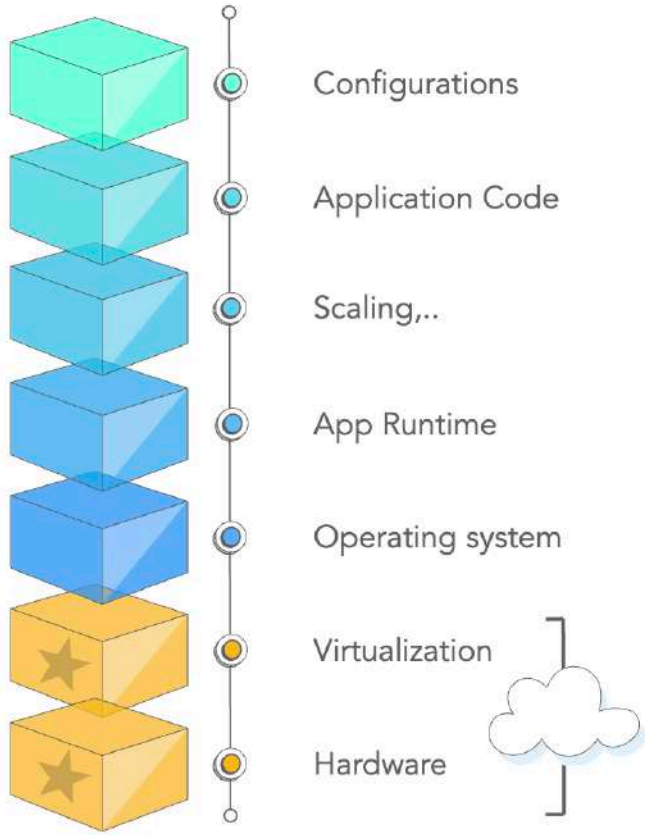
**1: Provision VMs** – Create VMs on cloud

**2: Install Software** – Install OS, runtimes etc

**3: Deploy Code** – Upload code and configure your application manually

**4: Configure Ops** – Set up load balancing and auto scaling yourself

**5: Maintain** – You are responsible for OS patching, runtime upgrades and server health



# Example 1B: Setting up an App - PaaS Approach

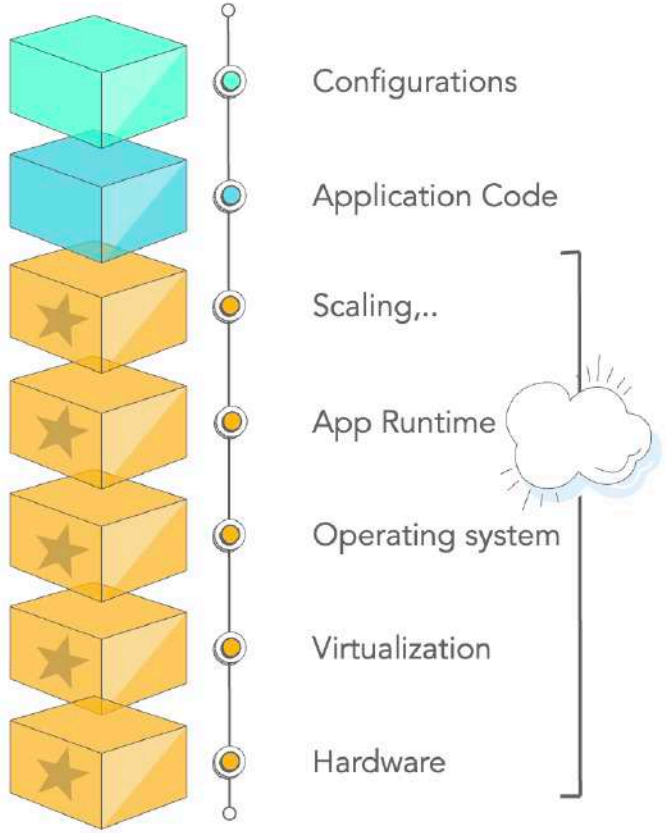
**1: Pick a Platform** – AWS Elastic Beanstalk, Azure App Service, or Google App Engine

**2: Upload Code** – Upload your code artifact

**3: Automatic Setup** – Platform provisions VMs, installs OS, and starts app

**4: Built-in Ops** – Scaling and Load Balancing are easily configurable

**5: Zero Maintenance** – No Manual OS patching or server management required



# Example 2A: Setting up a SQL Database - IaaS Approach

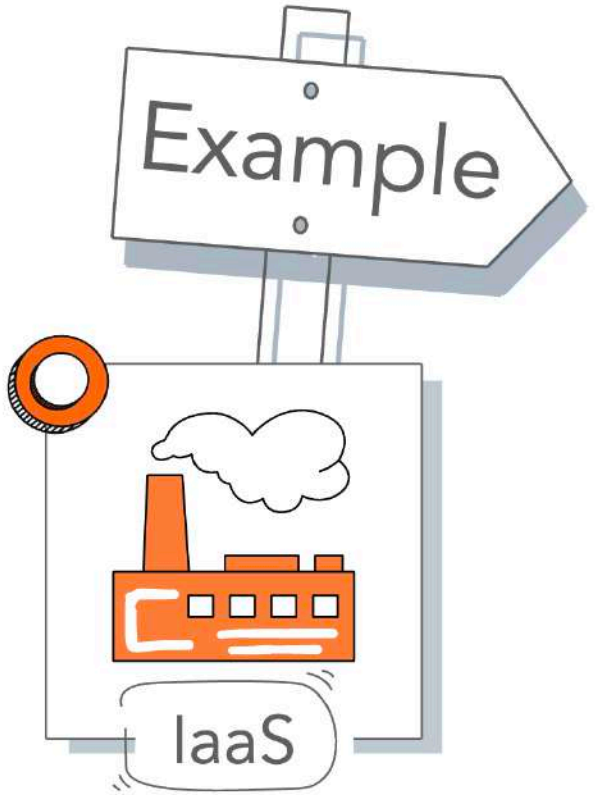
**1: Provision VMs** – Create VMs to host your database

**2: Install Database** – Set up MySQL, PostgreSQL, or any DB you need

**3: Configure Storage** – Attach disks

**4: Set Up Ops** – Manually configure backup jobs and replication

**5: Maintenance** – Manage scaling, patches, and monitor performance



# Example 2B: Setting up a SQL Database - PaaS Approach

**1: Pick a Service** – AWS RDS, Azure SQL Database, or Google Cloud SQL

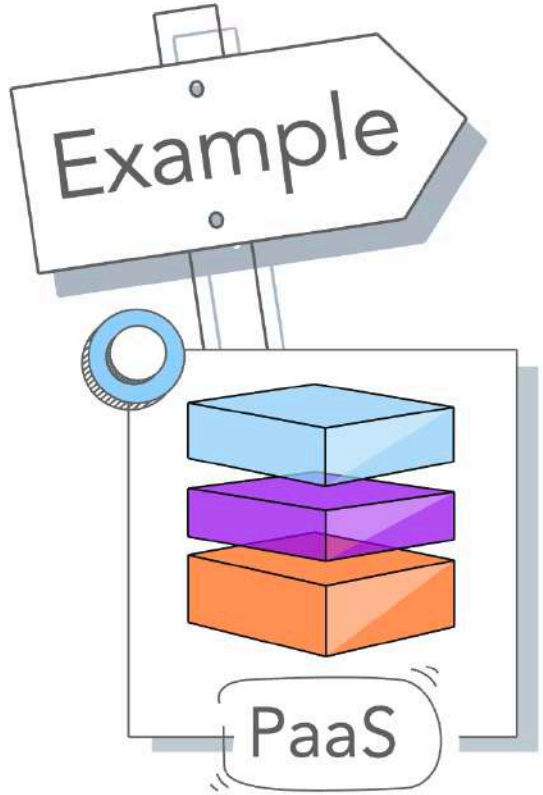
**2: Choose Type** – MySQL, PostgreSQL, SQL Server, etc

**3: Configure** – Decide CPU, memory, and storage size

**4: Enable Ops** – Turn on built-in backups & high availability

- **Ready** – Database is ready to use

**Focus**: Data and queries, NOT servers or maintenance



# What is IaaS?

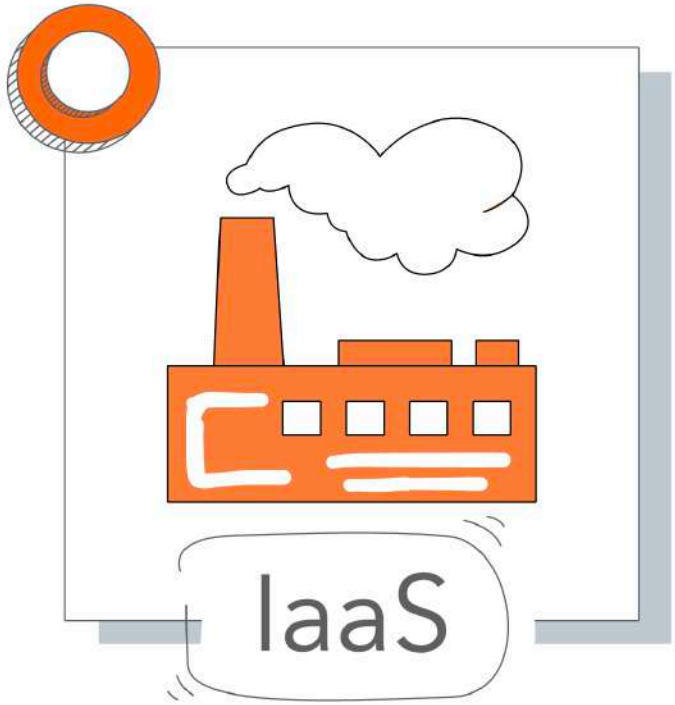
**IaaS**: Infrastructure as a Service

- **Concept**: Use only infrastructure from the cloud provider

**Customer Responsibility**: High

- **Components**: OS, runtime, and system software
- **Operations**: Availability, durability, and scaling

**Summary**: You manage almost everything except the hardware



# What is PaaS?

**PaaS (Platform as a Service)**: Use a complete platform provided by the cloud

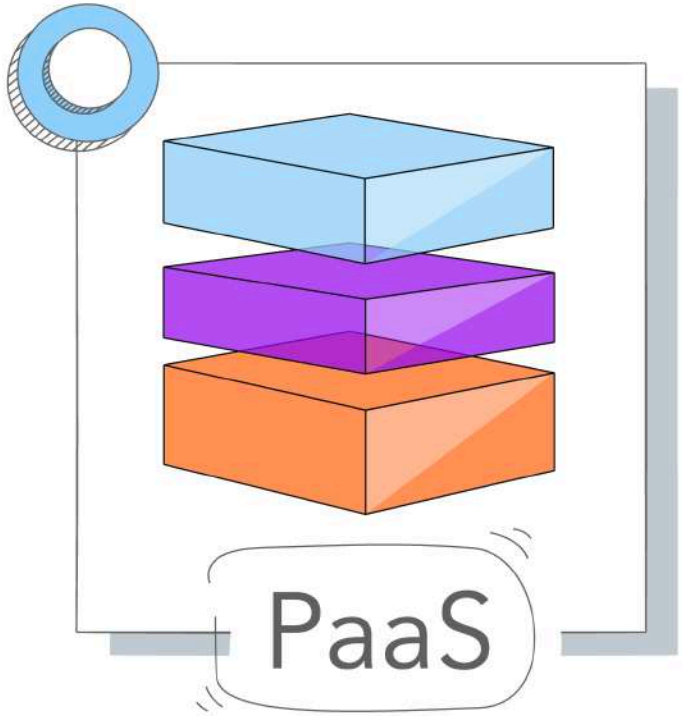
**Cloud Provider Responsibility**: High

- **Components**: OS, runtime, availability, durability, scaling

**Customer Responsibility**: Low

- **Focus**: Code, data, and configuration

**Summary**: Cloud provider manages the routine work



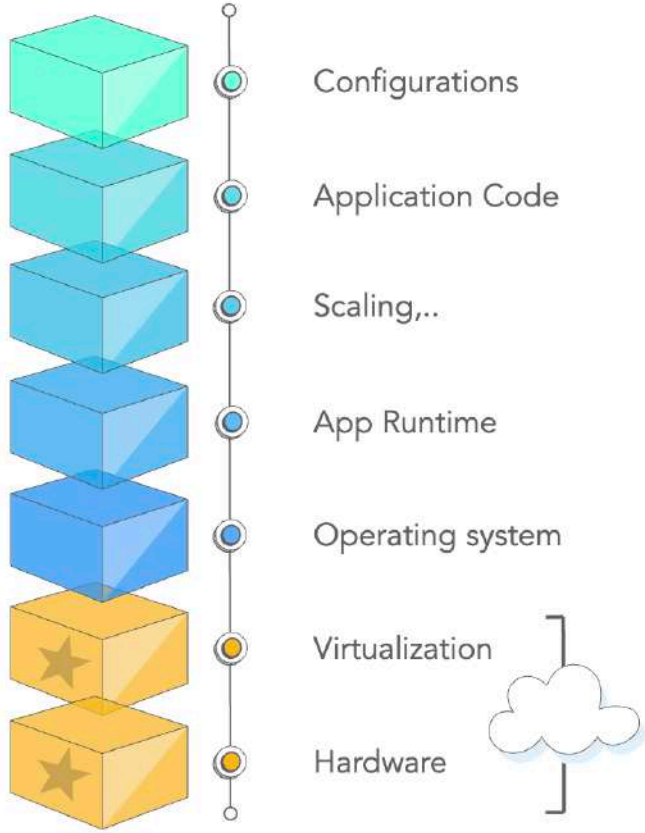
# Who is Responsible for What? - IaaS

## **Cloud platform**: The Basics

- Virtualization, Hardware and Networking

## **Customers of cloud**: Everything Else

- **Application**: Code and Runtime
- **Operations**: Load balancing, Auto scaling, Availability
- **Database**: Software, upgrades, configuration (tables, views,..), and data
- **OS**: Upgrades and patches



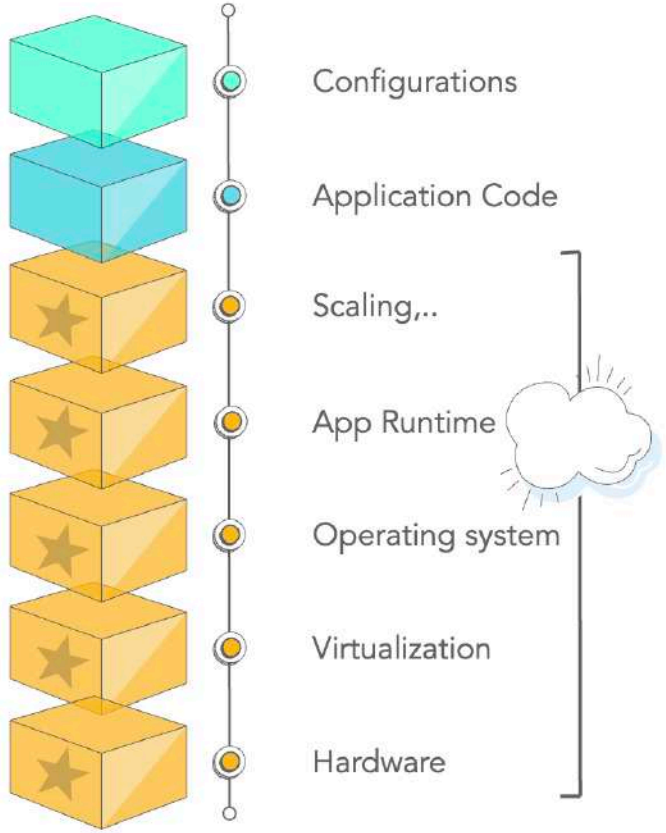
# Who is Responsible for What? - PaaS

## **Cloud Platform**: Heavy Lifting

- **Infrastructure**: Virtualization, Hardware, Networking, **OS*: Upgrades and patches
- **Platform**: Runtime, Database software, Scaling, Availability, Durability, Load Balancing

## **Customer**: Business Logic

- **Application**: Code & configuration
- **Database**: Schema, Tables, and Data
- **Restricted Access**: Usually no access to OS or VM instances



# PaaS Services Can Do a Lot More

**Scope**: PaaS is NOT limited to running apps and databases

**Managed Services**: Includes common building blocks

- **Storage**: Amazon S3, Azure Blob Storage, Google Cloud Storage
- **Analytics**: BigQuery, Azure Synapse, AWS Redshift

**Goal**: Let developers focus on logic and innovation

- **Benefit**: Skip the setup and maintenance

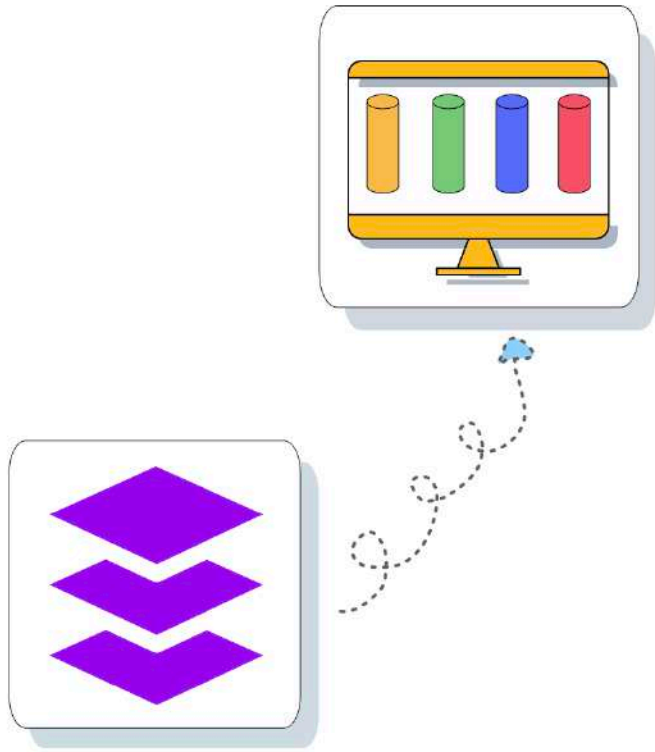
# From Platforms to Complete Applications

**IaaS**: You manage most things yourself (infra, scaling, app, data)

**PaaS**: The platform helps you focus on your application and data

**SaaS**: You use complete software - no setup needed!

**Next Step**: Let’s explore SaaS (Software as a Service)



# What is SaaS (Software as a Service)?

**Definition**: Centrally hosted software

- **Hosting**: Mostly on the cloud

**Billing**: Offered on a subscription basis (Pay as you go)

**Examples**:

- **Email**: Gmail, Outlook 365
- **Office tools**: Microsoft Office 365, Google Docs
- **Calendar**: Google Calendar

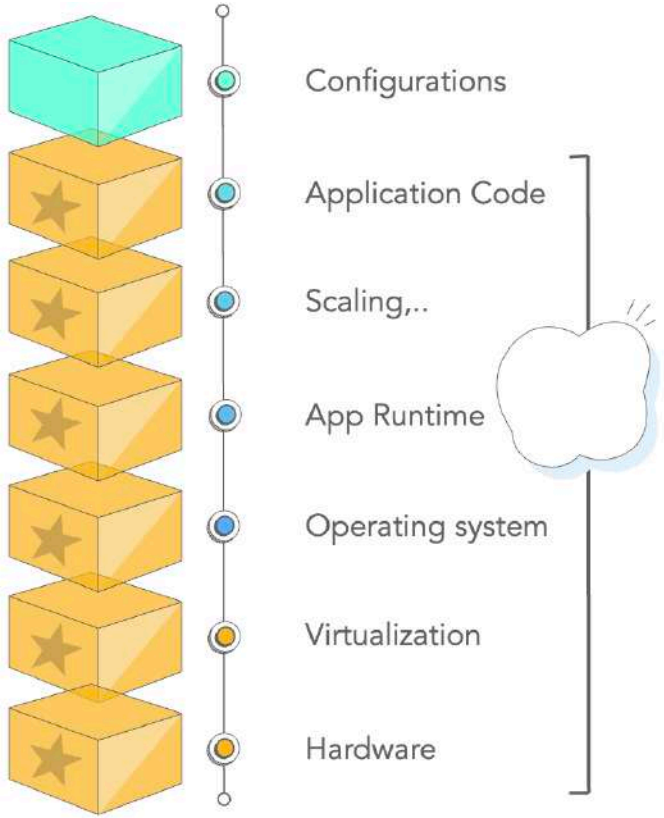
# Who is Responsible for What? SAAS

**Service Provider**: Almost Everything

- **Infrastructure**: OS, Runtime, Networking
- **Application**: Code updates and maintenance
- **Operations**: Auto scaling, Availability, Load balancing

**Customer**: Usage

- **Content**: Documents, emails, sheets
- **Configuration**: Users, permissions etc



# Quick Recap - IaaS vs PaaS vs SaaS

**IaaS**: Platform provides infra

- **You**: Manage all other layers

**PaaS**: Platform handles infra, setup, and scaling

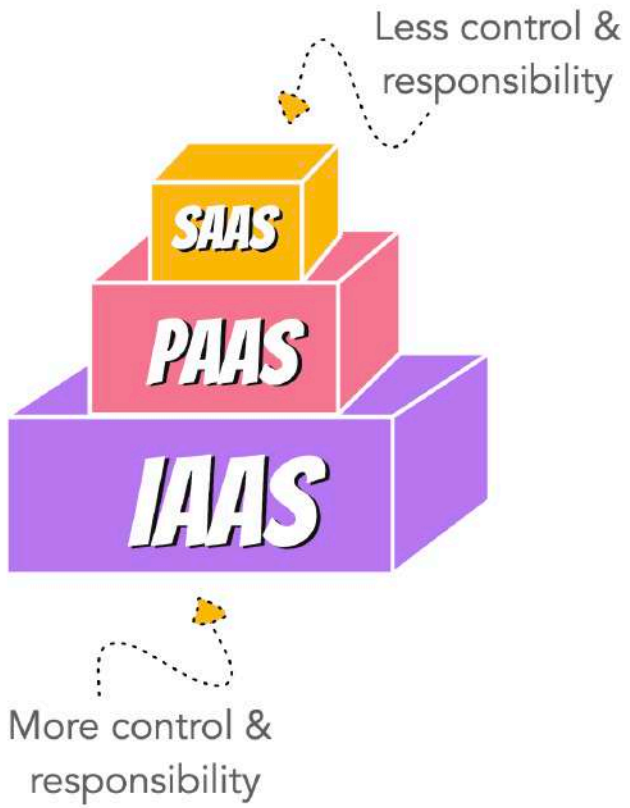
- **You**: Handle code, configuration, and data

**SaaS**: Platform provides everything

- **You**: Just use the app

**Trend**: Moving from IaaS → PaaS → SaaS

- **Control**: Decreases
- **Ease of Use**: Increases



# Service Categories - Scenarios

| Scenario | Solution |
| --- | --- |
| IaaS or PaaS or SaaS: Deploy Custom Application in Virtual Machines | IaaS |
| IaaS or PaaS or SaaS: Deploy a Database in Virtual Machines | IaaS |
| IaaS or PaaS or SaaS: Using Gmail | SaaS |
| IaaS or PaaS or SaaS: Using a Managed Service to setup a database | PaaS |
| True or False: Customer is responsible for OS updates when using PaaS | False |
| True or False: Customer is completely responsible for Availability when using PaaS | False |
| True or False: In PaaS, customer has access to VM instances | False |
| True or False: In PaaS, customer can customize OS and install custom software | False |
| True or False: PaaS services only offer Compute services | False |
| True or False: In PaaS, customer can configure hardware needs (memory, cpu etc) | True |

# Choosing Between IaaS, PaaS, and SaaS - Scenarios

| Scenario | Solution |
| --- | --- |
| Legacy Migration: You must migrate a 10-year-old app that requires a specific, older version of the OS with customization | IaaS |
| Rapid Development: A startup wants to deploy a new NodeJS web app immediately. They have no Ops team and don't want to manage servers | PaaS |
| Productivity: You need email, calendar, and document collaboration tools for 500 employees starting tomorrow | SaaS |
| Custom Database: You need to run a specialized database engine that needs customization of Operating System | IaaS |
| CRM Needs: Your sales team needs a tool to track leads and customers. You want zero maintenance overhead | SaaS |