Python Online Free Ramzan Course 2025
Taught by: Shaida Muhammad

Deployments in Python

Day 19 - Python Basics

Shaida Muhammad

Thursday, March 27, 2025

ShaidaSherpao@gmail.com

Agenda

1 What is Deployment?

2 Desktop Application Deployment

3 Web Application Deployment

4 Automation & Scripting Deployment

5 Machine Learning Model Deployment

6 Microservices & APIs Deployment

7 Comparison of Deployment Methods

8 Best Practices for Deployment

Introduction

- What is Deployment?
- Why Deployment is Necessary?
- Different Types of Python Applications:
  - **Web Applications**
  - **Desktop Applications**
  - **Automation & Scripting**
  - **Machine Learning Models**
  - **Microservices & APIs**

Abbreviations

- **PaaS** - Platform as a Service
- **IaaS** - Infrastructure as a Service
- **FaaS** - Function as a Service
- **CI/CD** - Continuous Integration / Continuous Deployment
- **GCP** - Google Cloud Platform
- **AWS** - Amazon Web Services
- **OS** - Operating System
- **API** - Application Programming Interface

Desktop Application Deployment

1. Standalone Executables (.exe/.dmg/.AppImage)

- **Tools Used:** PyInstaller, cx_Freeze, Py2exe
- **Process:** Convert Python script to executable
- **Pros:**
  - No need to install Python on the target system
  - Simple distribution
- **Cons:**
  - Large file size
  - Dependency issues on different OS
- **Cost:** One-time cost, Free (open-source tools), but potential licensing costs for commercial software

Desktop Application Deployment ...

2. Microsoft Store & Mac App Store

- **Tools Used:** Electron + Python, PyQt
- **Pros:**
  - Easy updates
  - Wider reach
- **Cons:**
  - Requires app store approval
  - Profit-sharing with store
- **Cost:** Annual developer fee ($99/year for Apple, $19 one-time for Microsoft)

Desktop Application Deployment ...

3. Docker Containers

- **Tools Used:** Docker, Docker Compose
- **Pros:**
  - Runs consistently across different environments
  - Solves dependency issues
- **Cons:**
  - Requires Docker setup on the user's machine
- **Cost:** Free for local use; Docker Hub plans start at $0/month to $24/month

Web Application Deployment

1. Shared Hosting

- **Providers:** Bluehost, Hostinger, GoDaddy
- **Pros:**
  - Cheap
  - Good for small apps
- **Cons:**
  - No root access
  - Limited scalability
- **Cost:** $3–$10/month

Web Application Deployment ...

2. Virtual Private Server (VPS, IaaS)

- **Providers:** DigitalOcean Droplet, Linode, AWS EC2, Azure VM, Google Compute Engine, Oracle VM
- **Pros:**
  - Full control over server
  - More performance
- **Cons:**
  - Requires server management knowledge
- **Cost:** $4–$999/month but can be in thousands as of machine specs (GPUs)

Web Application Deployment ...

3. Platform as a Service (PaaS)

- **Providers:** Heroku, PythonAnywhere, Google App Engine, Render
- **Pros:**
  - Simple deployment
  - No server maintenance
- **Cons:**
  - Limited resources in free plans
  - Cost increases with scale
- **Cost:** Free to $50+/month depending on resources

Web Application Deployment ...

4. Cloud Deployment (FaaS)

- **Providers:** DigitalOcean Functions, AWS Lambda, Google Cloud Functions, Azure Functions
- **Pros:**
  - Highly scalable
  - Serverless
  - Cost-efficient for small apps
- **Cons:**
  - Cold start issues when idle for sometime
- **Cost:** Pay-as-you-go, starts at cents per million requests

Web Application Deployment ...

5. Containerized Deployment with Kubernetes

- **Tools Used:** Docker, Kubernetes
- **Providers:** Google Kubernetes Engine, Amazon Elastic Kubernetes Service, Azure Kubernetes Service, DigitalOcean Kubernetes
- **Pros:**
  - Scalable and resilient
  - Works across clouds
- **Cons:**
  - Complex setup
  - Requires DevOps knowledge
- **Cost:** $10–$100/month (depends on cloud provider) and can go up, cost can be hourly or on demand

Automation & Scripting Deployment

1. CRON Jobs & Task Scheduler

- **Use Cases:** Running scripts automatically at intervals
- **Pros:**
  - No need for continuous execution
  - Lightweight
- **Cons:**
  - Limited logging and monitoring
- **Cost:** Free (built into Linux/Windows)

Automation & Scripting Deployment ...

2. System Services (Linux Systemd, Windows Services)

- **Use Cases:** Running automation scripts as background services, Backup etc.
- **Pros:**
  - Runs continuously
  - Logs outputs properly
- **Cons:**
  - Requires system admin access
- **Cost:** Free (built into OS)

Machine Learning Model Deployment

1. Flask/Django API

- **Use Cases:** Deploy models as REST APIs
- **Pros:**
  - Simple
  - Works with web frameworks
- **Cons:**
  - Not optimized for high performance
- **Cost:** Free (but requires hosting, ~$5–$50/month)

Machine Learning Model Deployment ...

2. FastAPI for High Performance

- **Pros:**
  - Async support
  - Faster than Flask
- **Cons:**
  - Requires more configurations
- **Cost:** Free (but requires hosting, ~$5–$50/month)

Machine Learning Model Deployment ...

3. Cloud ML Platforms

- **Providers:** Google AI Platform, AWS Sagemaker, Azure Machine Learning
- **Pros:**
  - Auto-scalability
  - Pre-built optimizations
- **Cons:**
  - Expensive
  - Vendor lock-in
- **Cost:** $50–$500/month (depends on usage)

Microservices & APIs Deployment

1. Docker Containers with Kubernetes

- **Best for:** Scalable APIs & Microservices
- **Pros:**
  - Easy horizontal scaling
- **Cons:**
  - DevOps expertise required
- **Cost:** $10–$100/month (varies with cloud provider)

Microservices & APIs Deployment ...

2. Serverless (AWS Lambda, GCP Functions)

- **Pros:**
  - No need to manage infrastructure
- **Cons:**
  - Cold start delays
- **Cost:** Pay-as-you-go, starts at cents per million requests

Important Note

The prices mentioned in this document are approximate and may fluctuate over time. Different cloud providers offer varying pricing models and features. It is difficult to list exact prices, but this guide aims to provide a comprehensive cost estimation.

Best Practices for Deployment

- Use **CI/CD pipelines** for automation (GitHub Actions, GitLab CI)
- Use **environment variables** instead of hardcoding credentials
- Keep dependencies updated
- Monitor logs and errors
- Optimize performance before deploying
- Different types of Python applications require different deployment methods
- Choose the right method based on requirements and budget

Q&A

- Do you have any questions?
- Share your thoughts.

Closing

Next class: Live Project in Python