

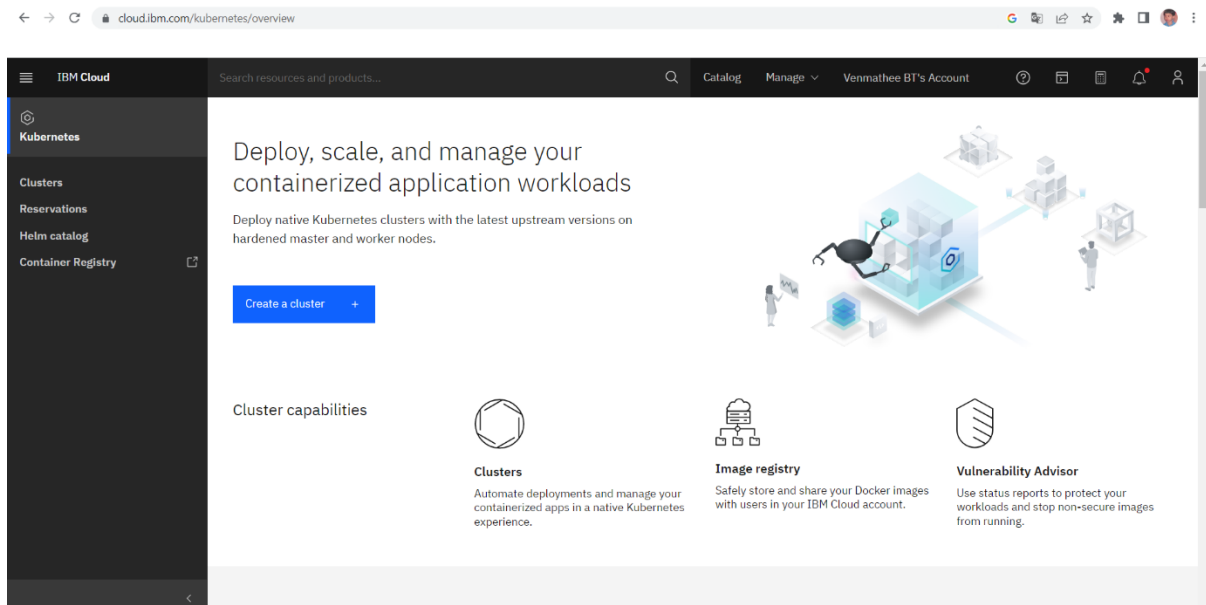
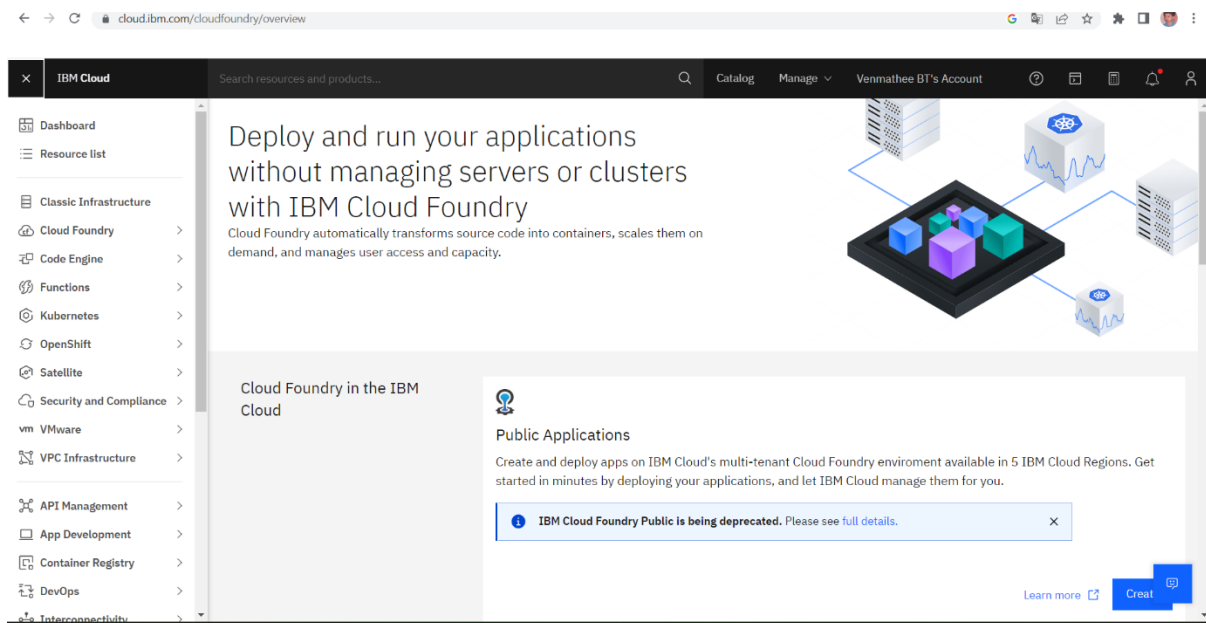
DEPLOYMENT OF APP IN IBM CLOUD

UPLOAD IMAGE TO IBM CONTAINER REGISTRY

| | |
|--------------|--------------------------|
| TEAM ID | PNT2022TMID23085 |
| PROJECT NAME | PLASMA DONOR APPLICATION |

Create a Kubernetes cluster

- Sign in to your [IBM Cloud Dashboard](#).
- Open **IBM Kubernetes Service**.



cloud.ibm.com/kubernetes/catalog/create

IBM Cloud Search resources and products...

Kubernetes / Clusters /

Kubernetes cluster

Author: IBM • Docs • API docs

Create About

Plan details
Learn more about the differences between plans in our docs.

Pricing plan
Standard

Infrastructure
Choose which network and compute environment to run your cluster on. [Learn more about the differences.](#)

Classic
Run your cluster with native subnet and VLAN networking on our classic infrastructure.

VPC
Create a fully customizable, software-defined virtual network with superior isolation using IBM Cloud VPC.

Summary United States

Kubernetes cluster

- Worker nodes** US\$2.65/hr
b3c.4x16 - 4 vCPUs 16GB RAM
Virtual - shared
Ubuntu 18
- Multizone load balancer** US\$0.02/hr
Multizone clusters require a cross-zone load balancer.

Total estimated cost US\$1,919.52/mo

*Additional charges for networking and bandwidth might apply.
Actual monthly total will vary with tiered pricing.
Estimate does not include costs for integrations.*

Upgrade to create

Add to estimate

- Click on the **Worker Nodes** tab to note the cluster's Public IP.

cloud.ibm.com/kubernetes/clusters/cdmfv7if085nhdamd2cg/nodes

IBM Cloud Search resources and products...

Clusters /

mycluster-free

Preparing master, workers... Expires in 30 days Add tags

Overview

Worker nodes

Worker pools

DevOps New

Pool: Filter... Search

| Name | Status | Worker pool | Zone | Private IP | Public IP | Version |
|----------|-------------------|-------------|----------|------------|-----------|-----------------|
| 0000002a | Provision pending | default | Milan 01 | | | --> 1.24.7_1543 |

ID
kube-cdmfv7if085nhdamd2cg-myclusterfr-default-0000002a

Status -- Flavor Free - 2 vCPUs 4GB RAM Private VLAN 2218181 Public VLAN 2218179

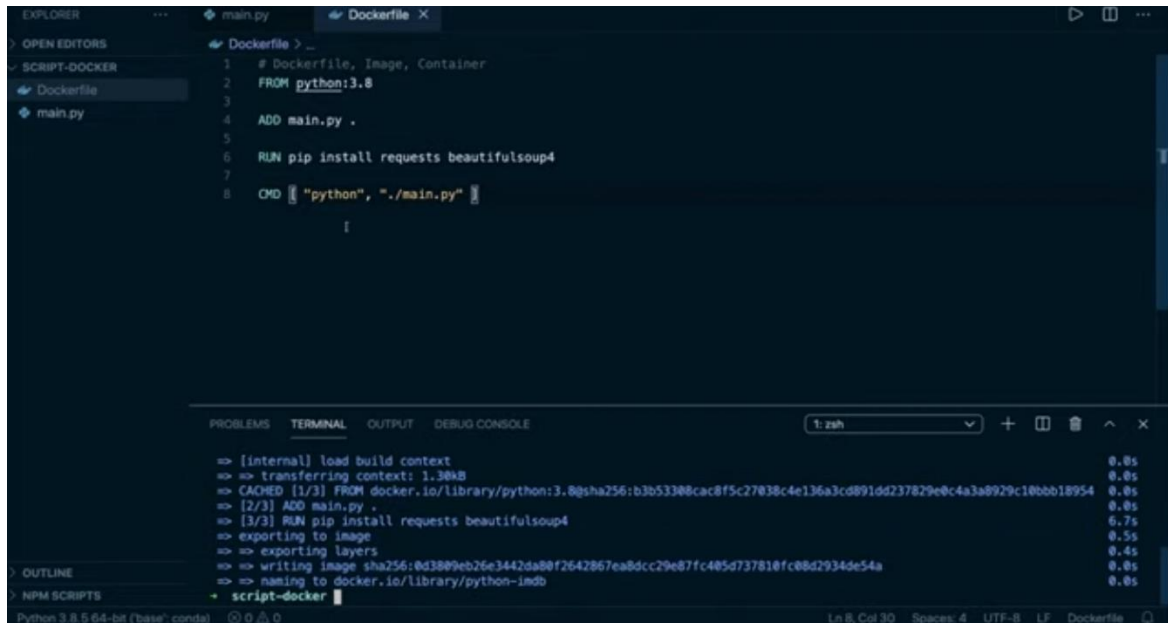
Items per page: 25 1-1 of 1 item 1 1 of 1 page

Help

- Log in to your cluster
- Deploy your app
- Expose your app
- Add storage to your app
- Connect integrations
- Install add-ons
- Troubleshoot

10-11-2022

Containerize your Flask application



The screenshot shows a VS Code editor with a file explorer on the left containing 'main.py' and 'Dockerfile'. The 'Dockerfile' is open in the editor, showing the following content:

```
# Dockerfile, Image, Container
FROM python:3.8

ADD main.py .

RUN pip install requests beautifulsoup4

CMD ["python", "./main.py"]
```

The terminal at the bottom shows the build process for the Docker image:

```
[internal] load build context
=> transferring context: 1.38kB
=> CACHED [1/3] FROM docker.io/library/python:3.8@sha256:b3b53308cac8f5c27038c4e136a3cd891dd237829e0c4a3a8929c10bbb18954
=> [2/3] ADD main.py .
=> [3/3] RUN pip install requests beautifulsoup4
=> exporting to image
=> exporting layers
=> writing image sha256:ed3809eb26e3442da80f2642867ea8dccc29e87fc485d737810fc88d2934de54a
=> naming to docker.io/library/python-lmbd
-> script-docker
```

```
Step 1/8 : FROM python:3.8
--> 6c76e39e7cfe
Step 2/8 : LABEL maintainer="Kunal Mishra, kunal.mishra@ibm.com"
--> Using cache
--> d857d41531c
Step 3/8 : RUN apt-get update
--> Using cache
--> 6262a134e40e
Step 4/8 : COPY ./app
--> f07f737099f
Step 5/8 : WORKDIR /app
moving intermediate container f9010e99a2fe
--> 08c6ef20e3d
Step 6/8 : RUN pip install -r requirements.txt
--> Running in 815304000b7
collecting click==6.7 (from -r requirements.txt (line 1))
  Downloading https://files.pythonhosted.org/packages/34/c1/8806f99713ad993c5366c36b2f908f18269f8d79a0f1abfd700775a77/click-6.7-py2.py3-none-any.whl (71kB)
collecting Flask==1.0.2 (from -r requirements.txt (line 2))
  Downloading https://files.pythonhosted.org/packages/7f/e7/08578774ed4536d3242b14d0cb4e96386634607af824ee997202c00db4b/Flask-1.0.2-py2.py3-none-any.whl (93kB)
collecting itsdangerous==0.24 (from -r requirements.txt (line 3))
  Downloading https://files.pythonhosted.org/packages/dc/b4/a60bcabc945c00f6a008d8975131ab3f25b22f2bcfd0b221165194b204/itsdangerous-0.24.tar.gz (40kB)
collecting Jinja2==2.10 (from -r requirements.txt (line 4))
  Downloading https://files.pythonhosted.org/packages/7f/f6/ae64b0cdfc95f27a016a7bed8e8686763b0fd27a78ca76f32659220a731/Jinja2-2.10-py2.py3-none-any.whl (126kB)
collecting MarkupSafe==1.0 (from -r requirements.txt (line 5))
  Downloading https://files.pythonhosted.org/packages/4d/86/2d741db316d8fd7680822d37001ef7a448255de9699ab4bfcbdf4172b/MarkupSafe-1.0.tar.gz
collecting Werkzeug==0.14.1 (from -r requirements.txt (line 6))
  Downloading https://files.pythonhosted.org/packages/20/c4/12e3e56473e52575aa29ca764a70d1b8f3ef06682b0f8d00aa94fe335243/Werkzeug-0.14.1-py2.py3-none-any.whl (322kB)
building wheels for collected packages: itsdangerous, MarkupSafe
Running setup.py bdist_wheel for itsdangerous: started
Running setup.py bdist_wheel for itsdangerous: finished with status 'done'
Stored in directory: /root/.cache/pip/wheels/2c/4a/61/559631c1554768c6290b08c02c72d7317910374ca082ff1e5
Running setup.py bdist_wheel for MarkupSafe: started
Running setup.py bdist_wheel for MarkupSafe: finished with status 'done'
Stored in directory: /root/.cache/pip/wheels/33/56/20/ebef9a5c612ffef1c5a632140b16996f9e64676768661e4e46
successfully built itsdangerous MarkupSafe
installing collected packages: click, itsdangerous, MarkupSafe, Jinja2, Werkzeug, Flask
successfully installed Flask-1.0.2 Jinja2-2.10 MarkupSafe-1.0 Werkzeug-0.14.1 click-6.7 itsdangerous-0.24
moving intermediate container 815304000b7
--> 6d82306079e
Step 7/8 : ENTRYPOINT ["python"]
--> Running in bdc1c83815e1
moving intermediate container bdc1c83815e1
--> 73cef38ac1c
Step 8/8 : CMD ["app.py"]
--> Running in a784d430dd8f
moving intermediate container a784d430dd8f
--> d866b83763a5
```

Create configuration files for Kubernetes

deployment.yaml

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: flask-node-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: flasknode
  template:
    metadata:
      labels:
        app: flasknode
    spec:
      containers:
        - name: flasknode
          image: registry.ng.bluemix.net/flask-node/app
          imagePullPolicy: Always
          ports:
            - containerPort: 5000
```

service.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: flask-node-deployment
spec:
  ports:
    - port: 5000
      targetPort: 5000
  selector:
    app: flasknode
```