

Sprint-4

Team ID	PNT2022TMID37472
Project Name	Smart Fashion Recommender Application
Batch No	B7-1A3E

Building Docker:

```
C:\Users\SUGAIEL FATHIMA>cd C:\Users\SUGAIEL FATHIMA\Downloads\sprint2-main\sprint2-main\
C:\Users\SUGAIEL FATHIMA\Downloads\sprint2-main\sprint2-main>docker build -t sprint2-main .
[+] Building 57.9s (12/12) FINISHED
=> [internal] load build definition from Dockerfile                                0.0s
=> => transferring dockerfile: 32B                                              0.0s
=> [internal] load .dockerignore                                                 0.0s
=> => transferring context: 2B                                                  0.0s
=> [internal] load metadata for docker.io/library/python:3.6                   4.3s
=> [auth] library/python:pull token for registry-1.docker.io                  0.0s
=> [internal] load build context                                                0.0s
=> => transferring context: 4.66kB                                             0.0s
=> [1/6] FROM docker.io/library/python:3.6@sha256:f8652afaf88c25f0d22354d547d892591067aa4026a7fa9a6819df9f300af6 0.0s
=> CACHED [2/6] WORKDIR /app                                                    0.0s
=> [3/6] ADD . /app                                                            0.1s
=> [4/6] COPY requirements.txt /app                                             0.0s
=> [5/6] RUN python3 -m pip install -r requirements.txt                        51.3s
=> [6/6] RUN python3 -m pip install ibm_db                                    1.0s
=> exporting to image                                                            1.0s
=> => exporting layers                                                            1.0s
=> => writing image sha256:6ce2c074559a79aac76bd066243b3005317495bab99e1674c5f06142f9efdc0d 0.0s
=> => naming to docker.io/library/sprint2-main                                0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

C:\Users\SUGAIEL FATHIMA\Downloads\sprint2-main\sprint2-main>docker images
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
sprint2-main        latest      6ce2c074559a  28 seconds ago 1.08GB
<none>              <none>      35762d3ac67a  11 hours ago  1.08GB
uifd/ui-for-docker  latest     965940f98fa5   6 years ago   8.1MB

C:\Users\SUGAIEL FATHIMA\Downloads\sprint2-main\sprint2-main>docker run -p 5000:5000 sprint2-main
* Serving Flask app 'app' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on all addresses.
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://172.17.0.2:5000/ (Press CTRL+C to quit)
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET /static/home.css HTTP/1.1" 200 -
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET /static/img/accessories1.png HTTP/1.1" 200 -
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET /static/img/men1.png HTTP/1.1" 200 -
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET /static/img/women2.png HTTP/1.1" 200 -
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET /static/img/accessories2.png HTTP/1.1" 200 -
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET /static/img/women1.png HTTP/1.1" 200 -
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET /static/img/men2.png HTTP/1.1" 200 -
172.17.0.1 - - [17/Nov/2022 14:23:43] "GET /static/img/home2.jpg HTTP/1.1" 200 -
```

Creating repository in Docker Hub:

The screenshot shows the Docker Hub repository page for `sugaiefathima / sp2-main`. The page includes a description section, Docker commands, tags and scans, automated builds, and a README section.

Repository: sugaiefathima / sp2-main

Description: This repository does not have a description. Last pushed: 8 minutes ago.

Docker commands: To push a new tag to this repository, use the command: `docker push sugaiefathima/sp2-main:tagname`.

Tags and scans: This repository contains 1 tag(s). The tag `latest` is of type `Image` and was pushed 8 minutes ago. Vulnerability scanning is disabled.

Automated Builds: Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating. Available with Pro, Team and Business subscriptions.

README: Repository description is empty. Click [here](#) to edit.

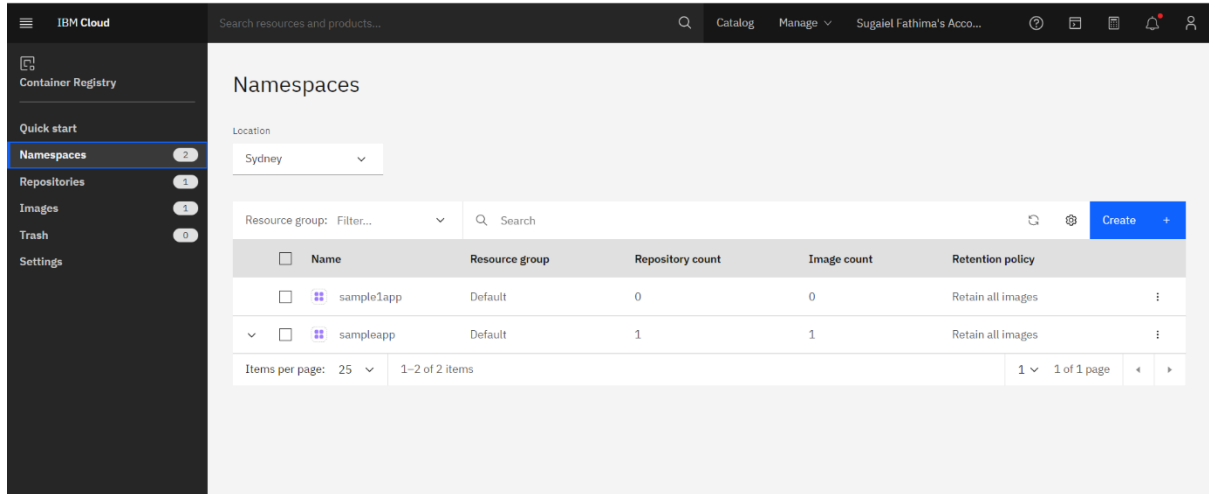
Pushing Docker into Container:

The screenshot shows the Docker Desktop interface with a list of running containers. The containers are:

NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
trusting_carson	au.icr.io/sampleapp/sampleapp:fashionapp	Running	5000:5000	2 minutes ago	Stop, Restart, Delete
k8s_POD_kube-controller-manager-docker-deskt	k8s.gcr.io/pause:3.8	Running		1 hour ago	Stop, Restart, Delete
k8s_POD_kube-apiserver-docker-desktop_kube-s	k8s.gcr.io/pause:3.8	Running		1 hour ago	Stop, Restart, Delete

Pushing Image to Container Registry:

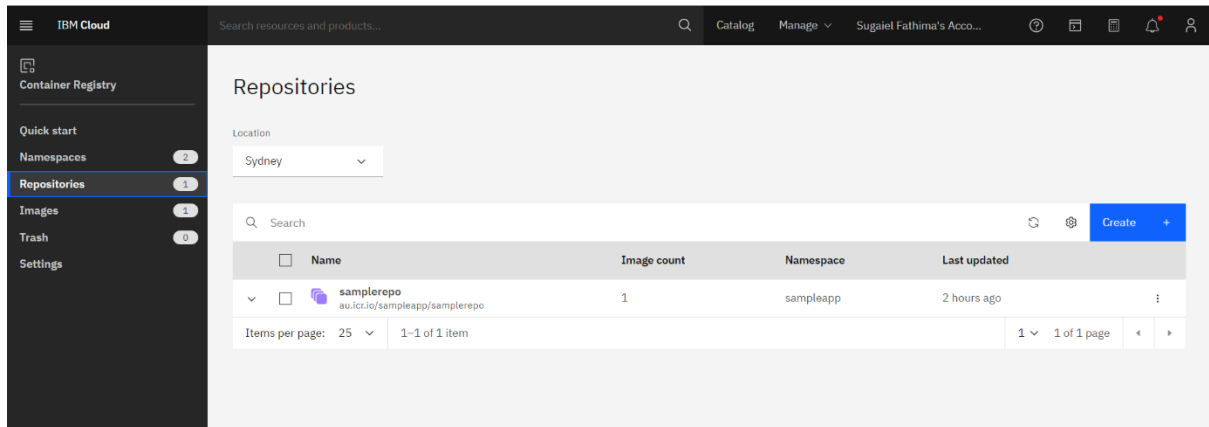
Creating Namespace:



The screenshot shows the IBM Cloud Container Registry interface. The left sidebar has a 'Namespaces' tab selected, showing 2 items. The main area is titled 'Namespaces' and shows a table of existing namespaces. The location is set to 'Sydney'. A 'Create' button is visible in the top right of the table area.

Name	Resource group	Repository count	Image count	Retention policy
sample1app	Default	0	0	Retain all images
sampleapp	Default	1	1	Retain all images

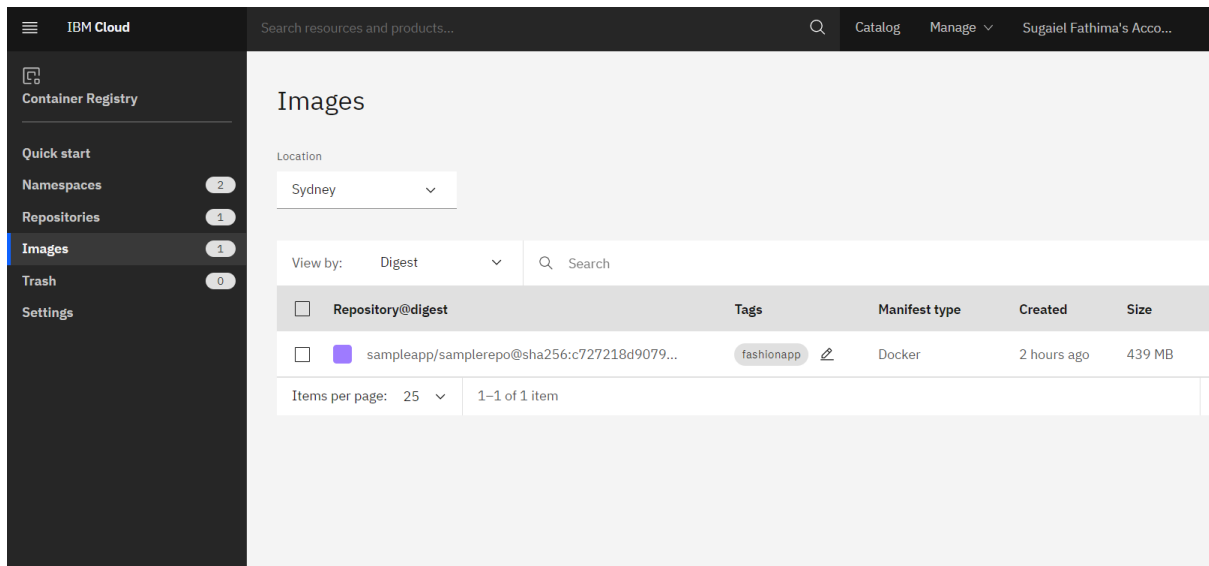
Creating Repository:



The screenshot shows the IBM Cloud Container Registry interface with the 'Repositories' tab selected. The left sidebar shows 1 repository. The main area is titled 'Repositories' and shows a table of existing repositories. The location is set to 'Sydney'. A 'Create' button is visible in the top right of the table area.

Name	Image count	Namespace	Last updated
samplerepo	1	sampleapp	2 hours ago

Creating Image:

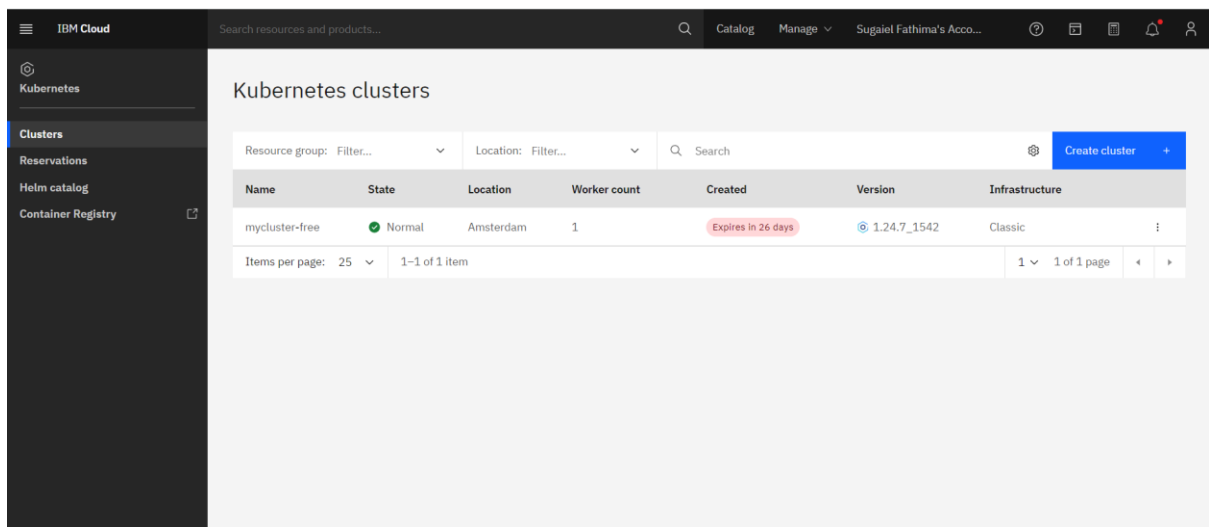


The screenshot shows the IBM Cloud Container Registry interface. The left sidebar contains navigation links: Container Registry, Quick start, Namespaces (2), Repositories (1), Images (1), Trash (0), and Settings. The main content area is titled 'Images' and shows a list of images. The location is set to 'Sydney'. The view is set to 'Digest'. A search bar is present. The table lists one image: 'sampleapp/samplerepo@sha256:c727218d9079...' with a tag 'fashionapp', manifest type 'Docker', created '2 hours ago', and size '439 MB'. The bottom of the table shows 'Items per page: 25' and '1-1 of 1 item'.

Repository@digest	Tags	Manifest type	Created	Size
sampleapp/samplerepo@sha256:c727218d9079...	fashionapp	Docker	2 hours ago	439 MB

Deploy in Kubernetes:

Creating Cluster:



The screenshot shows the IBM Cloud Kubernetes clusters interface. The left sidebar contains navigation links: Kubernetes, Clusters, Reservations, Helm catalog, and Container Registry. The main content area is titled 'Kubernetes clusters'. It features a table with columns: Name, State, Location, Worker count, Created, Version, and Infrastructure. A 'Create cluster' button is visible. The table lists one cluster: 'mycluster-free' with state 'Normal', location 'Amsterdam', worker count '1', and version '1.24.7_1542'. The 'Created' column shows 'Expires in 26 days'. The bottom of the table shows 'Items per page: 25' and '1-1 of 1 item'.

Name	State	Location	Worker count	Created	Version	Infrastructure
mycluster-free	Normal	Amsterdam	1	Expires in 26 days	1.24.7_1542	Classic

Deployment:

```
Command Prompt
C:\Users\SUGAIEL FATHIMA>kubectl
kubectl controls the Kubernetes cluster manager.

Find more information at: https://kubernetes.io/docs/reference/kubectl/

Basic Commands (Beginner):
  create      Create a resource from a file or from stdin
  expose      Take a replication controller, service, deployment or pod and expose it as a new Kubernetes service
  run         Run a particular image on the cluster
  set         Set specific features on objects

Basic Commands (Intermediate):
  explain     Get documentation for a resource
  get         Display one or many resources
  edit        Edit a resource on the server
  delete      Delete resources by file names, stdin, resources and names, or by resources and label selector

Deploy Commands:
  rollout     Manage the rollout of a resource
  scale       Set a new size for a deployment, replica set, or replication controller
  autoscale   Auto-scale a deployment, replica set, stateful set, or replication controller

Cluster Management Commands:
  certificate Modify certificate resources.
  cluster-info Display cluster information
  top         Display resource (CPU/memory) usage
  cordon      Mark node as unschedulable
  uncordon    Mark node as schedulable
  drain       Drain node in preparation for maintenance
  taint       Update the taints on one or more nodes

Troubleshooting and Debugging Commands:
  describe    Show details of a specific resource or group of resources
  logs        Print the logs for a container in a pod
  attach      Attach to a running container
  exec        Execute a command in a container
  port-forward Forward one or more local ports to a pod
  proxy       Run a proxy to the Kubernetes API server
  cp          Copy files and directories to and from containers
  auth        Inspect authorization
  debug       Create debugging sessions for troubleshooting workloads and nodes

Advanced Commands:
  diff        Diff the live version against a would-be applied version
  apply       Apply a configuration to a resource by file name or stdin
  patch       Update fields of a resource
  replace     Replace a resource by file name or stdin
  wait        Experimental: Wait for a specific condition on one or many resources
  kustomize   Build a kustomization target from a directory or URL.
```