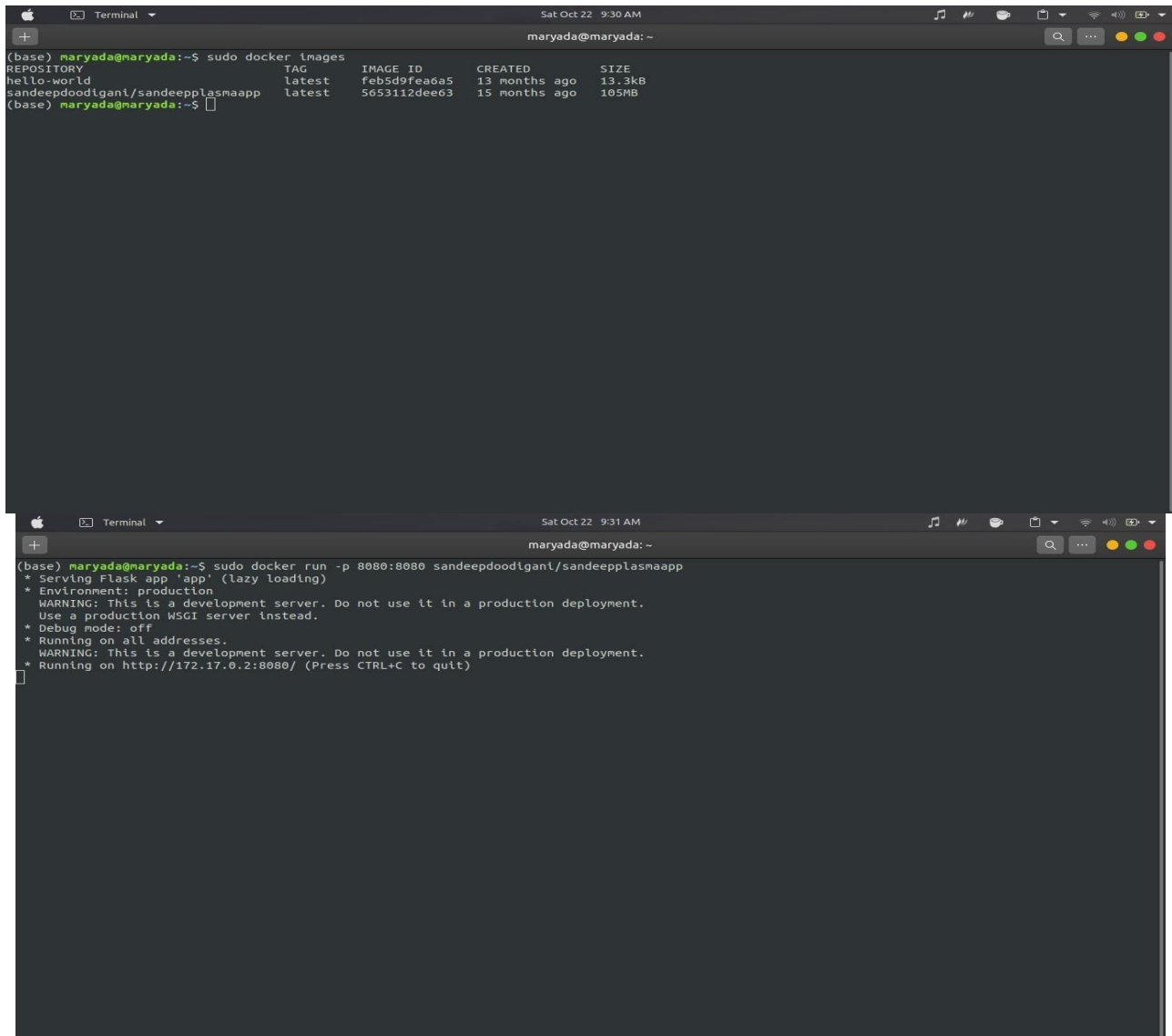


Assignment-4

1. Pull an Image from docker hub and run it in docker .



The image shows two terminal windows from a macOS system. The top window displays the output of the `sudo docker images` command, listing three Docker images: `hello-world`, `sandeepdoodigani/sandeepplasmaapp`, and `sandeepdoodigani/sandeepplasmaapp`. The bottom window shows the output of the `sudo docker run -p 8080:8080 sandeepdoodigani/sandeepplasmaapp` command, which starts a Flask application on port 8080. The output includes warnings about the development server and the production WSGI server.

```
(base) maryada@maryada:~$ sudo docker images
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
hello-world          latest      feb5d9fea6a5  13 months ago  13.3kB
sandeepdoodigani/sandeepplasmaapp latest      5653112dee63  15 months ago  105MB
(base) maryada@maryada:~$
```

```
(base) maryada@maryada:~$ sudo docker run -p 8080:8080 sandeepdoodigani/sandeepplasmaapp
* 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://172.17.0.2:8080/ (Press CTRL+C to quit)
```

2. Create a docker file for the jobportal application and deploy it in Docker desktop application.

Dockerfile:

FROM python:3.6

WORKDIR /app

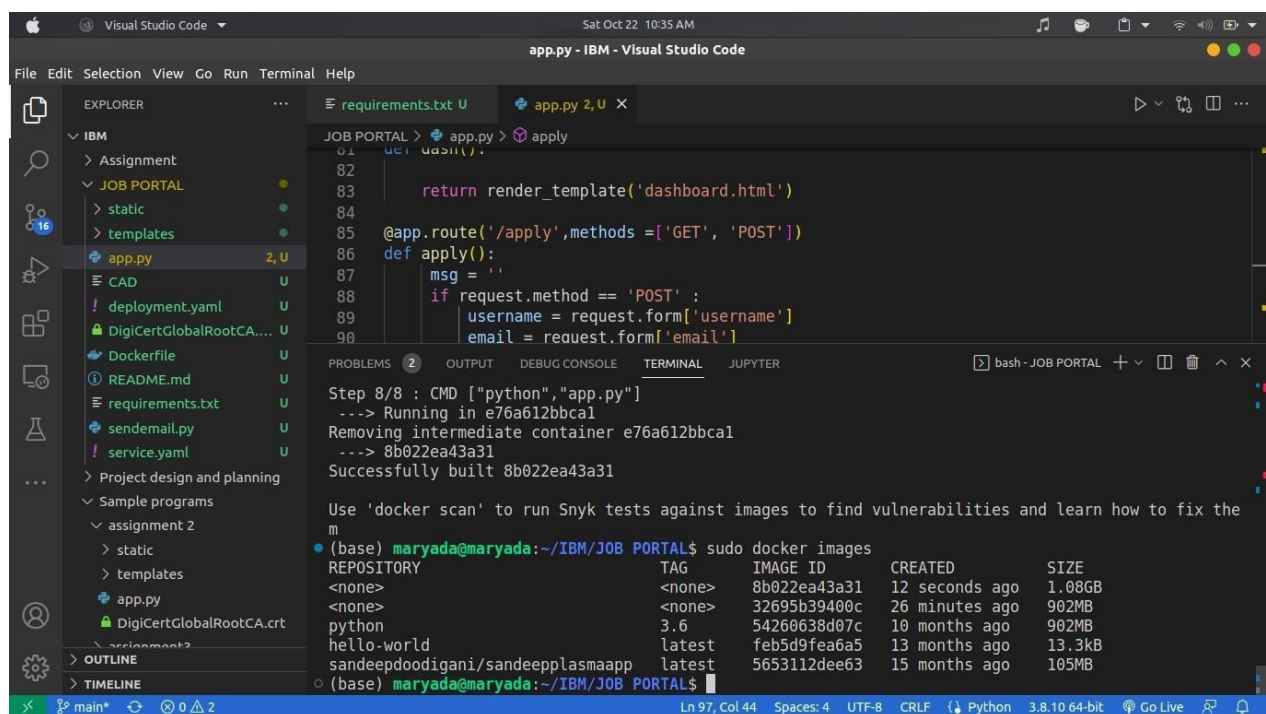
ADD . /app

COPY requirements.txt /app

RUN python3 -m pip install -r requirements.txt

RUN python3 -m pip install ibm_db

EXPOSE 5000



The screenshot shows the Visual Studio Code interface with a project named 'JOB PORTAL'. The Explorer sidebar on the left shows the file structure: IBM, Assignment, JOB PORTAL (containing static, templates, app.py, deployment.yaml, DigiCertGlobalRootCA..., Dockerfile, README.md, requirements.txt, sendemail.py, and service.yaml), and Sample programs. The main editor displays the 'requirements.txt' file with the following content:

```
ibm_db
Flask
Flask-Mail
Flask-Security
Flask-Session
Flask-User
Flask-WebSession
Flask-Whoosh
Flask-Whoosh-Search
Flask-Whoosh-Search-Index
Flask-Whoosh-Search-Index-Test
Flask-Whoosh-Search-Index-Test-Test
```

The 'app.py' file is also visible in the editor, showing a Flask application with a route for '/apply'.

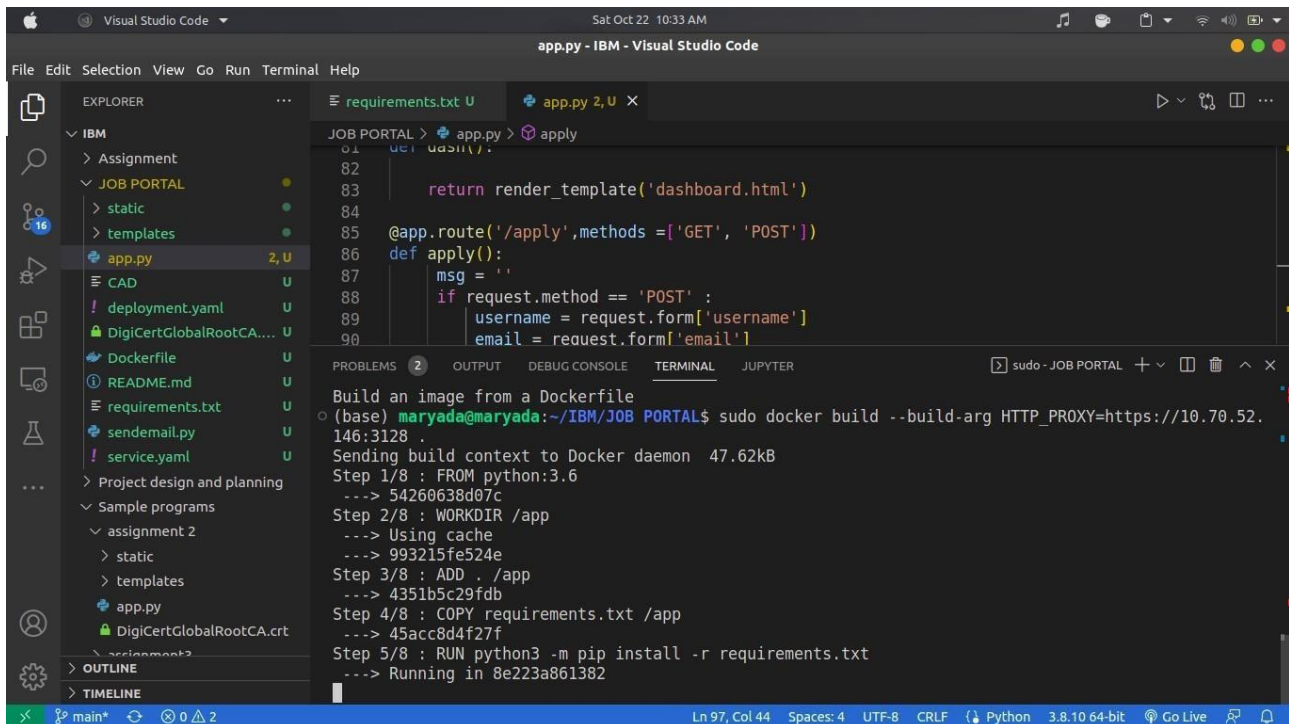
The terminal at the bottom shows the command 'python3 app.py' being executed, which results in a successful build of the Docker image. The terminal output includes the following steps:

```
Step 8/8 : CMD ["python","app.py"]
--> Running in e76a612bbca1
Removing intermediate container e76a612bbca1
--> 8b022ea43a31
Successfully built 8b022ea43a31
```

The terminal also shows the command 'sudo docker images' and the resulting output:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
<none>	<none>	8b022ea43a31	12 seconds ago	1.08GB
<none>	<none>	32695b39400c	26 minutes ago	902MB
python	3.6	54260638d07c	10 months ago	902MB
hello-world	latest	feb5d9fea6a5	13 months ago	13.3kB
sandeepdoodigani/sandeepplasmaapp	latest	5653112dee63	15 months ago	105MB

CMD ["python","app.py"]

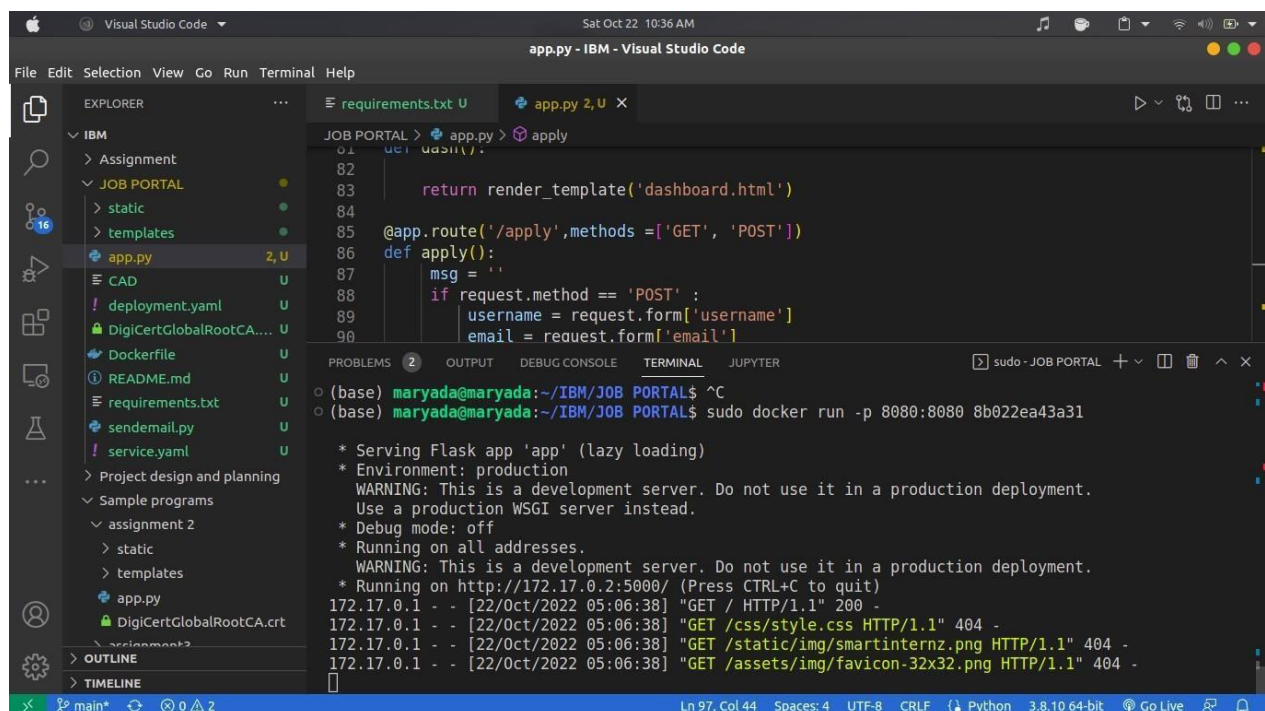


The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left displaying the project structure. The main editor shows the `app.py` file with the following code:

```
01 def dash():
82
83     return render_template('dashboard.html')
84
85 @app.route('/apply', methods = ['GET', 'POST'])
86 def apply():
87     msg = ''
88     if request.method == 'POST':
89         username = request.form['username']
90         email = request.form['email']
```

The TERMINAL panel at the bottom shows the output of the `sudo docker build` command:

```
Build an image from a Dockerfile
(base) maryada@maryada:~/IBM/JOB PORTAL$ sudo docker build --build-arg HTTP_PROXY=https://10.70.52.146:3128 .
Sending build context to Docker daemon 47.62kB
Step 1/8 : FROM python:3.6
--> 54260638d07c
Step 2/8 : WORKDIR /app
--> Using cache
--> 993215fe524e
Step 3/8 : ADD . /app
--> 4351b5c29fdb
Step 4/8 : COPY requirements.txt /app
--> 45acc8d4f27f
Step 5/8 : RUN python3 -m pip install -r requirements.txt
--> Running in 8e223a861382
```



The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left displaying the project structure. The main editor shows the `app.py` file with the following code:

```
01 def dash():
82
83     return render_template('dashboard.html')
84
85 @app.route('/apply', methods = ['GET', 'POST'])
86 def apply():
87     msg = ''
88     if request.method == 'POST':
89         username = request.form['username']
90         email = request.form['email']
```

The TERMINAL panel at the bottom shows the output of the `sudo docker run` command:

```
(base) maryada@maryada:~/IBM/JOB PORTAL$ ^C
(base) maryada@maryada:~/IBM/JOB PORTAL$ sudo docker run -p 8080:8080 8b022ea43a31

* 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 - - [22/Oct/2022 05:06:38] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [22/Oct/2022 05:06:38] "GET /css/style.css HTTP/1.1" 404 -
172.17.0.1 - - [22/Oct/2022 05:06:38] "GET /static/img/smartinternz.png HTTP/1.1" 404 -
172.17.0.1 - - [22/Oct/2022 05:06:38] "GET /assets/img/favicon-32x32.png HTTP/1.1" 404 -
```

Aboutus

Mission

SMARTBRIDGE is an edTech organization with a vision to bridge the gap between academia & industry. Our outcome-based experiential learning programs on emerging technologies (Internet of Things, Machine Learning, Data Science, Artificial Intelligence, Robotics) are building skilled entry-level engineers, for the corporate world..

Vission

Our main objective is to bridge the existing gaps between prevailing industry standards and what the academics offer to the graduates while passing out of university. SmartBridge offers suitable skill deployment and training to the young talent before on boarding their first job. Our skill development programs are designed considering the present expectations in the industry.

Objective

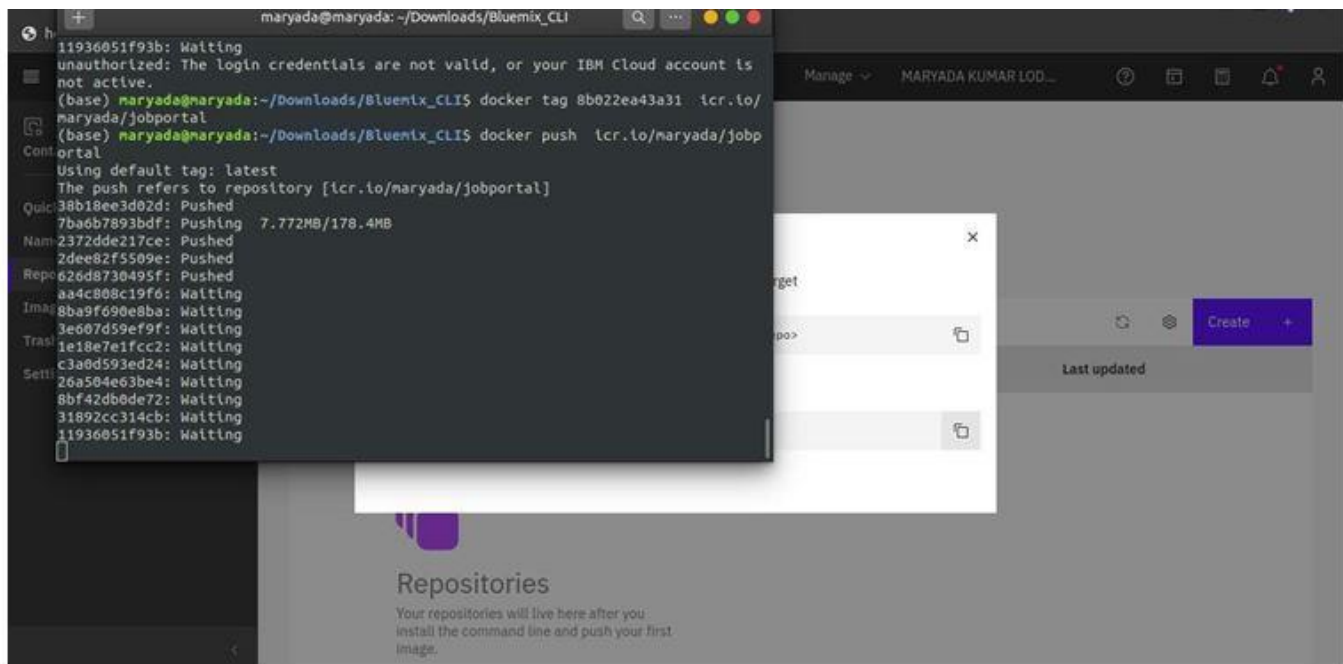
Well directed career guidance programs for educational institutions
Appropriate certification courses that suit the industry need
Train the trainers; expanded awareness about the current industry standards
Liaise with corporates to offer niche internships
Establish technology development centers in colleges
Specialised incubation centers in collaboration with corporates

JobPortal

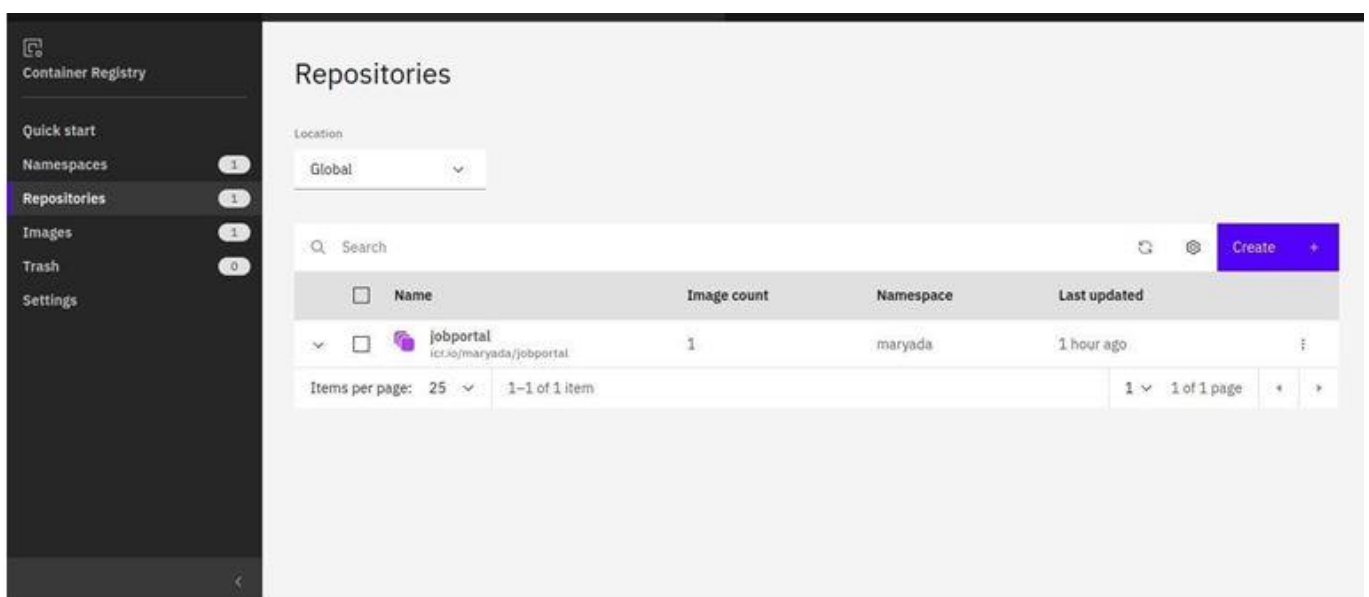
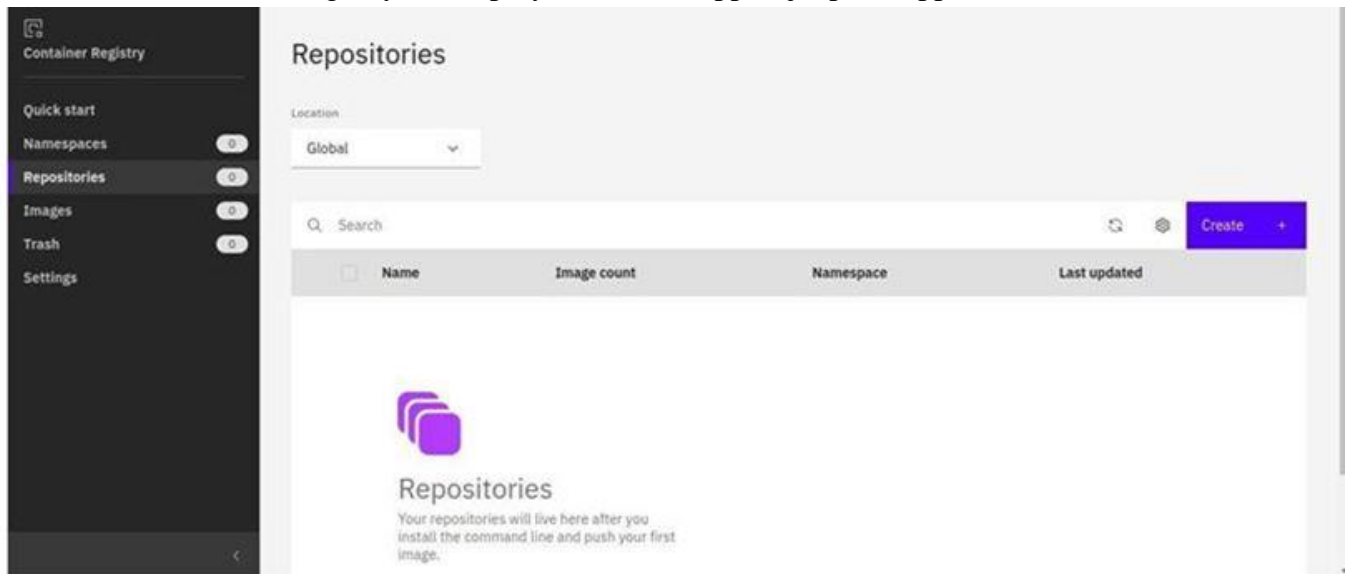
Lorem ipsum dolor sit amet consectetur adipiscing elit. Voluptatum quis, reiciendis id magni magnam, accusamus nobis in, temporibus molestias ab placeat rerum aperiam illum perspiciatis ducimus non! Fugiat, odit ducimus.

Get in Touch

- jobportal@gmail.com
- [+91 8977787657](tel:+918977787657)



3. Create a IBM container registry and deploy helloworld app or jobportalapp.



4. Create a Kubernetes cluster in IBM cloud and deploy helloworld image or jobportal image and also expose the same app to run in nodeport.

