

# Motivation

## What are Containers?

## What is Docker?

---

## How is Docker Used?

## Demonstration

## Overview

---

- Docker Playground -
- Create a Docker Hub Account -
  - Copy Demo Files -
- Build and Publish a Docker Image -
  - Pull and Run a Docker Image -

# Docker Playground

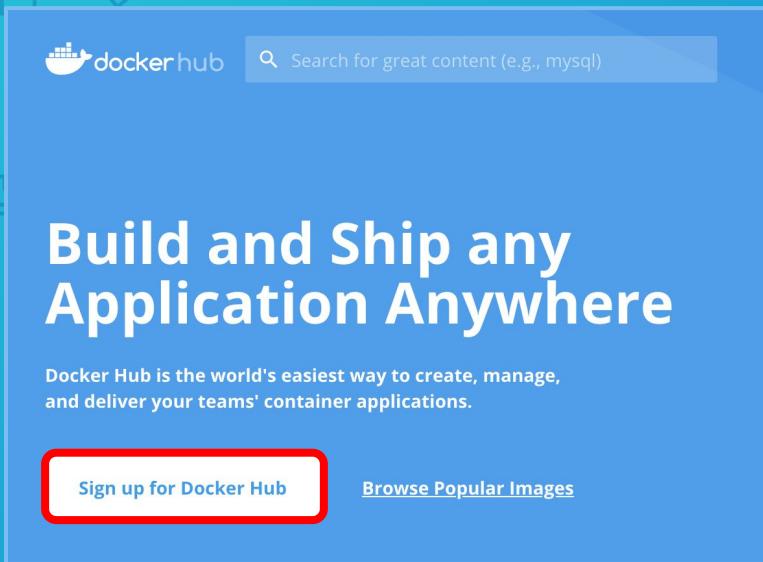
The screenshot shows the Katacoda Docker Playground interface. At the top, there's a navigation bar with the O'Reilly Katacoda logo, links for "SCENARIO AUTHORING INFORMATION" and "KATACODA OVERVIEW & SOLUTIONS", and a search bar. Below the header, a teal bar contains the title "Docker Playground". To the right of the teal bar is a "Terminal" section with a dark background, a dollar sign prompt (\$), and a plus sign (+) for starting new terminals. The main content area is titled "Helpful Links" in large blue text. It includes instructions for starting Docker with "docker run" and launching containers with "docker run redis". It also mentions using **CTRL+C** to stop running containers.

Start using Docker with `docker run`.

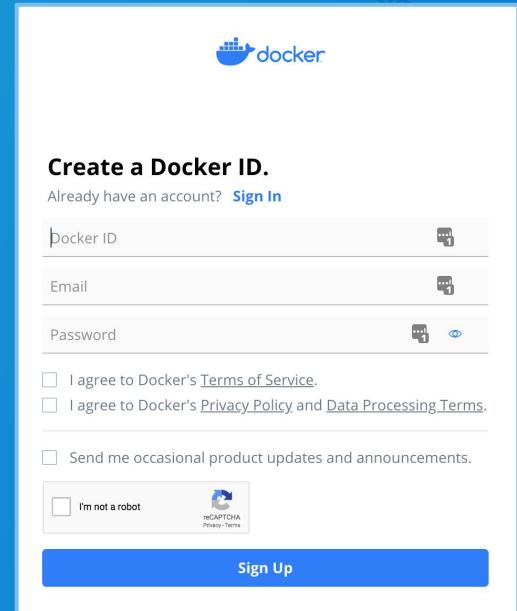
Launch containers with `docker run redis`. Use **CTRL+C** to stop the running container.

[katacoda.com/courses/docker/playground](https://katacoda.com/courses/docker/playground)

## Create a Docker Hub Account



The screenshot shows the Docker Hub homepage with a light blue background. At the top left is the Docker Hub logo. A search bar at the top right contains the placeholder text "Search for great content (e.g., mysql)". Below the search bar, a large white text area displays the headline "Build and Ship any Application Anywhere". Underneath this, a smaller text block reads: "Docker Hub is the world's easiest way to create, manage, and deliver your teams' container applications." At the bottom of the page are two buttons: "Sign up for Docker Hub" (which is highlighted with a red border) and "Browse Popular Images".



The screenshot shows a sign-up form titled "Create a Docker ID." It includes fields for "Docker ID", "Email", and "Password", each with a corresponding icon. Below these fields are three checkboxes: "I agree to Docker's Terms of Service.", "I agree to Docker's Privacy Policy and Data Processing Terms.", and "Send me occasional product updates and announcements.". At the bottom of the form are two buttons: "I'm not a robot" with a reCAPTCHA checkbox, and a large blue "Sign Up" button.

[hub.docker.com](https://hub.docker.com)

## Copy Demo Files

Terminal Host 1 +

```
master $ curl -O https://raw.githubusercontent.com/appian/hackathons/master/workshops/kubernetes/server.py
```

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
	Dload	Upload	Total	Spent	Left	Speed	
100	535	100	535	0	0	1280	0 --:--:-- --:--:-- --:--:-- 1279

```
master $ ls
```

```
go server.py
```

```
master $ curl -O https://raw.githubusercontent.com/appian/hackathons/master/workshops/kubernetes/Dockerfile
```

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
	Dload	Upload	Total	Spent	Left	Speed	
100	134	100	134	0	0	200	0 --:--:-- --:--:-- --:--:-- 200

```
master $ curl -O https://raw.githubusercontent.com/appian/hackathons/master/workshops/kubernetes/deployment.yaml
```

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
	Dload	Upload	Total	Spent	Left	Speed	
100	414	100	414	0	0	557	0 --:--:-- --:--:-- --:--:-- 567

App.py, fizzbuzz.py, requirements.txt, Dockerfile

# Build and Publish a Docker Image

---

## fizzbuzz.py

```
def fizzbuzz(max_int):
    ans = str()
    for i in range(max_int+1):
        if i % 3 == 0 and i % 5 == 0:
            ans = ans + "fizzbuzz"
            continue
        elif i % 3 == 0:
            ans = ans + "fizz"
            continue
        elif i % 5 == 0:
            ans = ans + "buzz"
            continue
        ans = ans + str(i)
    return ans
```

# Build and Publish a Docker Image

app.py

```
from flask import Flask
from fizzbuzz import fizzbuzz

app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'

@app.route('/fizzbuzz/<int:max_int>')
def get_fizzbuzz(max_int):
    return fizzbuzz(max_int)

if __name__ == '__main__':
    app.run(debug=True, host='0.0.0.0', port=8080)
```

## Dockerfile

```
FROM python:3.7-slim

# Adding application code to the Docker Image
COPY ./app.py /app.py
COPY ./fizzbuzz.py /fizzbuzz.py

# Updating pip
RUN pip3 install --upgrade pip

# Installing python dependencies
COPY requirements.txt requirements.txt
RUN pip3 install -r requirements.txt

# Running as a non-root USER
RUN adduser --uid 1000 --system user
USER 1000

# Exposing port 8080 to network traffic
EXPOSE 8080

ENTRYPOINT ["python", "app.py"]
```

## Build and Publish a Docker Image

---

```
docker build .
```

```
docker login --username <username>
```

```
docker tag <SHA> <username>/fizzbuzz-demo:<TAG>
```

```
docker push <username>/fizzbuzz-demo:<TAG>
```

## Pull and Run a Docker Image

---

```
docker pull 724ephil/fizzbuzz-demo:1.2
```

```
docker run -d -p 80:8080 724ephil/fizzbuzz-demo:1.2
```

# Pull and Run a Docker Image

Terminal +

Your Interactive Bash Terminal.  
A good starting point is executing `docker`  
\$  
\$ docker pull 724ephil/fizzbuzz-demo:1.2  
1.2: Pulling from 724ephil/fizzbuzz-demo



Terminal

Your Interactive Bash Terminal.

A good starting point is executing `docker`

\$

\$ docker pull 724ephil/fizzbuzz-demo:1.2

1.2: Pulling from 724ephil/fizzbuzz-demo

Open New Terminal

View HTTP port 80 on Host 1

Select port to view on Host 1

View HTTP port 80 on Client 1

Select port to view on Client 1

68bc1c23d27a: Pull complete

660a0e70323c: Pull complete



Docker Playground | Katacoda

https://2886795307-80-frugo01.environments.katacoda.com/fizzbuzz/90

fizzbuzz12fizz4buzzfizz78fizzbuzz11fizz1314fizzbuzz1617fizz19buzzfizz2223fizzbuzz26fizz2



# Thank You! Any Questions?

