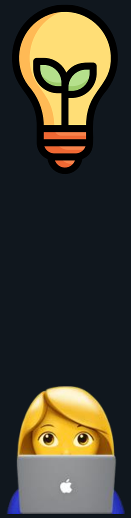
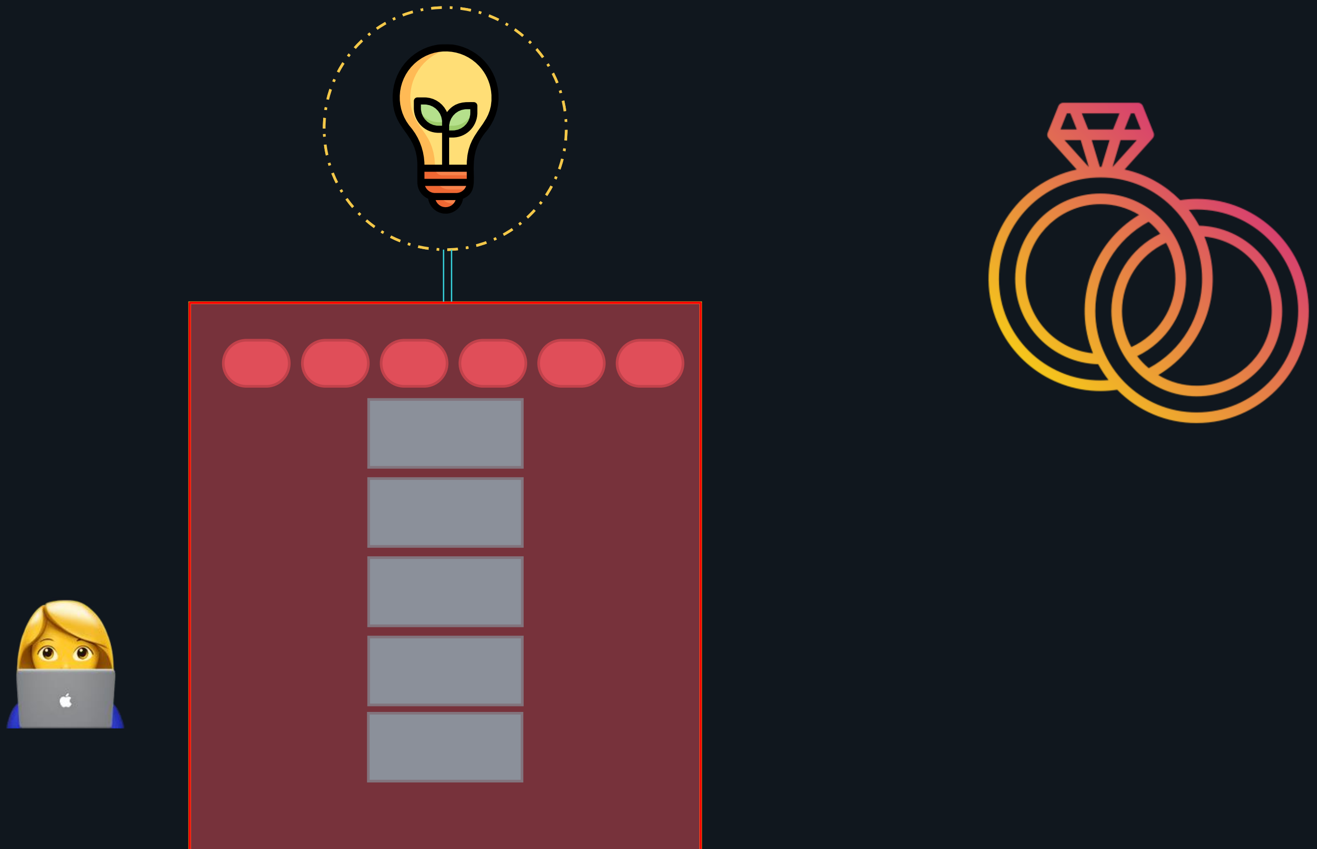
KodeKloud

KodeKloud

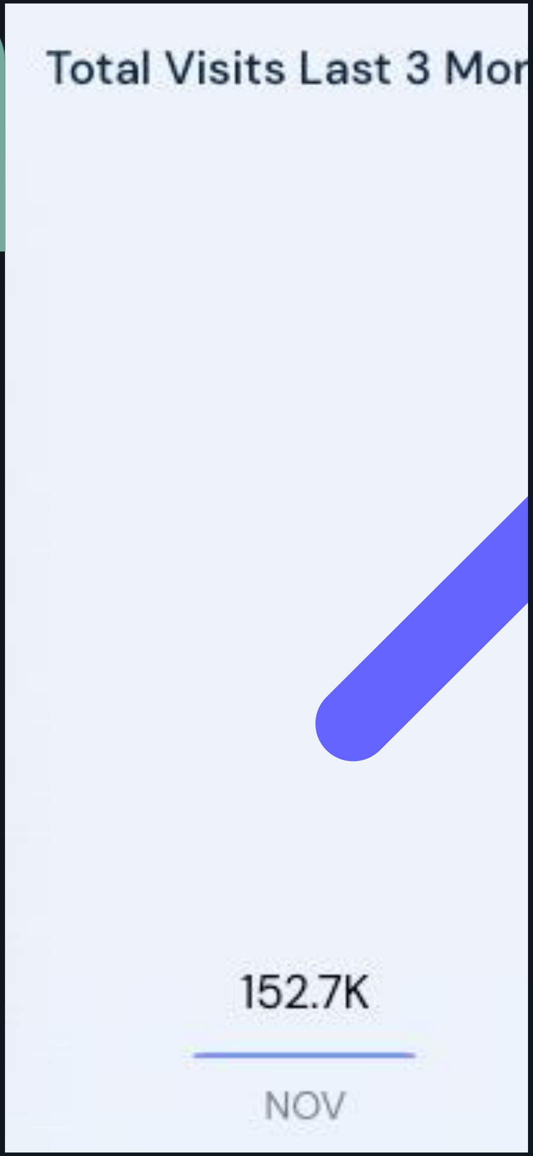
12 Factor App

KodeKloud





Fast forward to today!


Total Visits Last 3 Mor
152.7K
NOV

DEMO – Cloud Provisioning

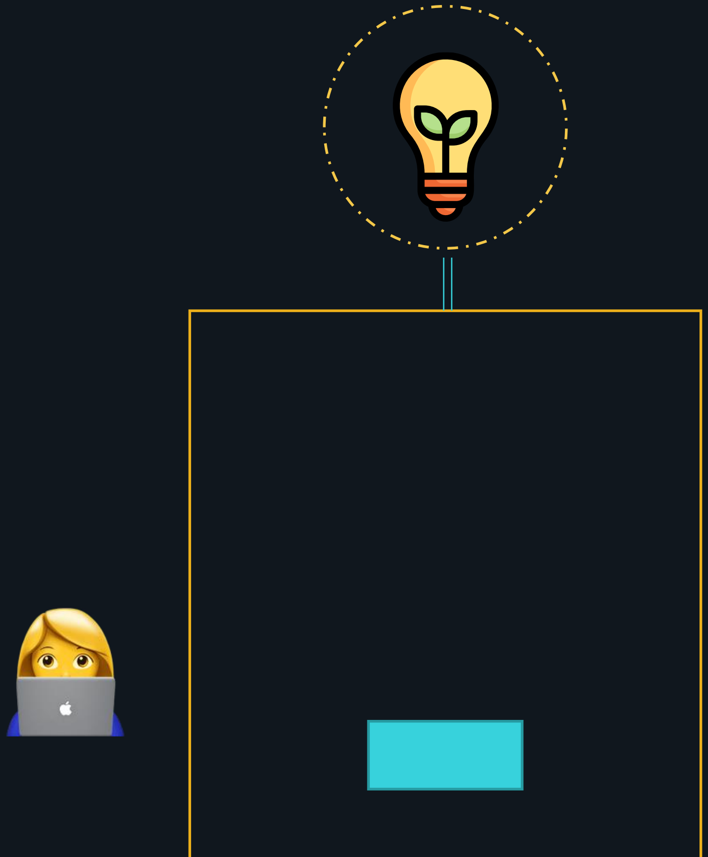
KodeKloud

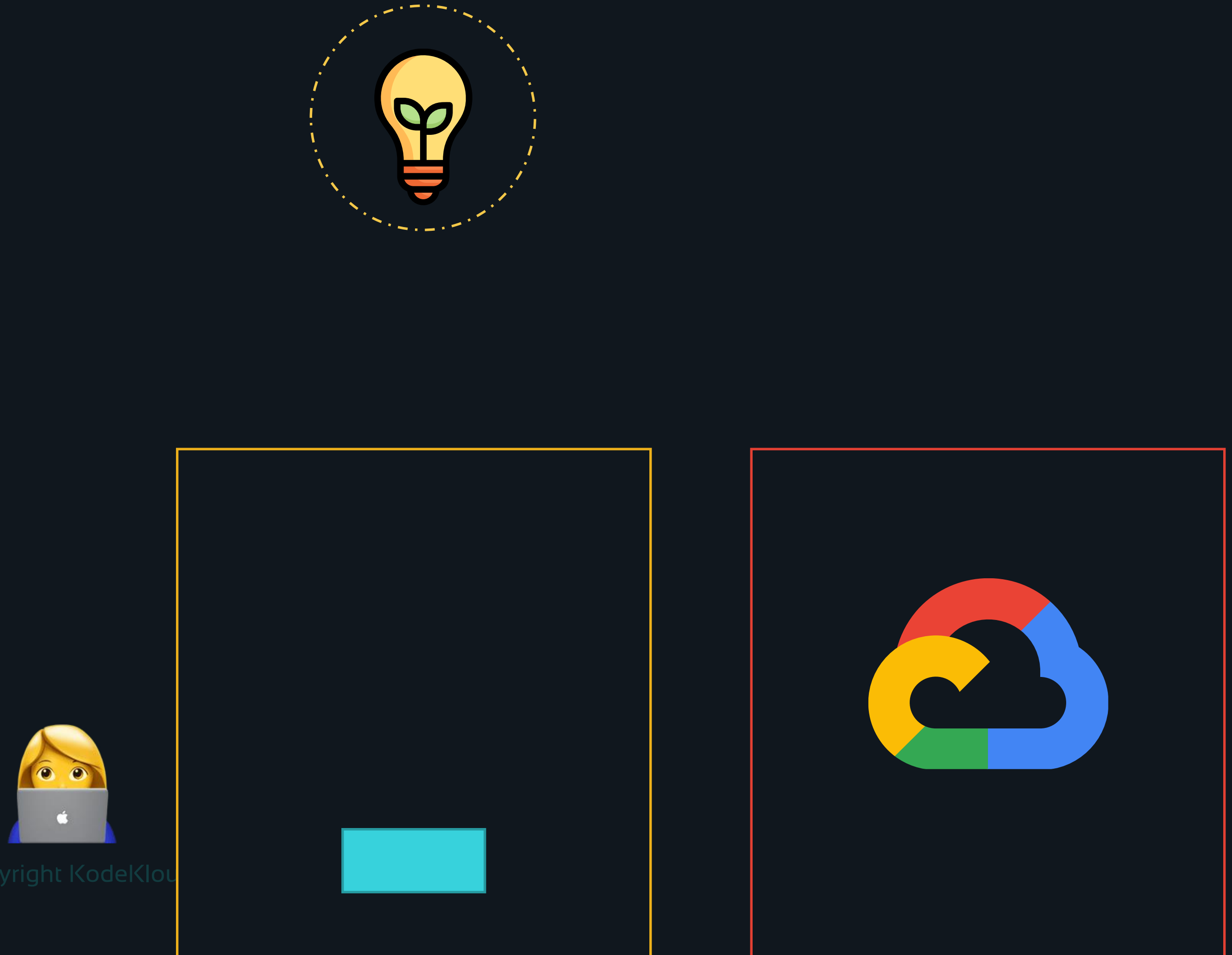
SERVERLESS!!!

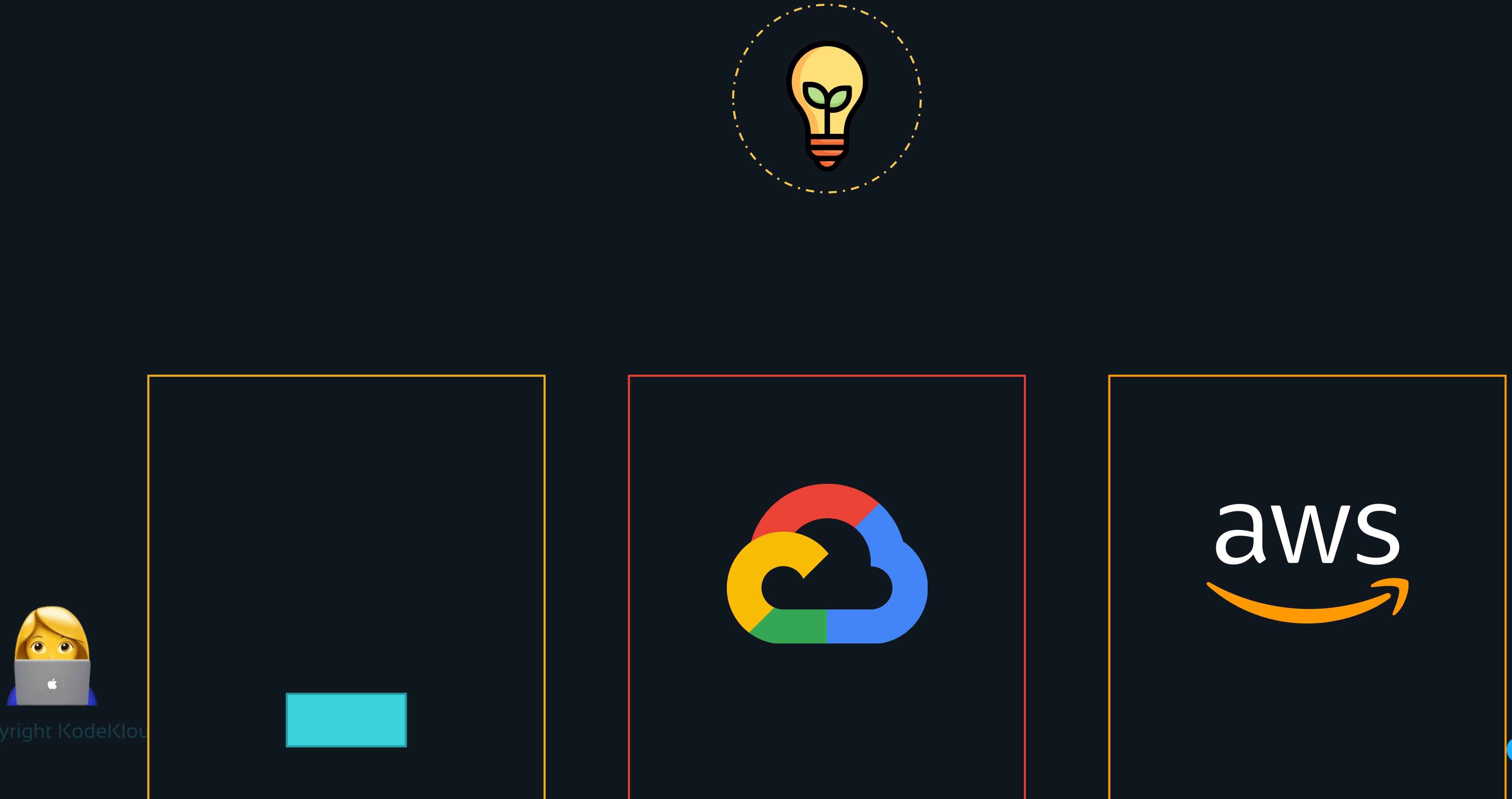
99.999%

99.999%

- NO DOWNTIME
- NO MAINTENANCE OUTAGES



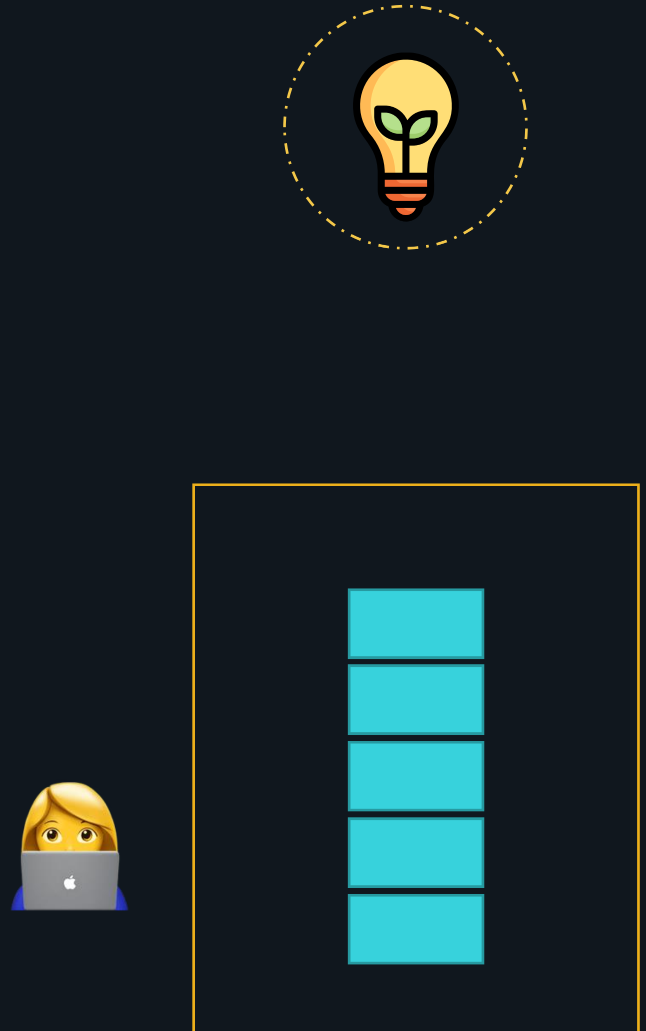



aws

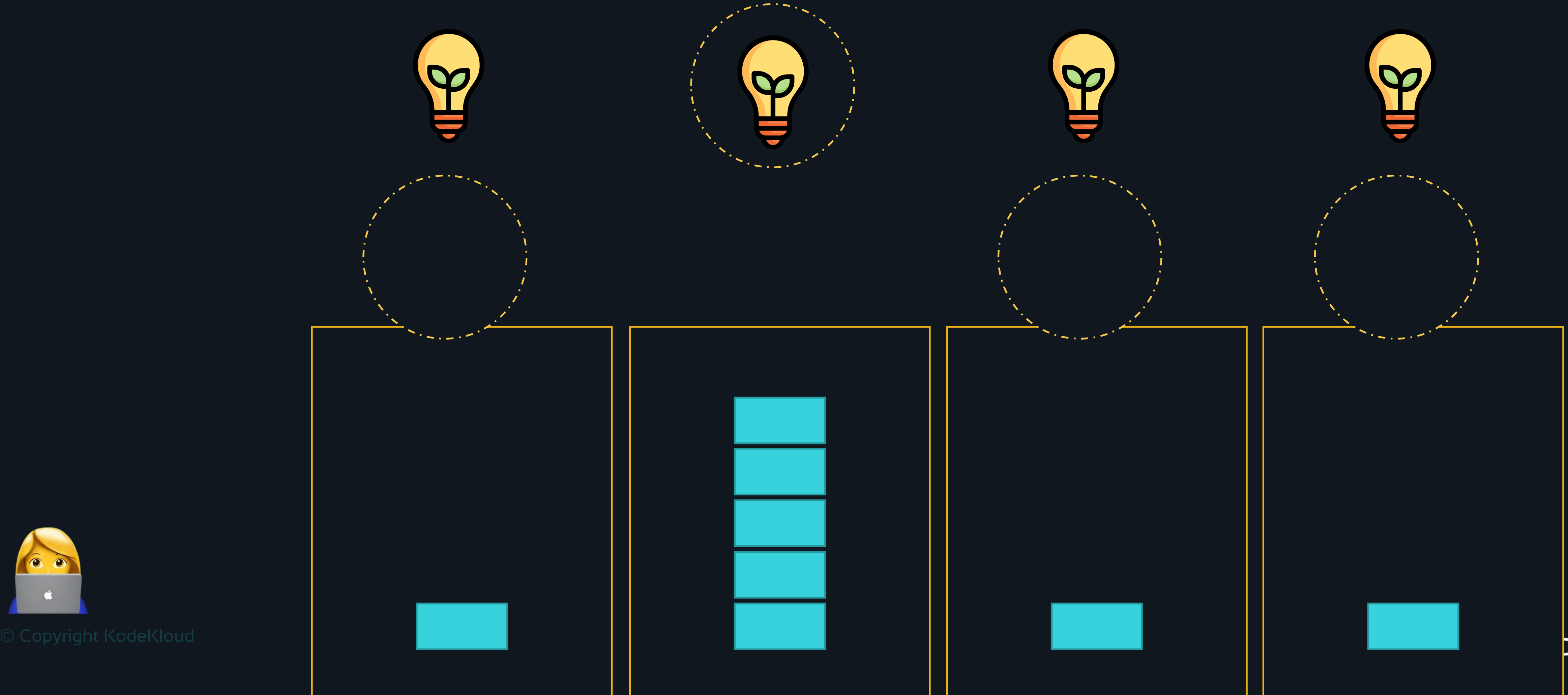
aws

Portability

Vertical Scaling



Horizontal Scaling



Portability

Continuous Deployment

Scalability

Modern Cloud Platforms

The Twelve-Factor App

https://12factor.net/

KodeKloud

The Twelve-Factor App

I
Codebase

II
Dependencies

III
Config

IV
Backing Services

V
Build, release, run

VI
Processes

VII
Port Binding

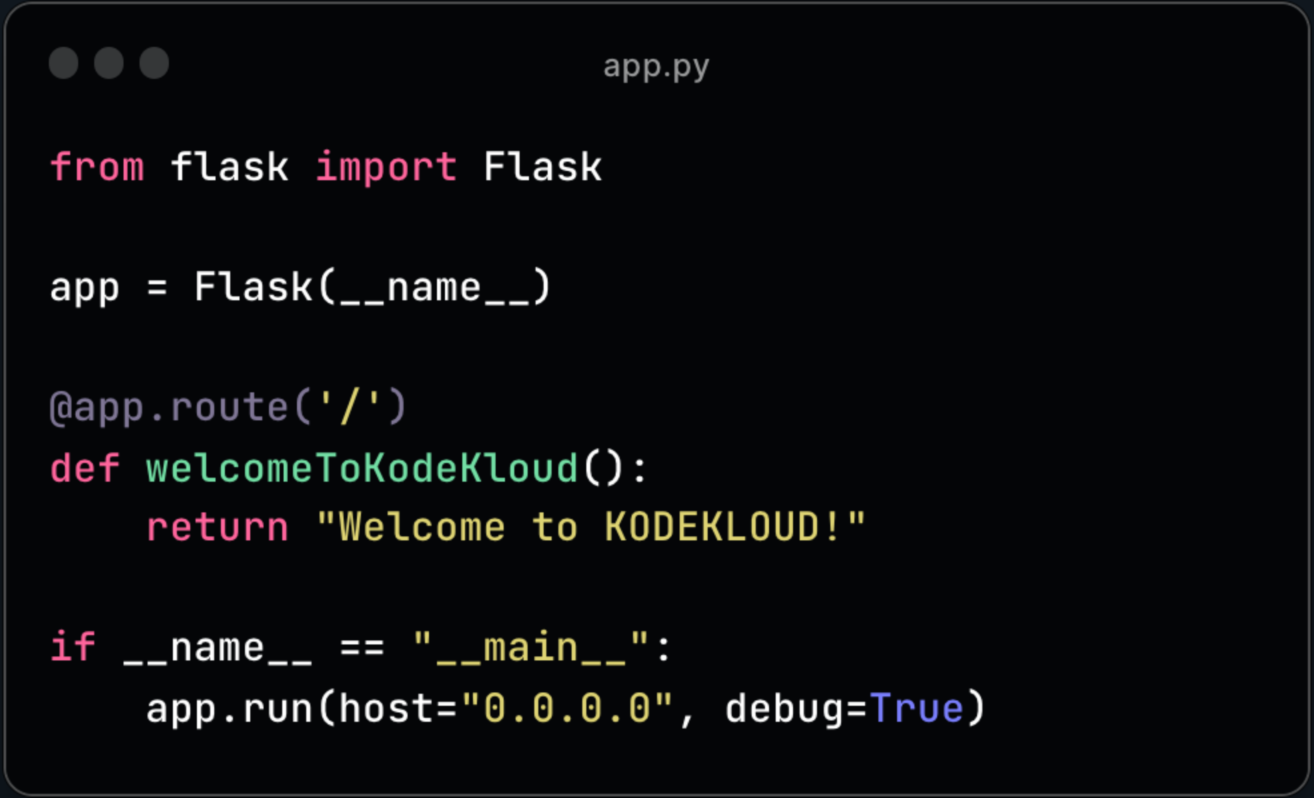
VIII
Concurrency

IX
Disposability

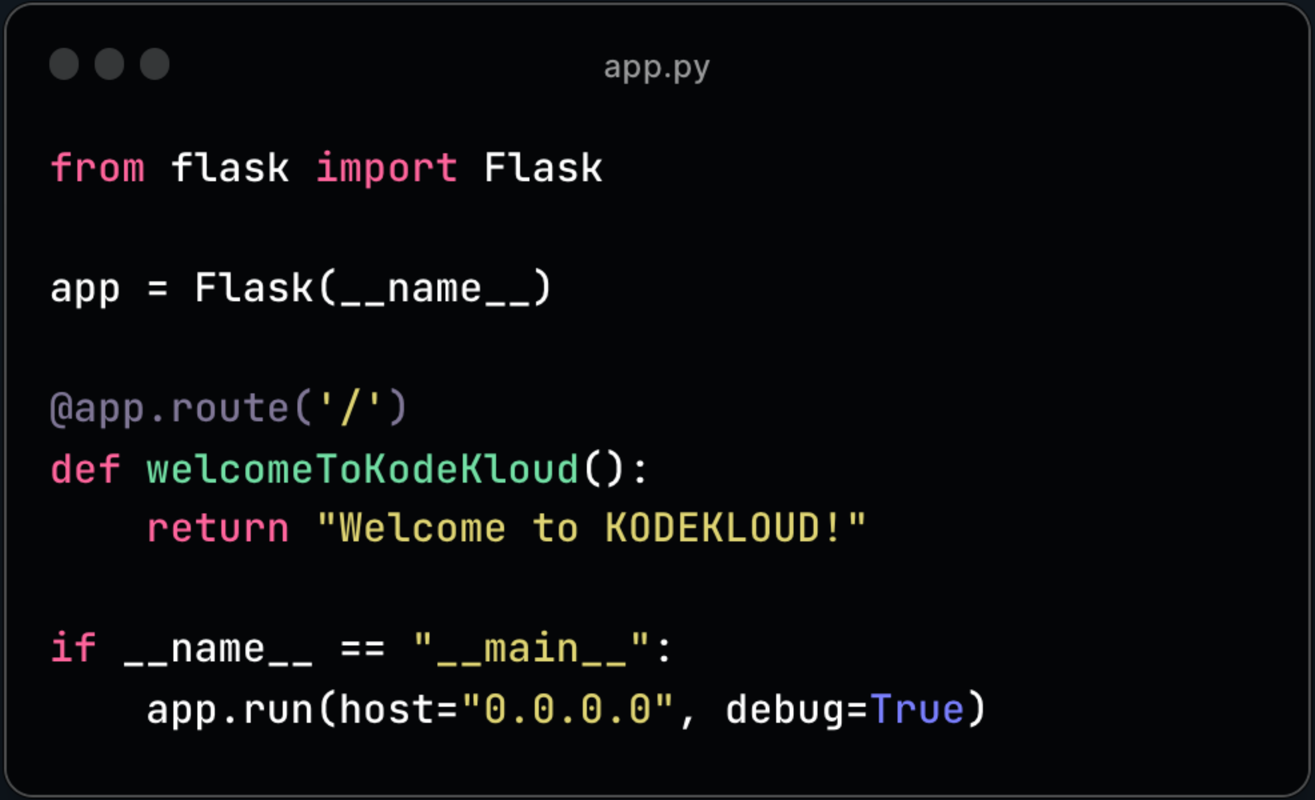
X
Dev/prod parity

XI
Logs

XII
Admin Processes

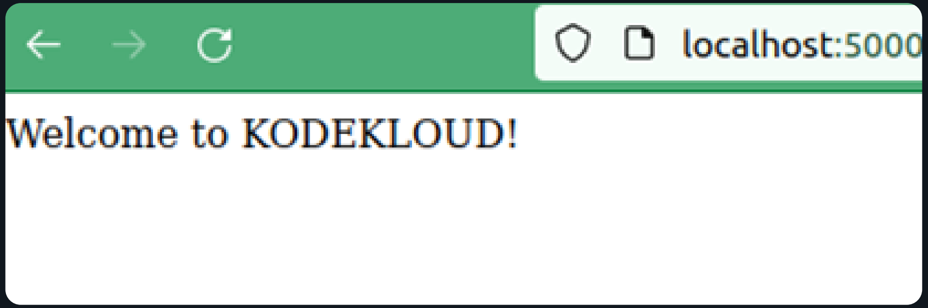


```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def welcomeToKodeKloud():
    return "Welcome to KODEKLOUD!"

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

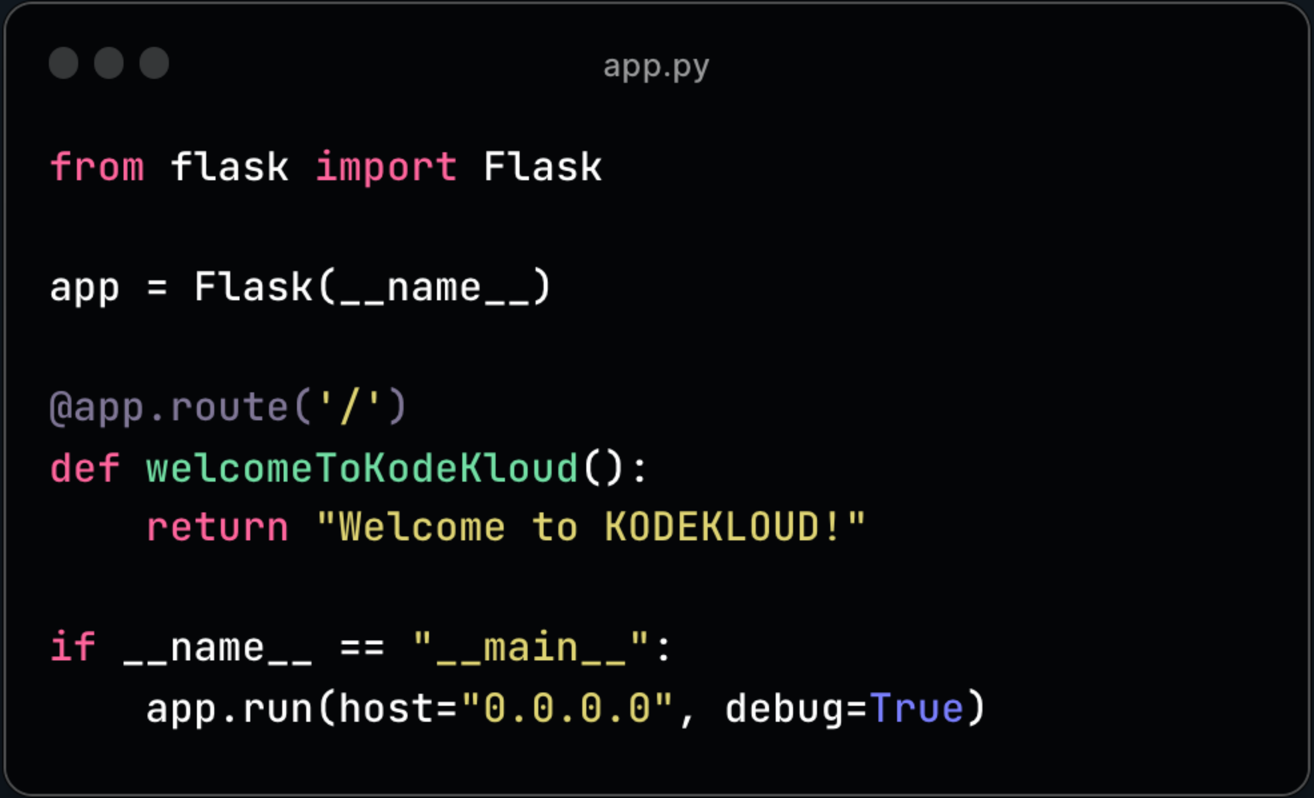


app.py

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def welcomeToKodeKloud():
    return "Welcome to KODEKLOUD!"

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



Welcome to KODEKLOUD!



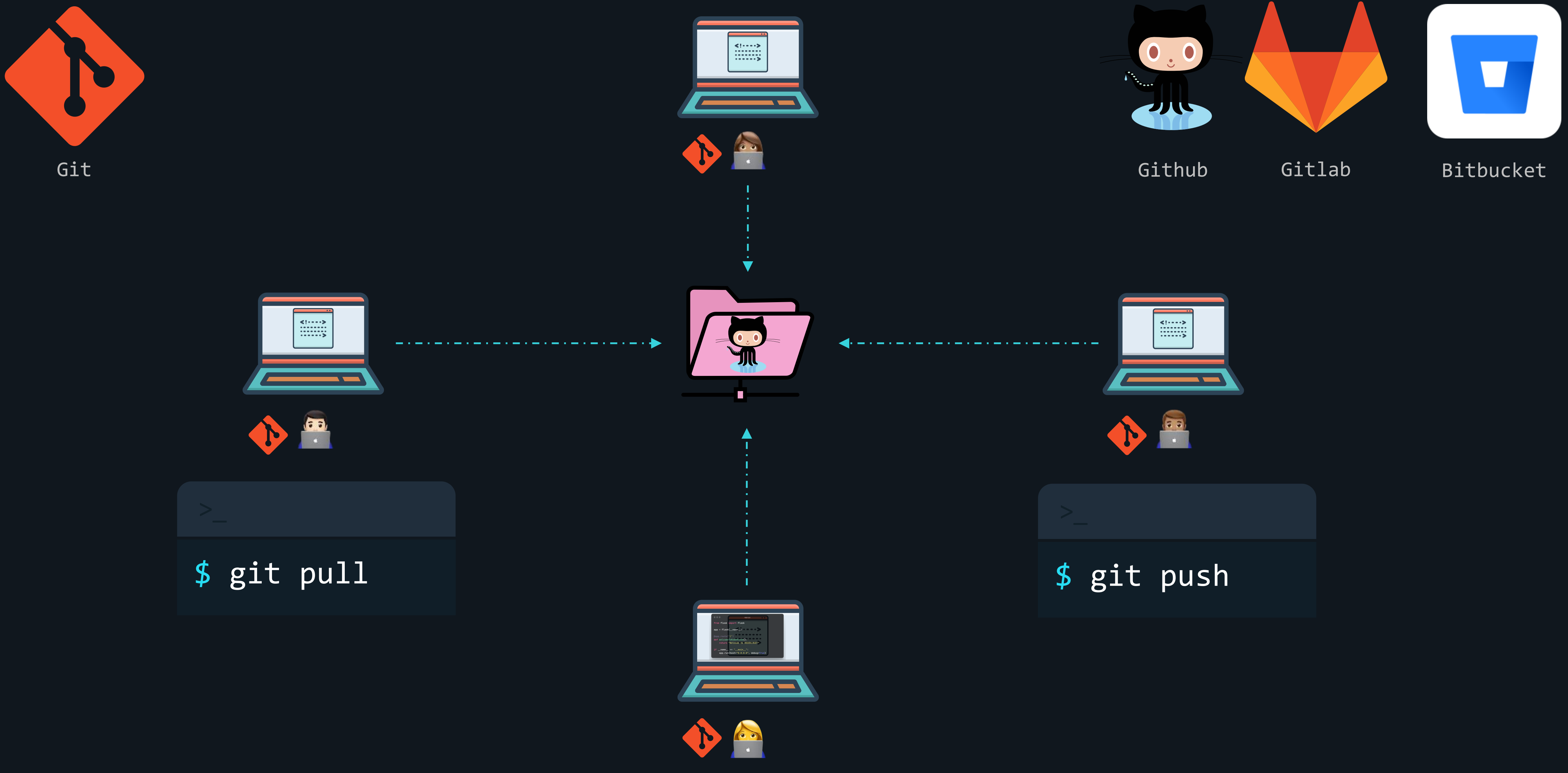
app.py

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def welcomeToKodeKloud():
    return "Welcome to KODEKLOUD!"

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

KodeKloud

KodeKloud

I
Codebase

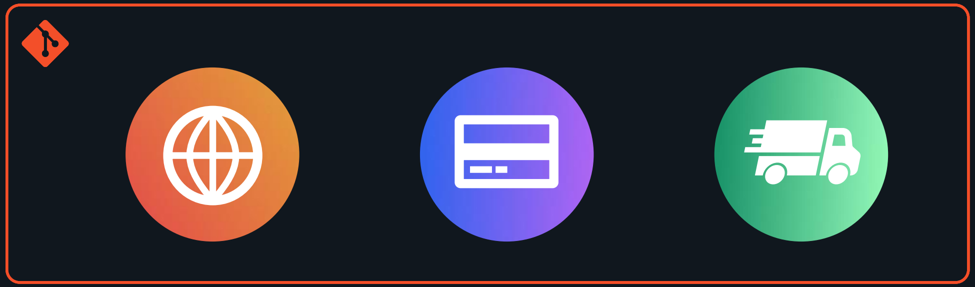


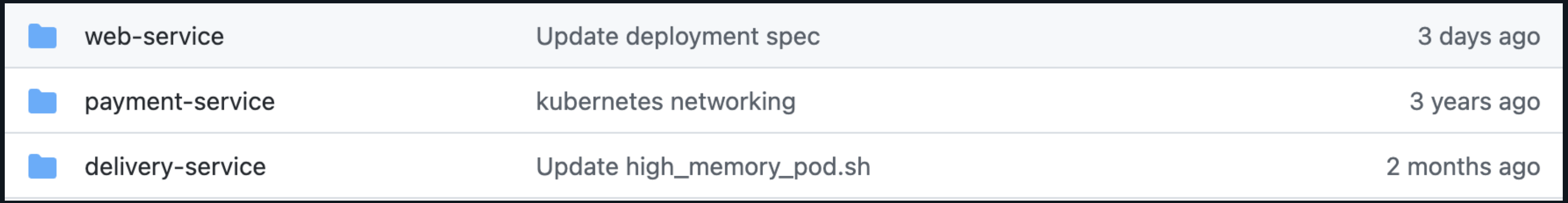
```
$ git pull
```

```
$ git push
```

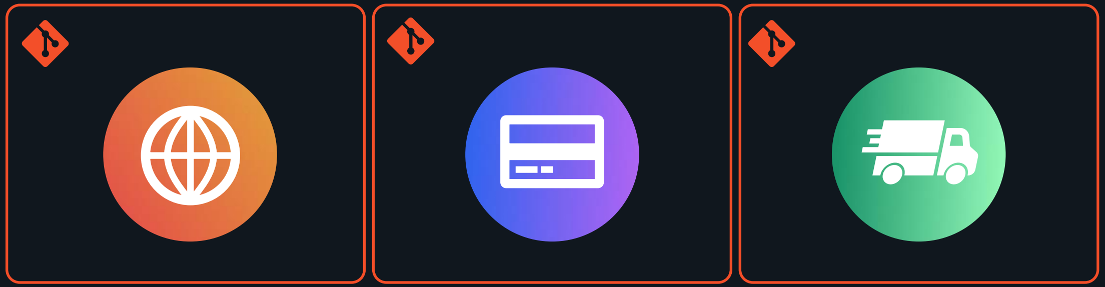
KodeKloud

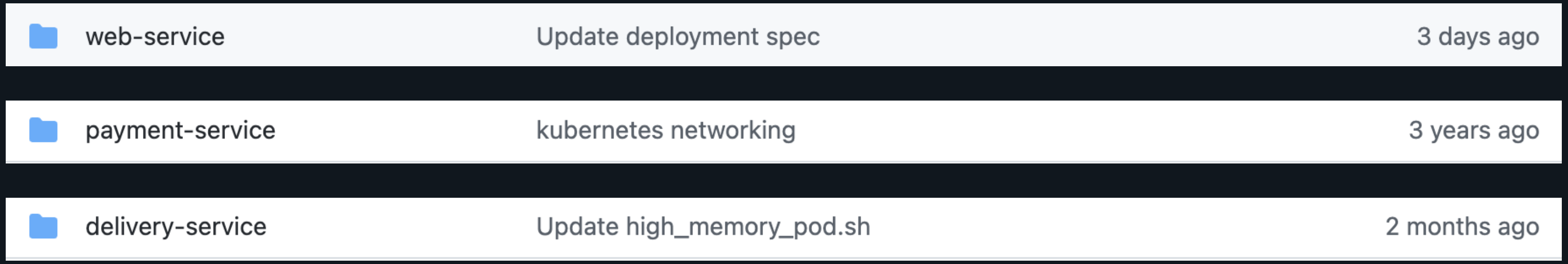
☝ Multiple apps sharing the same code is a violation of twelve-factor.



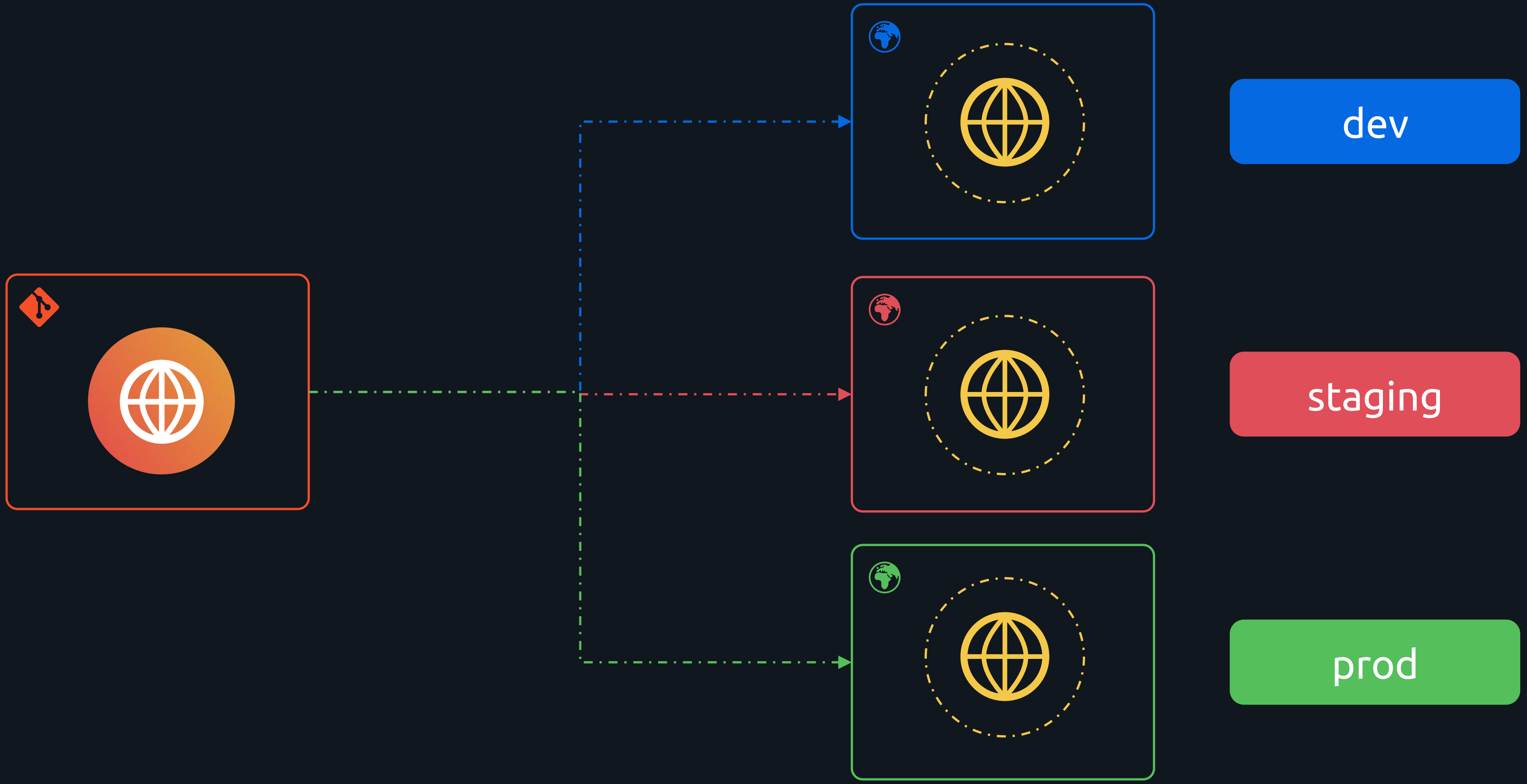


web-service	Update deployment spec	3 days ago
payment-service	kubernetes networking	3 years ago
delivery-service	Update high_memory_pod.sh	2 months ago





web-service	Update deployment spec	3 days ago
payment-service	kubernetes networking	3 years ago
delivery-service	Update high_memory_pod.sh	2 months ago


dev
staging
prod

KodeKloud

KodeKloud

II
Explicitly declare and isolate dependencies

KodeKloud

```
>_
$ pip install flask
```

app.py

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def welcomeToKodeKloud():
    return "Welcome to KODEKLOUD!"

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

☝️ A twelve-factor app never relies on implicit existence of system-wide packages.

II
Explicitly declare and isolate dependencies

```
$ pip install flask
```

app.py

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def welcomeToKodeKloud():
    return "Welcome to KODEKLOUD!"

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

KodeKloud

```
$ pip install flask
```

```
$ pip install -r requirements.txt
```

app.py

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def welcomeToKodeKloud():
    return "Welcome to KODEKLOUD!"

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

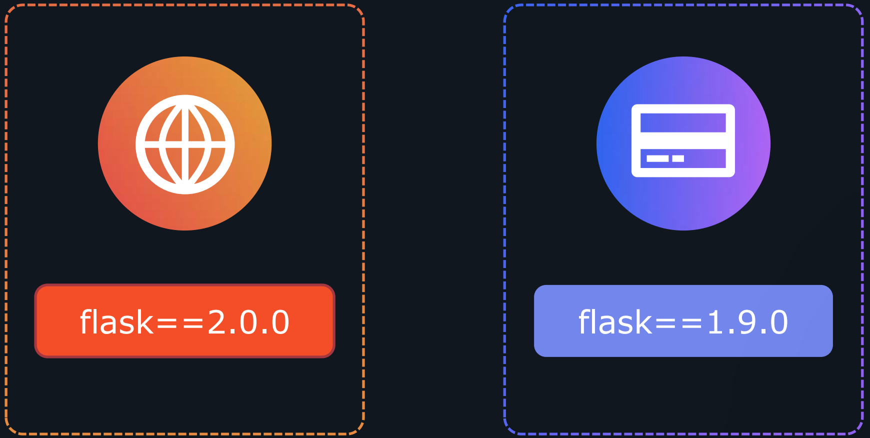
requirements.txt

```
flask==2.0.0
```

KodeKloud

II
Explicitly declare and isolate dependencies

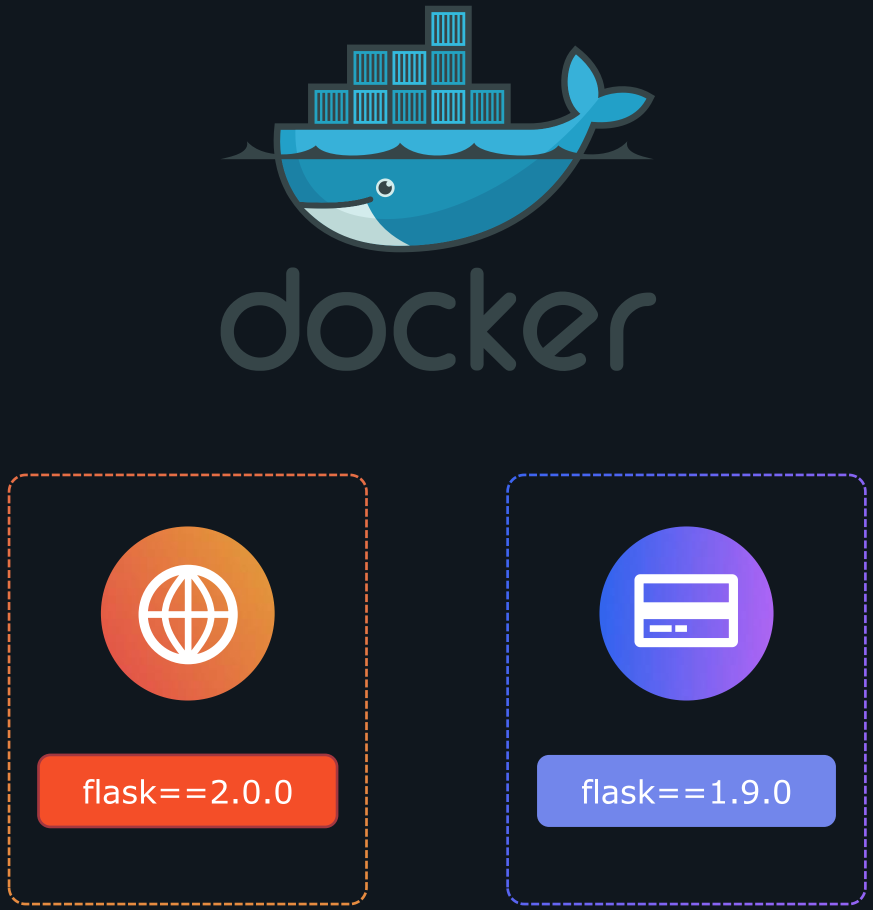
Virtual Environments (venv)



requirements.txt

```
flask==2.0.0
```

```
$ curl
```


docker
flask==2.0.0
flask==1.9.0

app.py

```python
from flask import Flask

app = Flask(__name__)

@app.route('/')
def welcomeToKodeKloud():
    return "Welcome to KODEKLOUD!"

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

requirements.txt

```
flask==2.0.0
```

Dockerfile

```dockerfile
FROM python:3.10-alpine
WORKDIR /kodekloud-twelve-factor-app
COPY requirements.txt /kodekloud-twelve-factor-app
RUN pip install -r requirements.txt --no-cache-dir
COPY . /kodekloud-twelve-factor-app
CMD python app.py
```

```
>_
$ docker build ....
```

KodeKloud

```
$ docker run ....
```

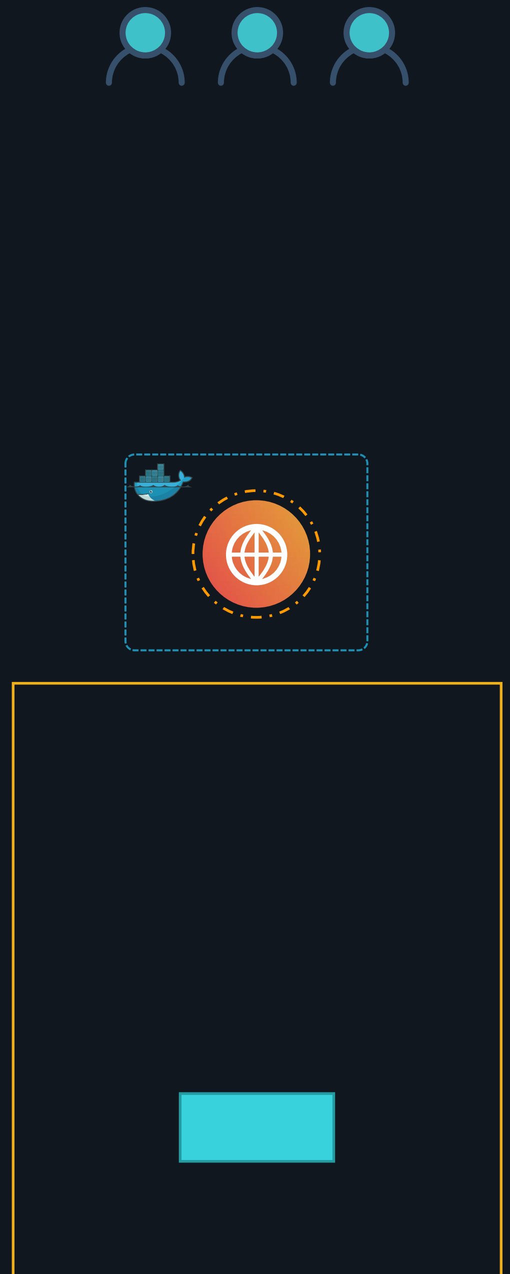
Docker Course & Labs Demo

KodeKloud

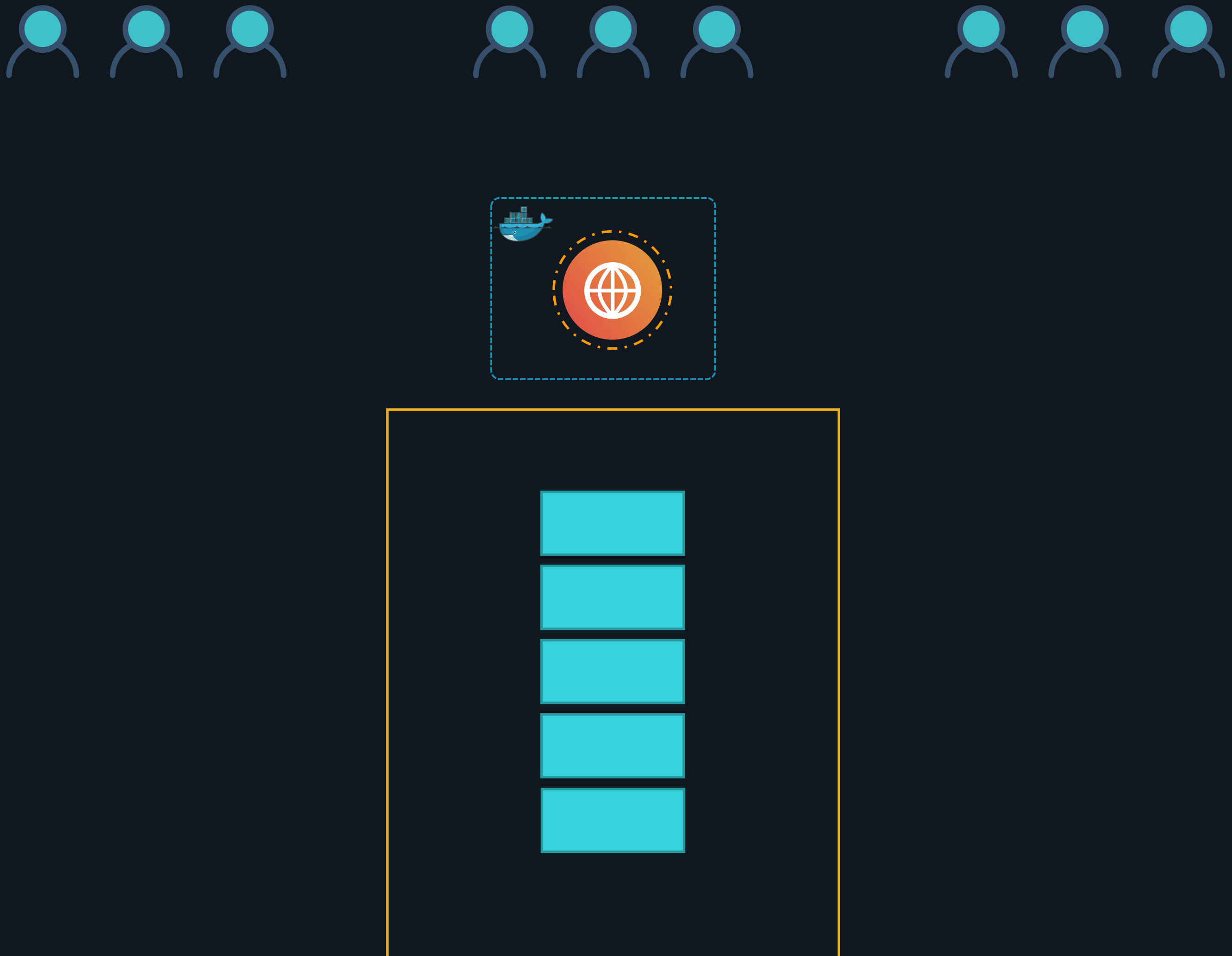
KodeKloud

III
Config

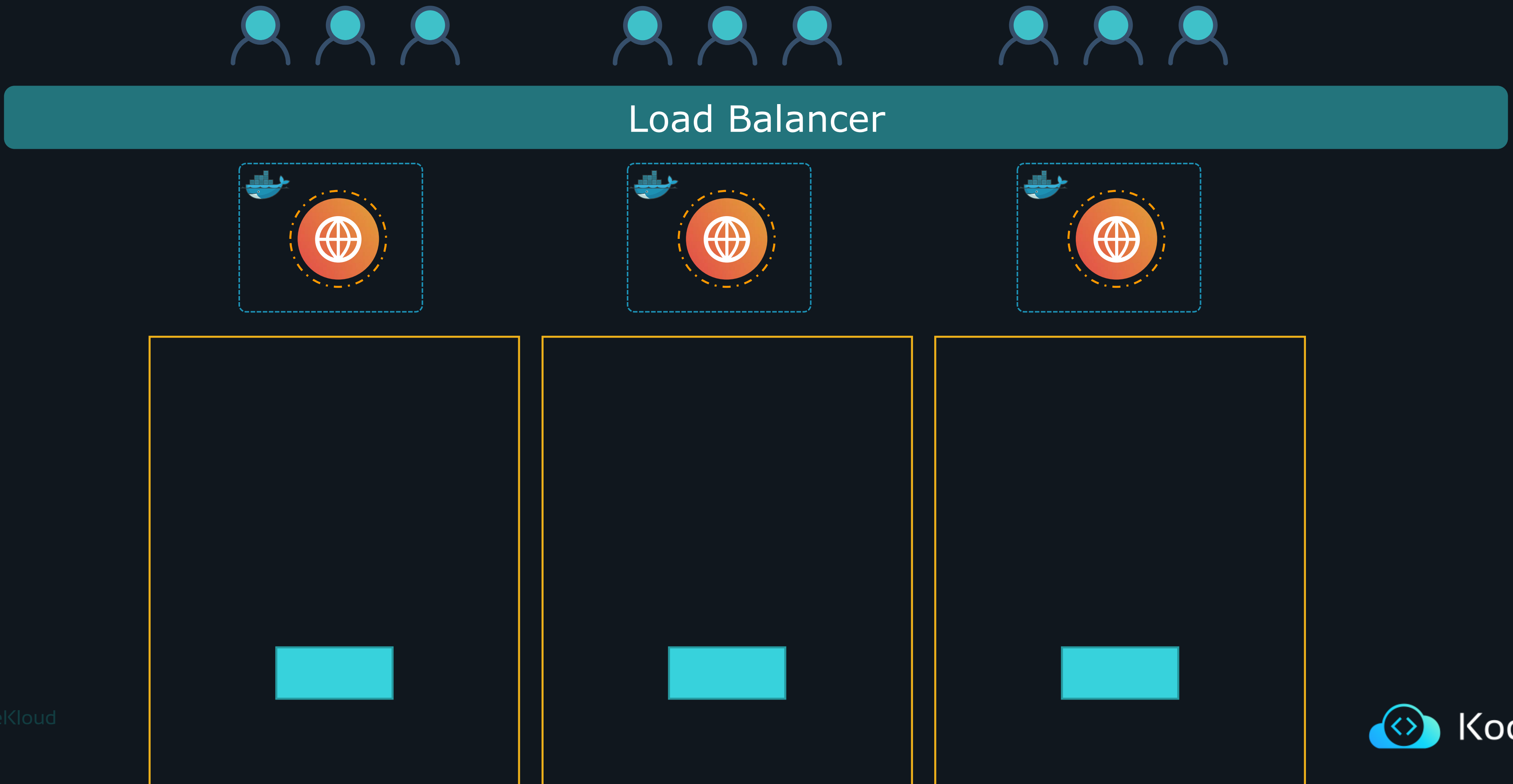
VIII
Concurrency



Vertical Scaling



Horizontal Scaling



VIII
Concurrency

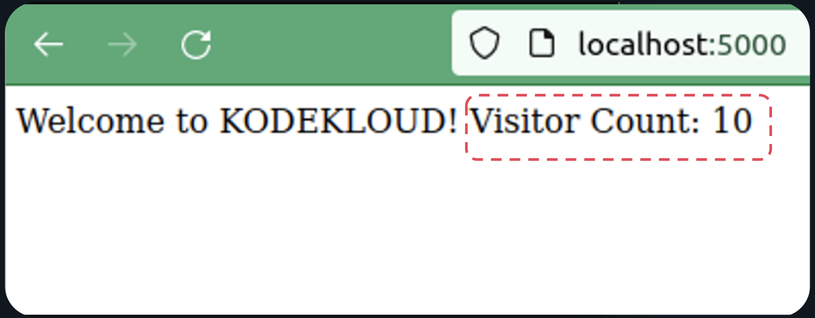
KodeKloud

KodeKloud

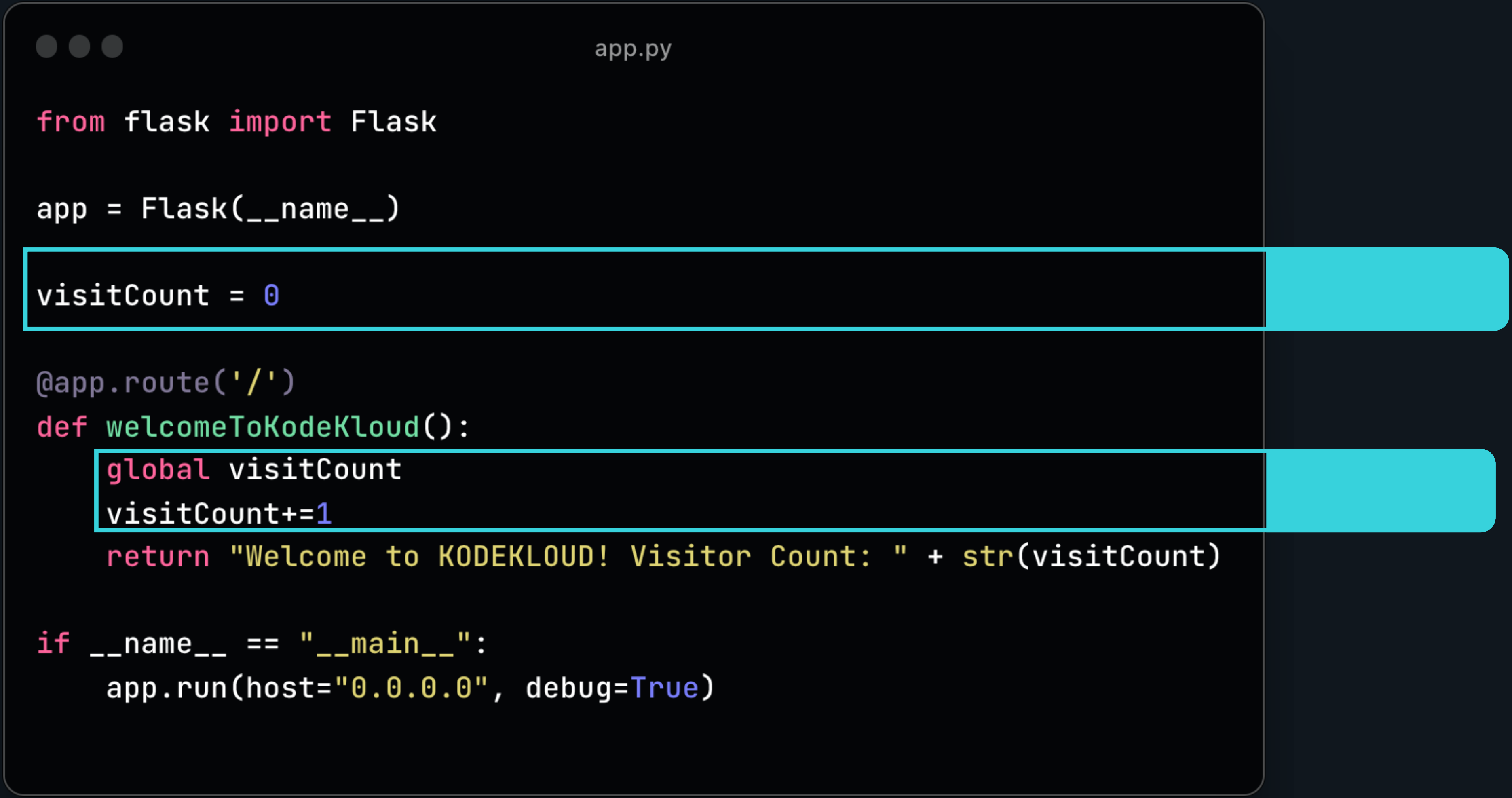
☝ Twelve-factor processes are stateless and share-nothing.

VI
Processes

KodeKloud



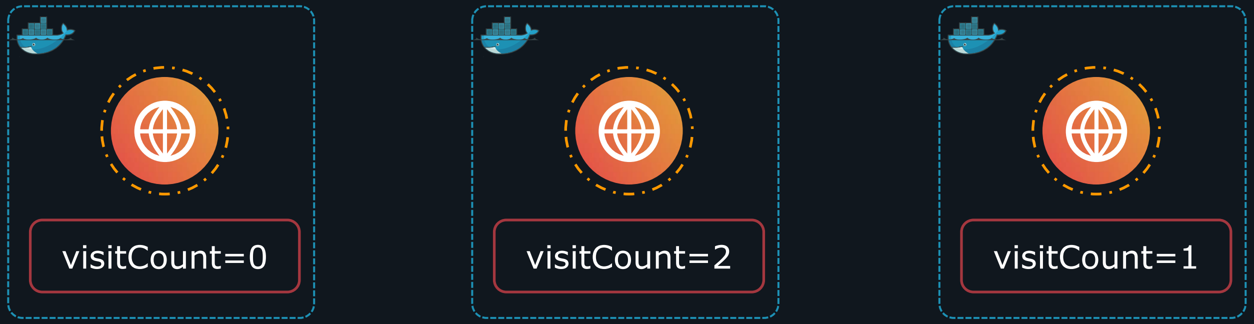
```
Welcome to KODEKLOUD! Visitor Count: 10
```



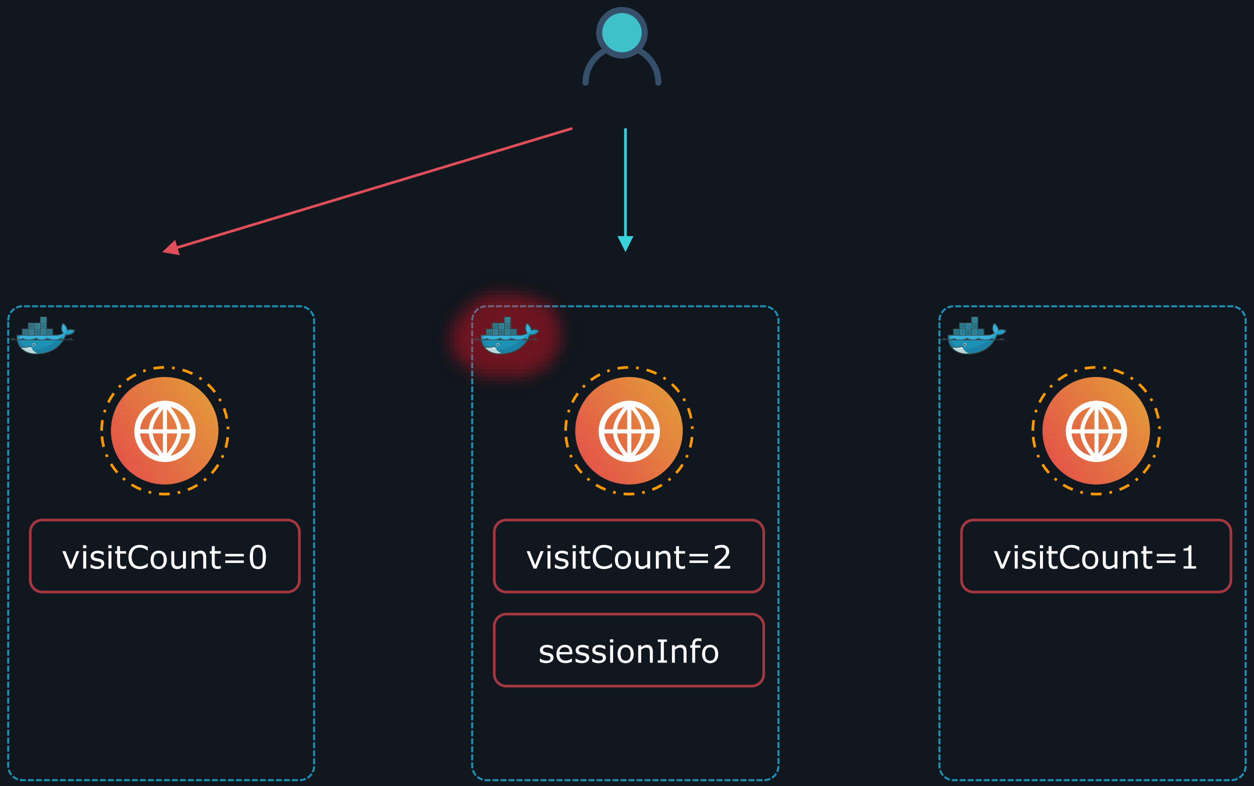
app.py

```python
from flask import Flask

app = Flask(__name__)

visitCount = 0

@app.route('/')
def welcomeToKodeKloud():
    global visitCount
    visitCount+=1
    return "Welcome to KODEKLOUD! Visitor Count: " + str(visitCount)

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

KodeKloud


visitCount=0
visitCount=2
visitCount=1

Sticky Sessions



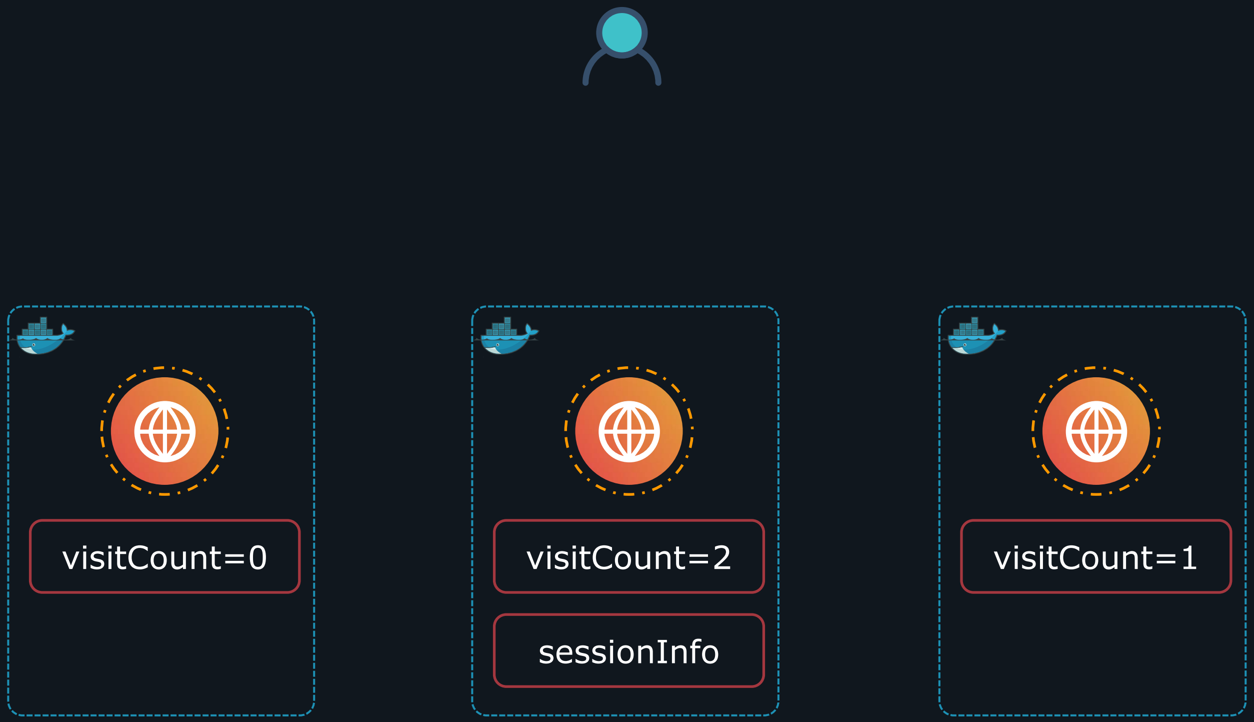
KodeKloud

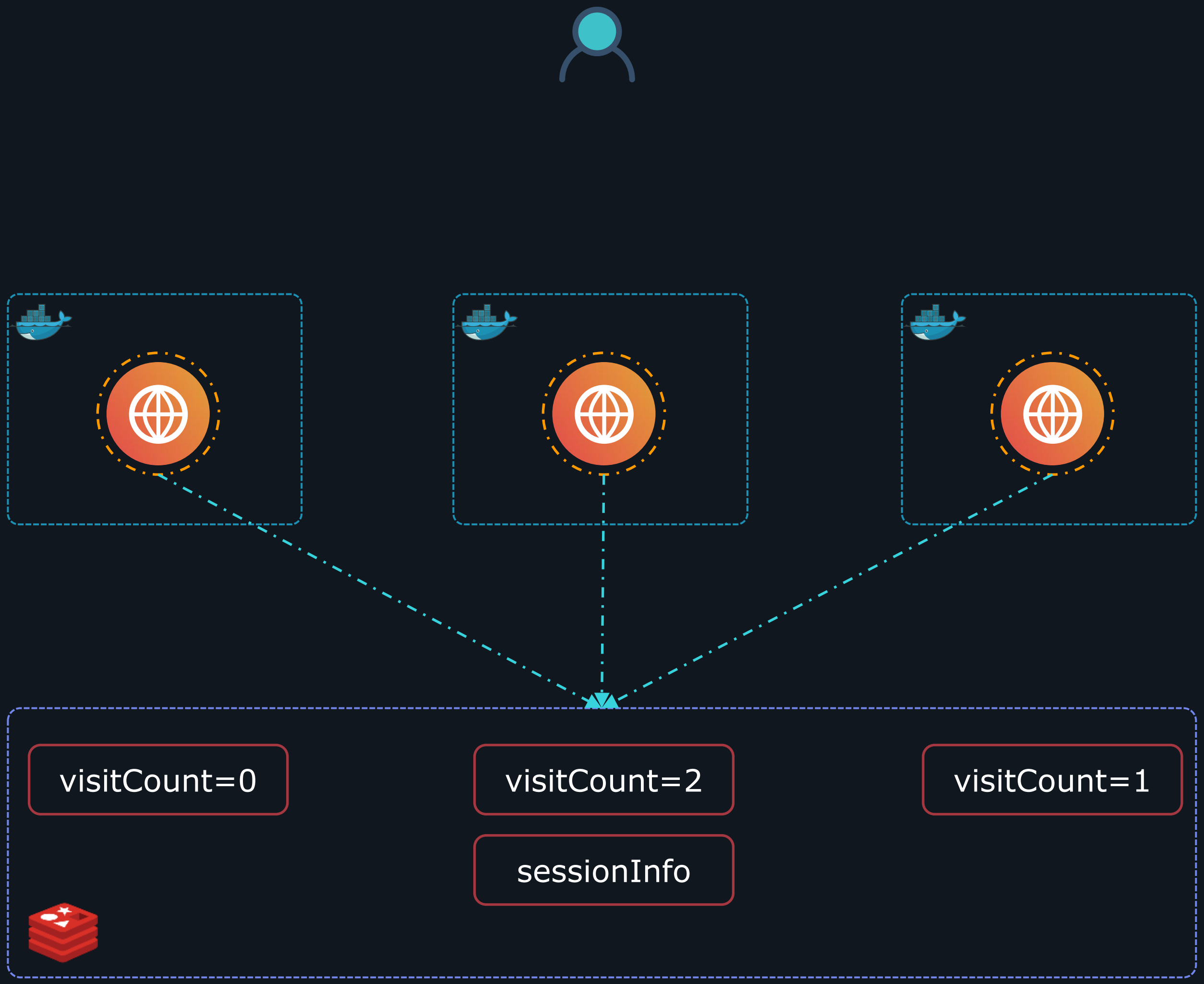
☝ Twelve-factor processes are stateless and share-nothing.

☝ Sticky sessions are a violation of twelve-factor and should never be used or relied upon.

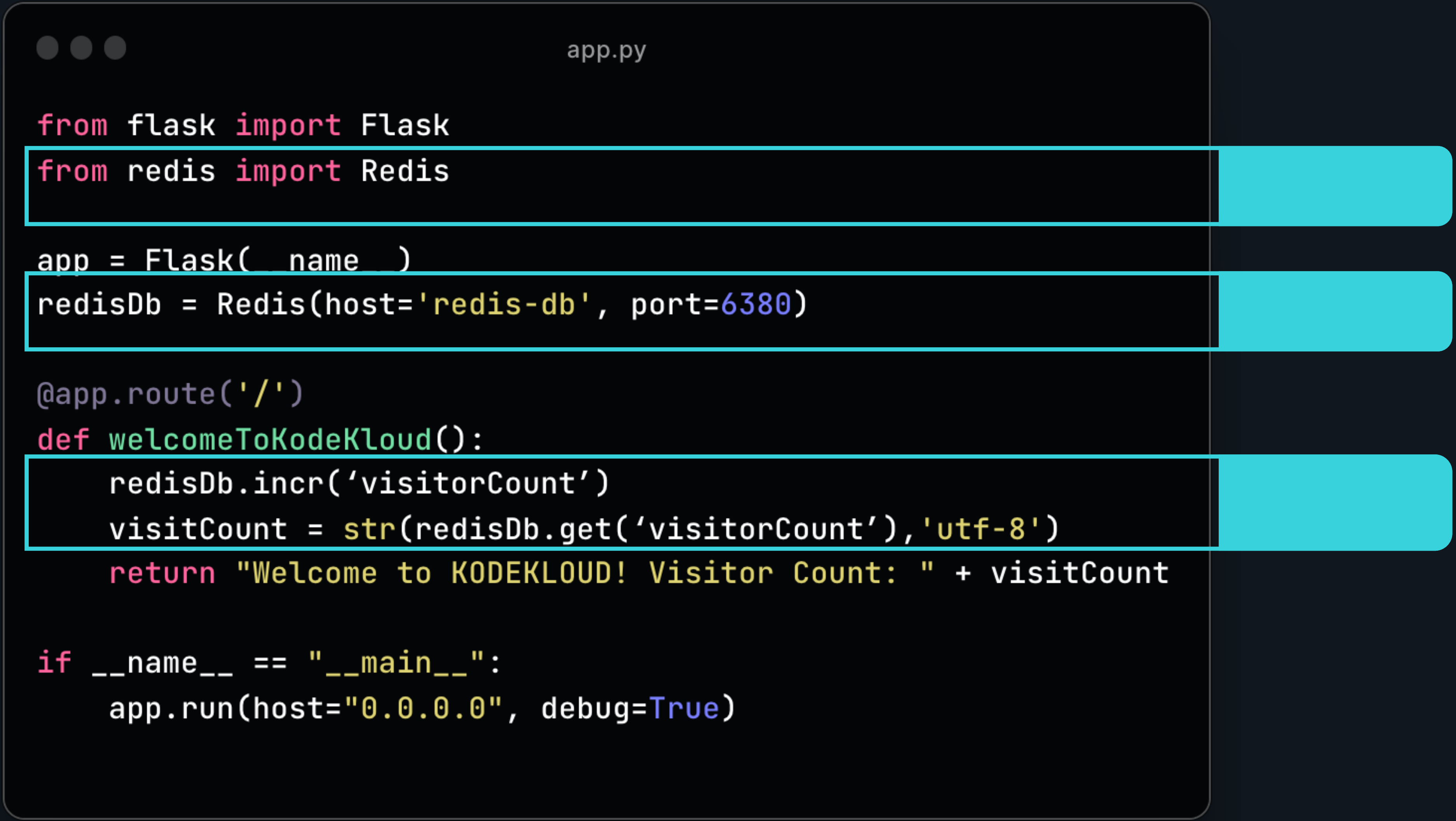
VI
Processes

KodeKloud


visitCount=0
visitCount=2
sessionInfo
visitCount=1



KodeKloud



app.py

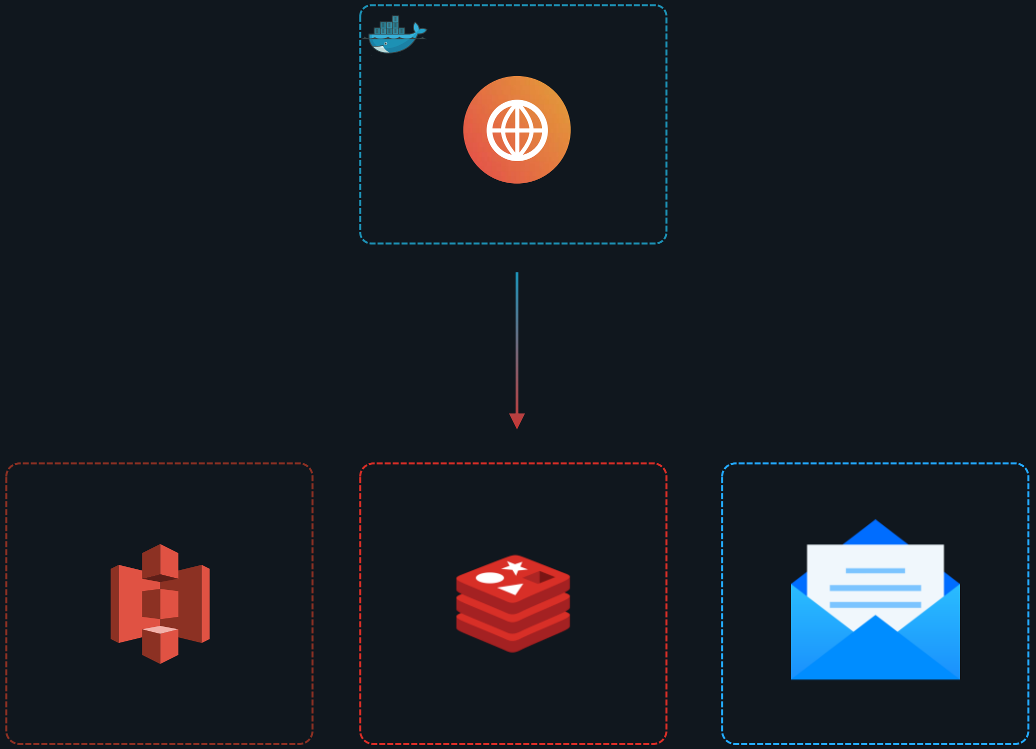
```python
from flask import Flask
from redis import Redis

app = Flask(__name__)
redisDb = Redis(host='redis-db', port=6380)

@app.route('/')
def welcomeToKodeKloud():
    redisDb.incr(‘visitorCount’)
    visitCount = str(redisDb.get(‘visitorCount’),'utf-8')
    return "Welcome to KODEKLOUD! Visitor Count: " + visitCount

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

KodeKloud

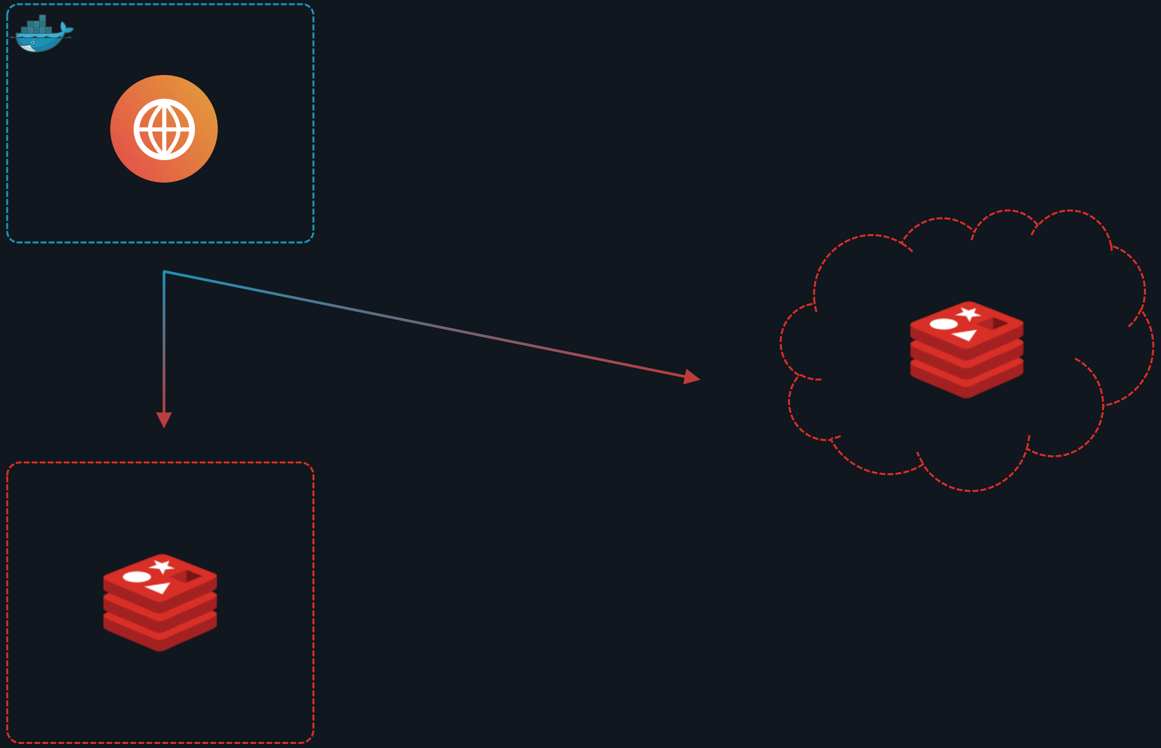
KodeKloud

IV
Backing Services

☝ Treat backing services as attached resources.



KodeKloud

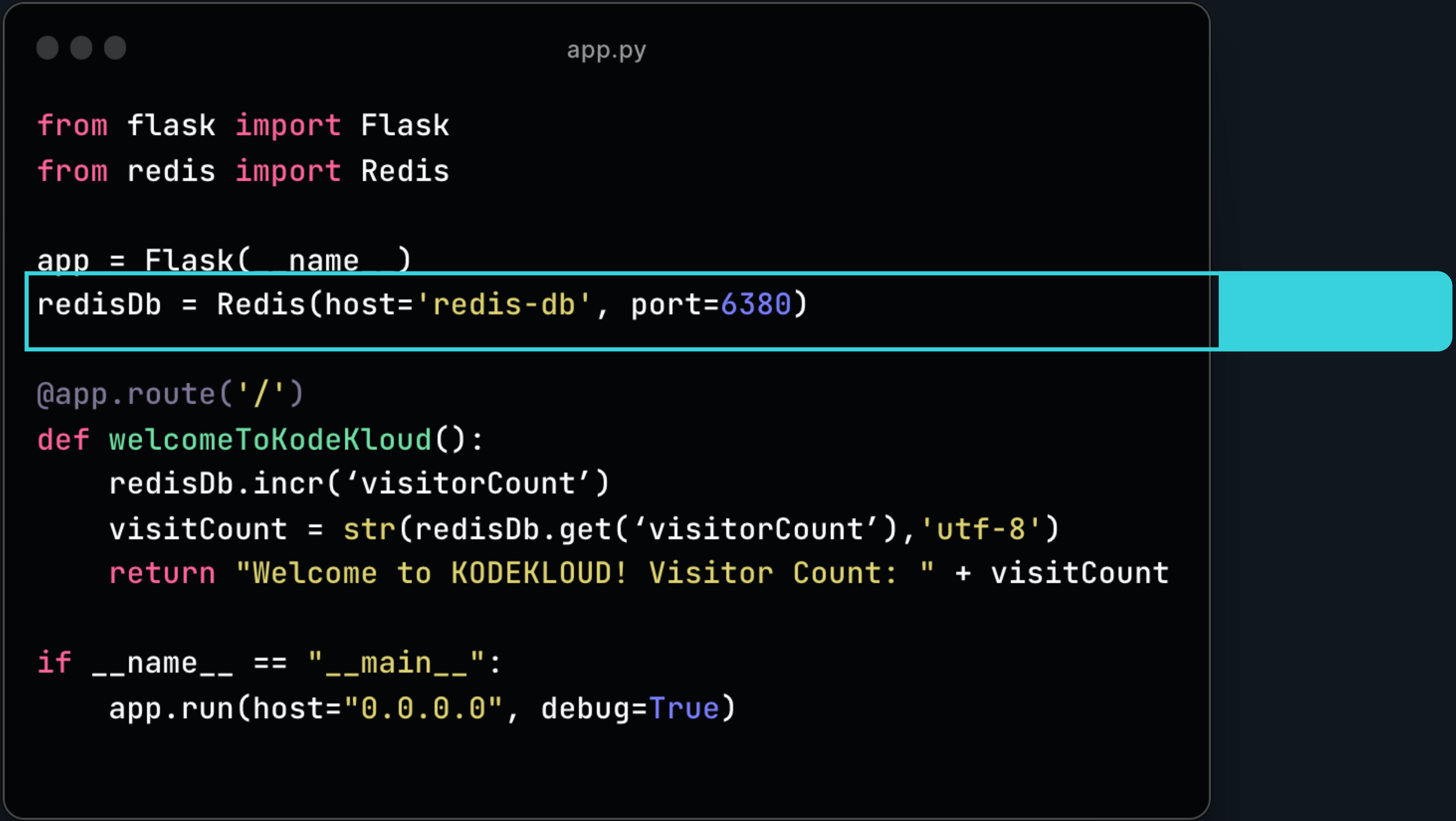


KodeKloud

KodeKloud

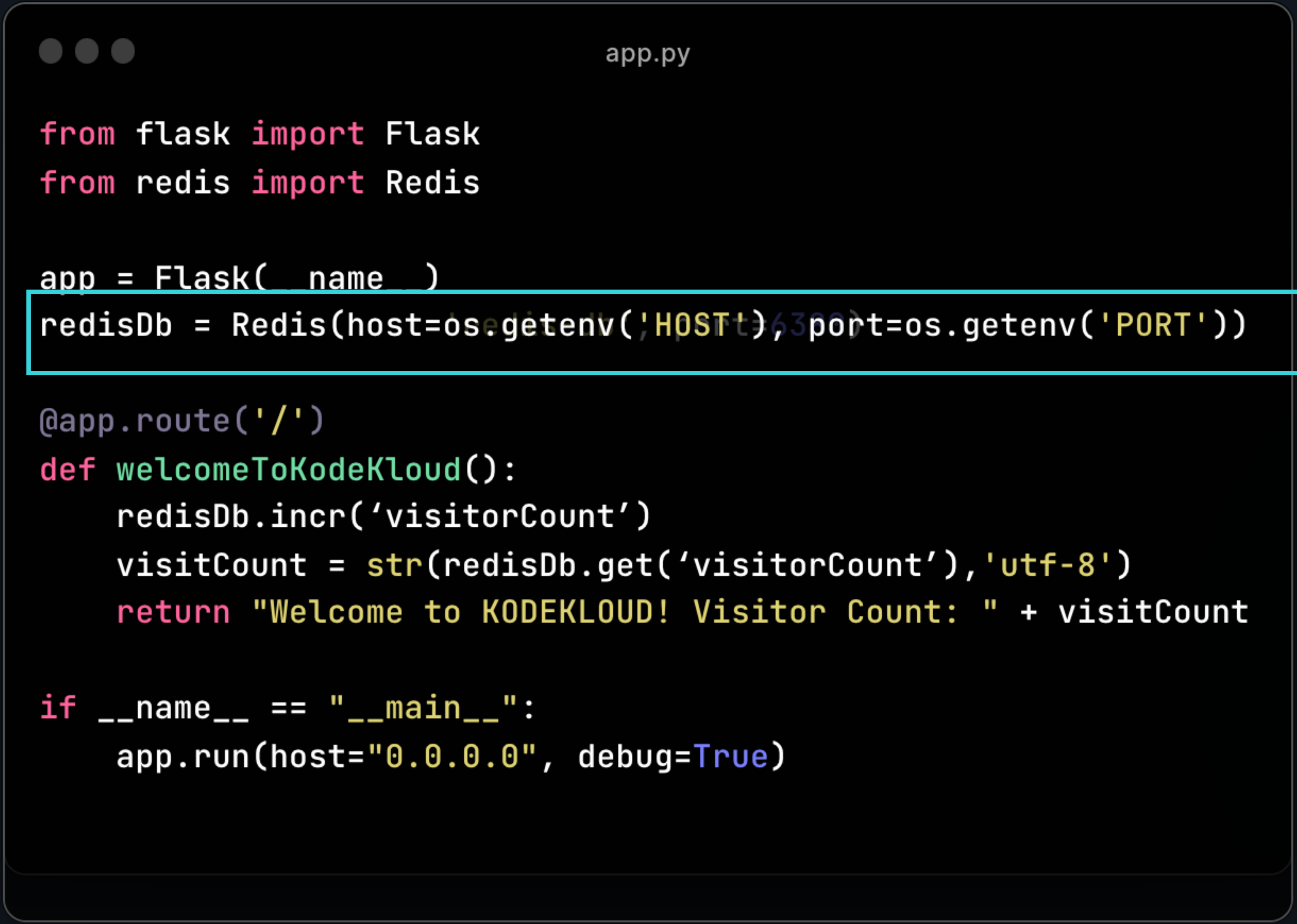
KodeKloud

III
Config



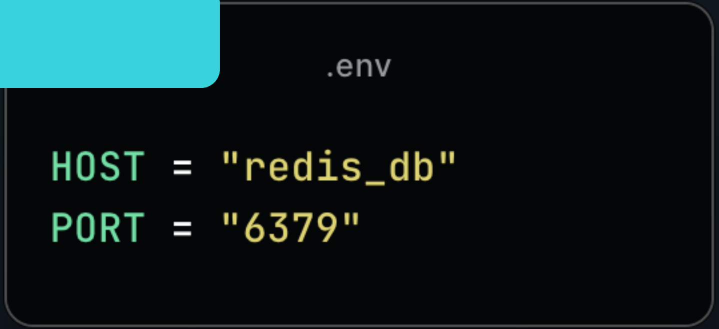
app.py

```python
from flask import Flask
from redis import Redis

app = Flask(__name__)
redisDb = Redis(host='redis-db', port=6380)

@app.route('/')
def welcomeToKodeKloud():
    redisDb.incr('visitorCount')
    visitCount = str(redisDb.get('visitorCount'),'utf-8')
    return "Welcome to KODEKLOUD! Visitor Count: " + visitCount

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



app.py

```
from flask import Flask
from redis import Redis

app = Flask(__name__)
redisDb = Redis(host=os.getenv('HOST'), port=os.getenv('PORT'))

@app.route('/')
def welcomeToKodeKloud():
    redisDb.incr(‘visitorCount’)
    visitCount = str(redisDb.get(‘visitorCount’),'utf-8')
    return "Welcome to KODEKLOUD! Visitor Count: " + visitCount

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



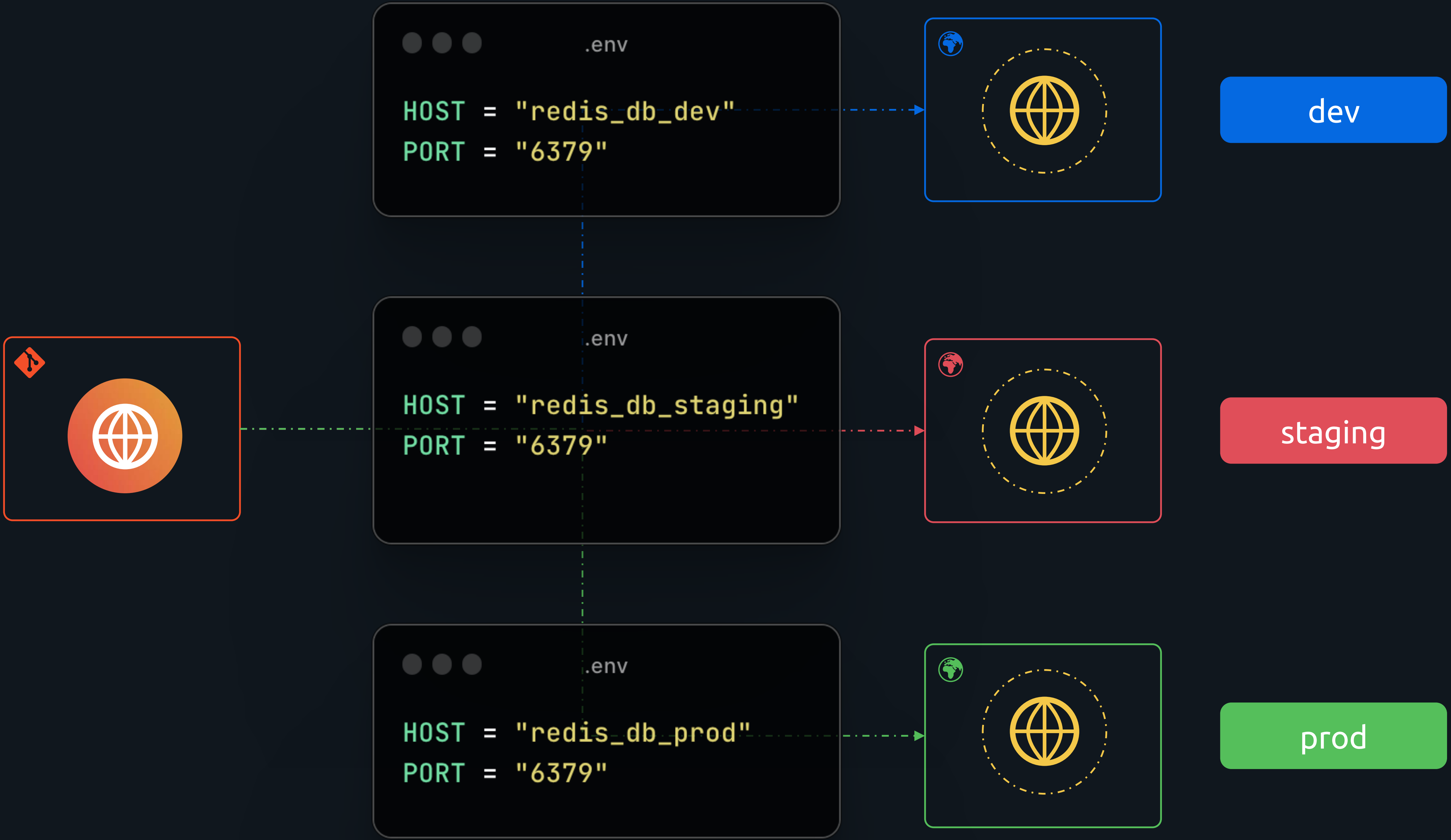
.env

```
HOST = "redis_db"
PORT = "6379"
```

KodeKloud

☝ The twelve-factor app stores config in environment variables.

III
Config



```
HOST = "redis_db_dev"
PORT = "6379"
```

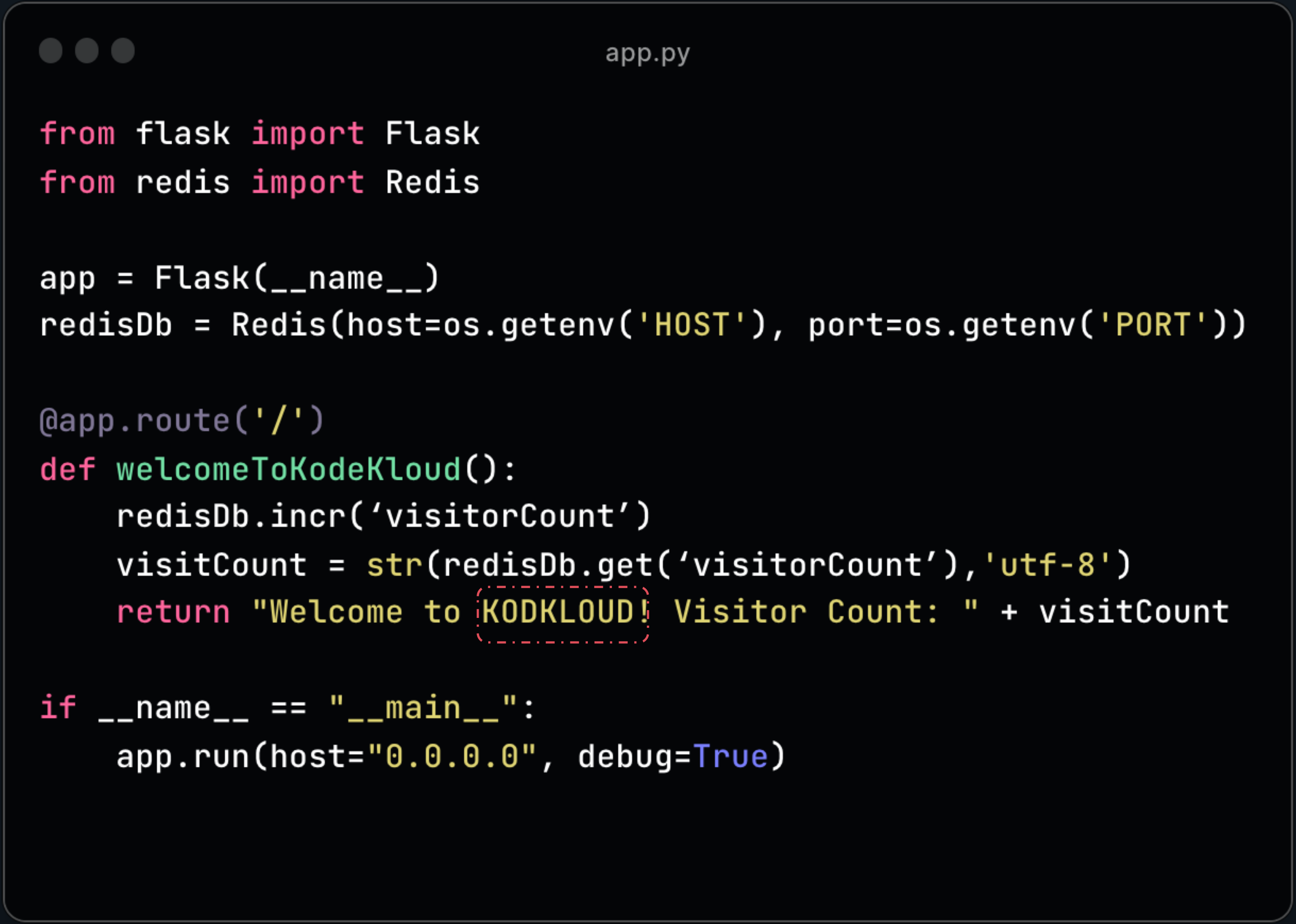
```
HOST = "redis_db_staging"
PORT = "6379"
```

```
HOST = "redis_db_prod"
PORT = "6379"
```

KodeKloud

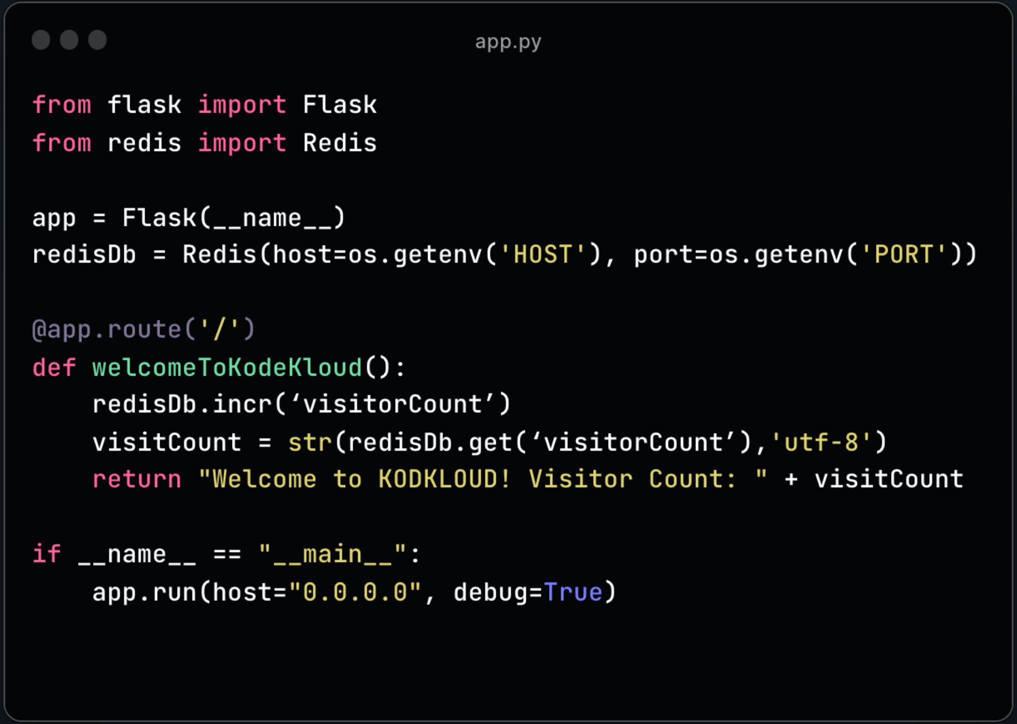
KodeKloud

V
Build, release, run



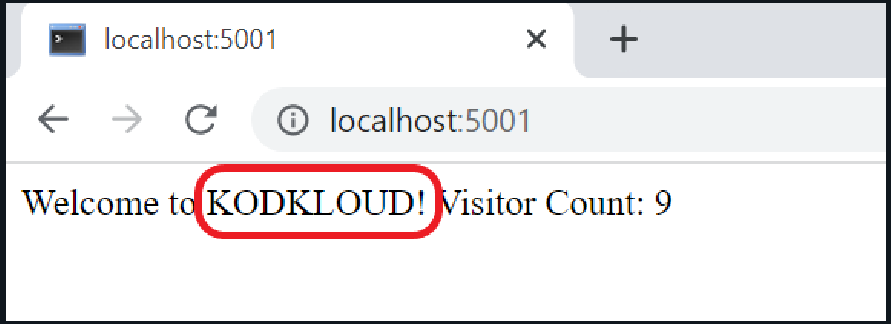
```python
from flask import Flask
from redis import Redis

app = Flask(__name__)
redisDb = Redis(host=os.getenv('HOST'), port=os.getenv('PORT'))

@app.route('/')
def welcomeToKodeKloud():
    redisDb.incr(‘visitorCount’)
    visitCount = str(redisDb.get(‘visitorCount’),'utf-8')
    return "Welcome to KODKLOUD! Visitor Count: " + visitCount

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

KodeKloud



app.py

```python
from flask import Flask
from redis import Redis

app = Flask(__name__)
redisDb = Redis(host=os.getenv('HOST'), port=os.getenv('PORT'))

@app.route('/')
def welcomeToKodeKloud():
    redisDb.incr('visitorCount')
    visitCount = str(redisDb.get('visitorCount'),'utf-8')
    return "Welcome to KODKLOUD! Visitor Count: " + visitCount

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



Welcome to KODKLOUD! Visitor Count: 9

☝️The twelve-factor app uses strict separation between the build, release, and run stages.

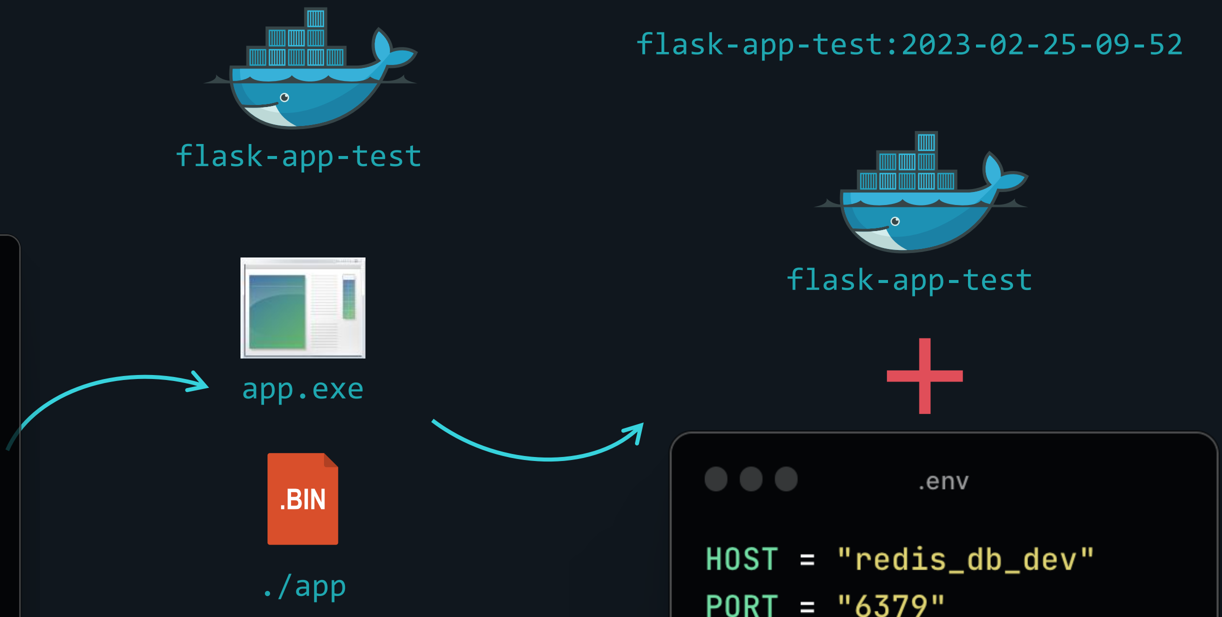
V
Build, release, run

flask-app-test:v1

flask-app-test:v2

flask-app-test:v3

flask-app-test:2023-02-25-09-52



app.py

```python
from flask import Flask
from redis import Redis

app = Flask(__name__)
redisDb = Redis(host=os.getenv('HOST'), port=os.getenv('PORT'))

@app.route('/')
def welcomeToKodeKloud():
    redisDb.incr(‘visitorCount’)
    visitCount = str(redisDb.get(‘visitorCount’),'utf-8')
    return "Welcome to KODEKLOUD! Visitor Count: " + visitCount

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

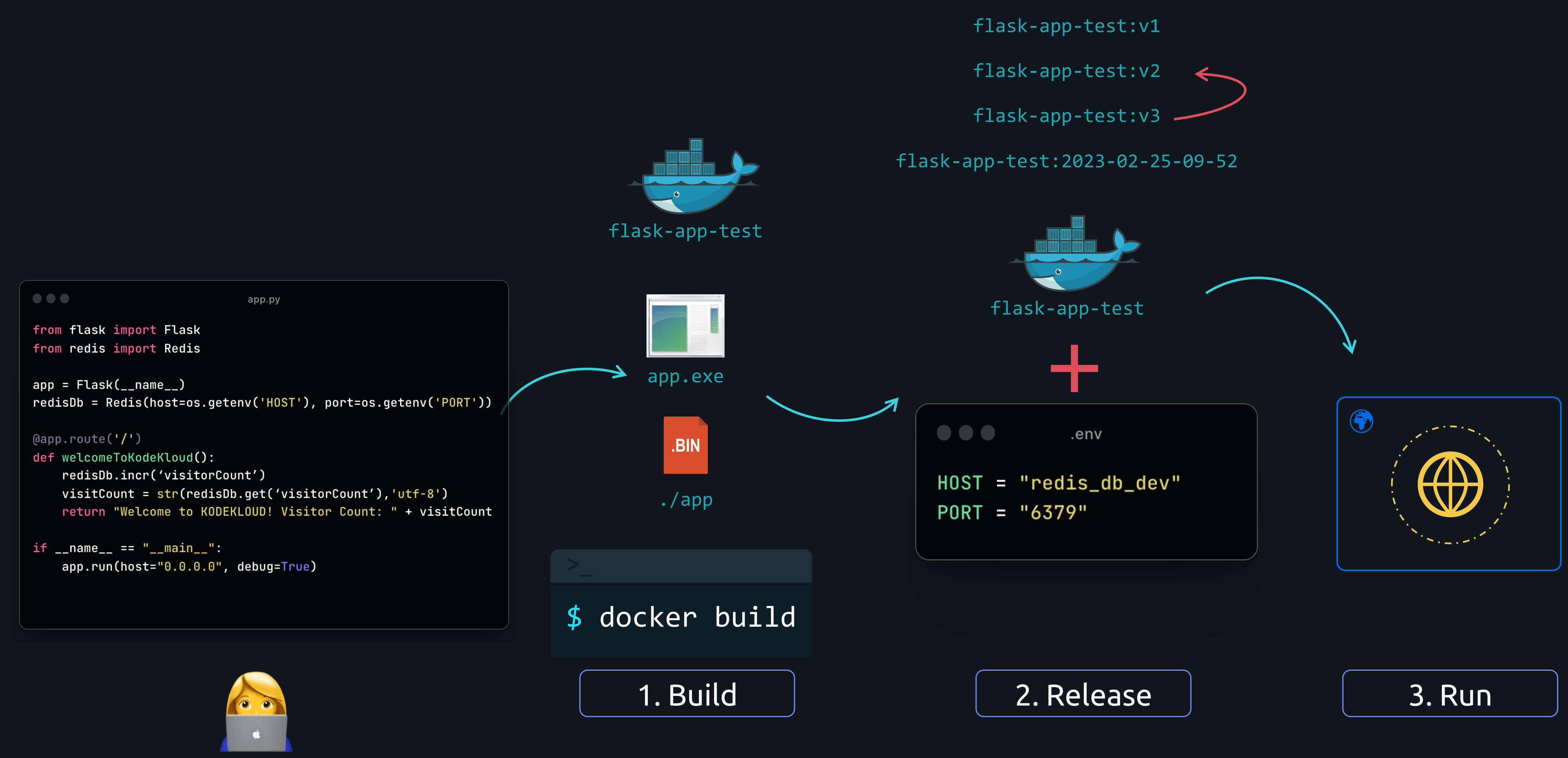
.env

```
HOST = "redis_db_dev"
PORT = "6379"
```

```
$ docker build
```

1. Build

2. Release



app.py

```python
from flask import Flask
from redis import Redis

app = Flask(__name__)
redisDb = Redis(host=os.getenv('HOST'), port=os.getenv('PORT'))

@app.route('/')
def welcomeToKodeKloud():
    redisDb.incr(‘visitorCount’)
    visitCount = str(redisDb.get(‘visitorCount’),'utf-8')
    return "Welcome to KODEKLOUD! Visitor Count: " + visitCount

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

```
$ docker build
```

.env

```
HOST = "redis_db_dev"
PORT = "6379"
```

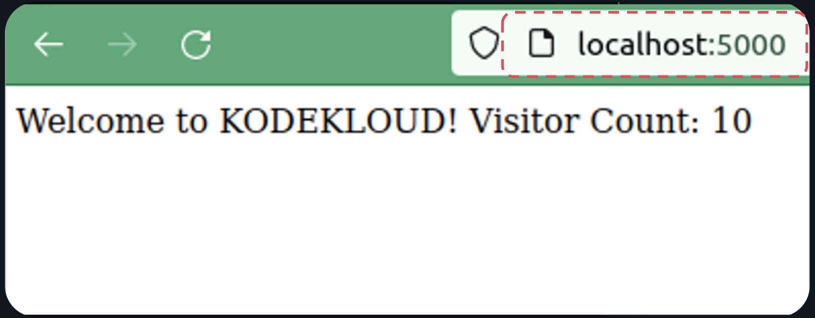
☝ The twelve-factor app uses strict separation between the build, release, and run stages.

V
Build, release, run

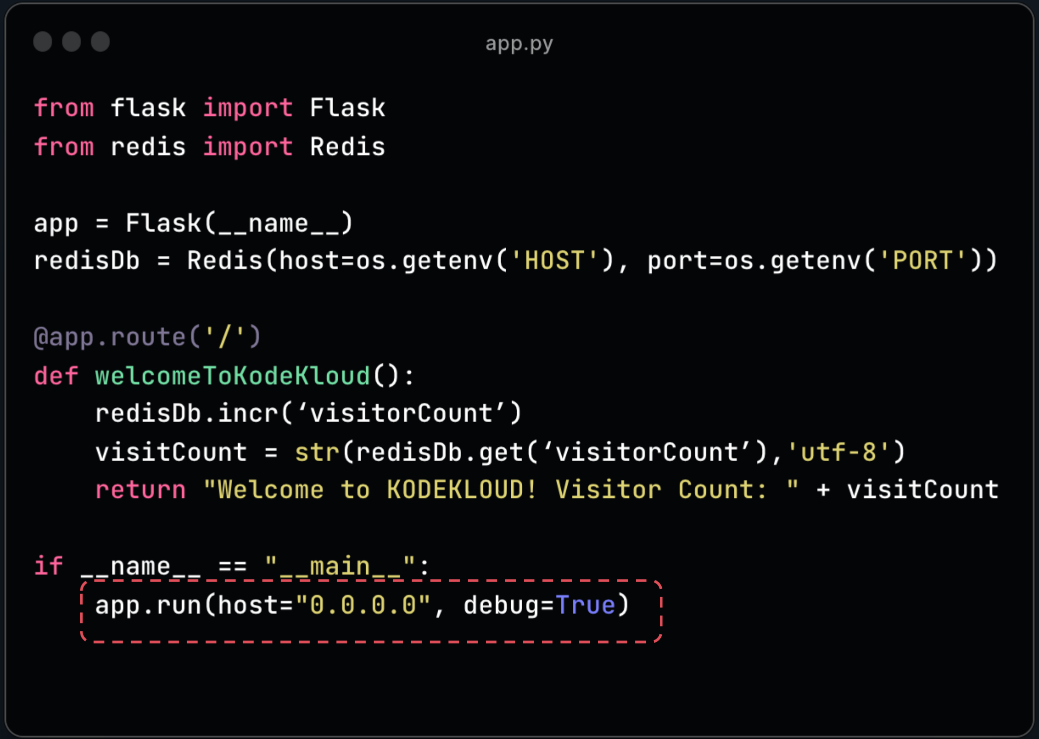
KodeKloud

KodeKloud

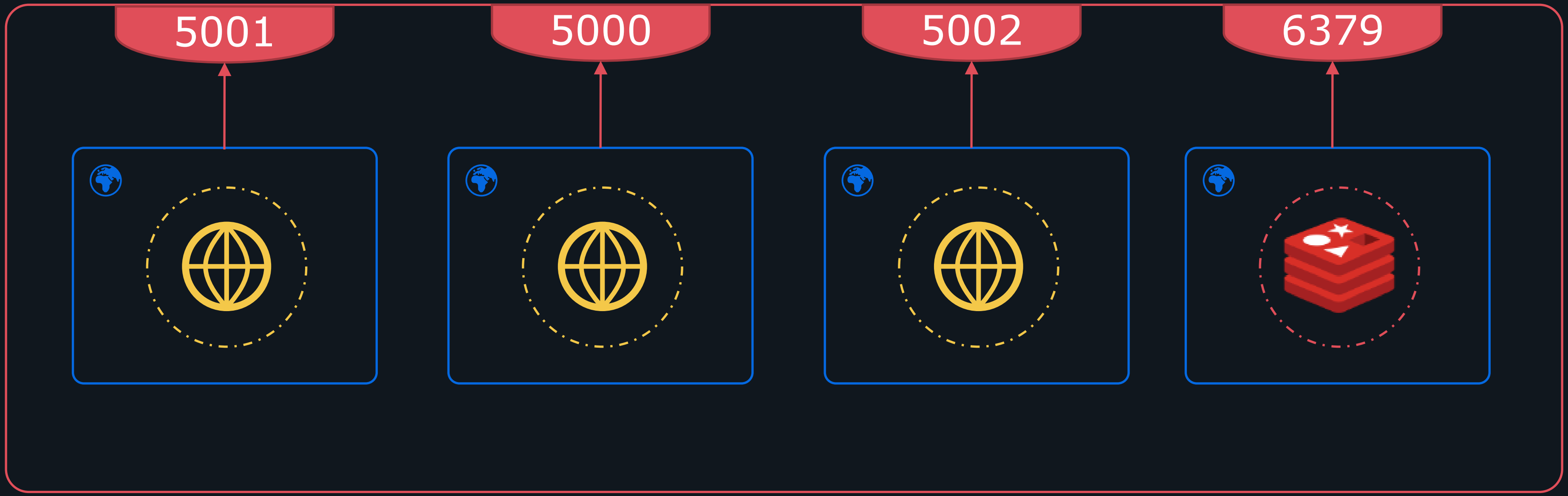
VII
Port Binding



Welcome to KODEKLOUD! Visitor Count: 10



```
from flask import Flask
from redis import Redis

app = Flask(__name__)
redisDb = Redis(host=os.getenv('HOST'), port=os.getenv('PORT'))

@app.route('/')
def welcomeToKodeKloud():
    redisDb.incr(‘visitorCount’)
    visitCount = str(redisDb.get(‘visitorCount’),'utf-8')
    return "Welcome to KODEKLOUD! Visitor Count: " + visitCount

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


5001
5000
5002
6379

☝ The twelve-factor app is completely self-contained

VII
Port Binding

KodeKloud

KodeKloud

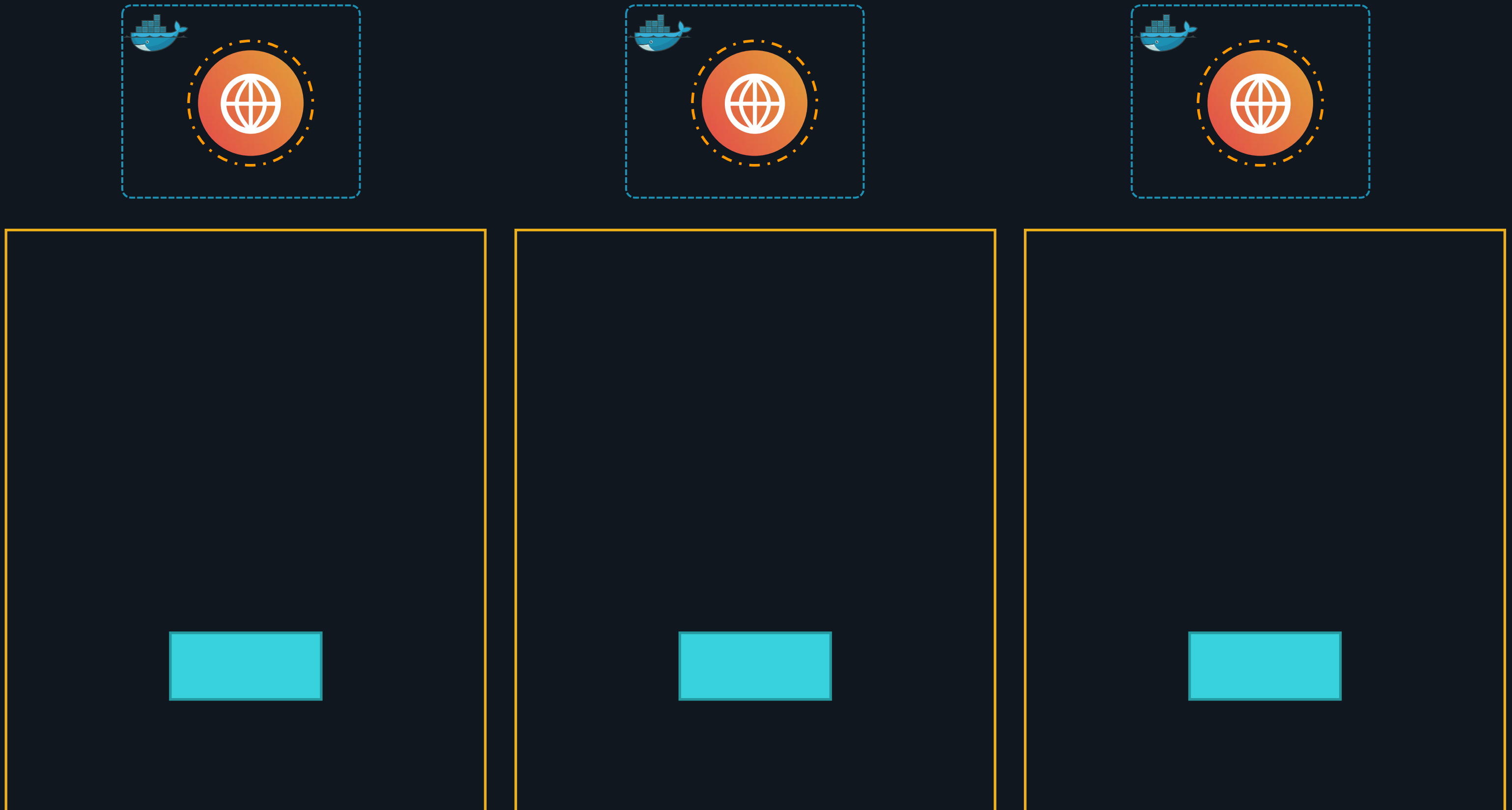
☝ The twelve-factor app's processes are disposable, meaning they can be started or stopped at a moment's notice.

IX
Disposability

Horizontal Scaling

```
>_
$ docker stop ....
```

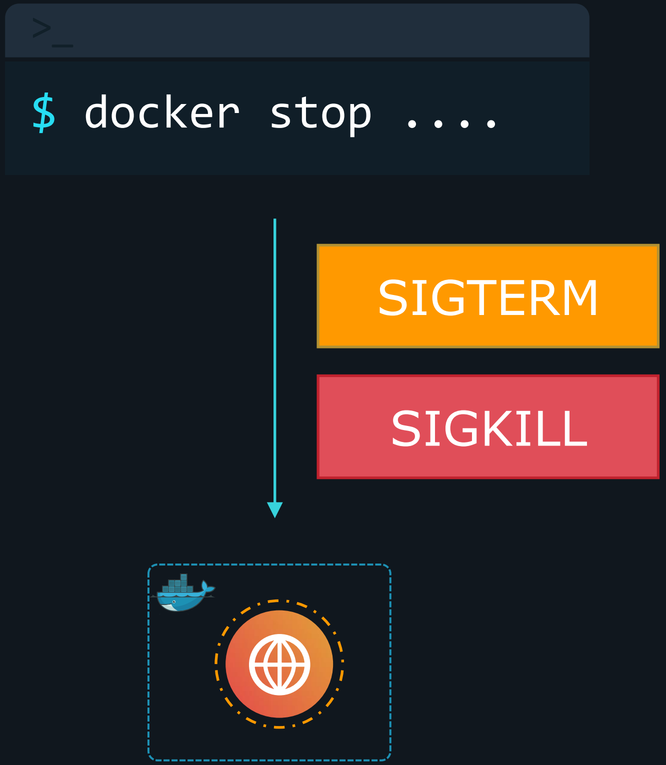
```
>_
$ docker stop ....
```

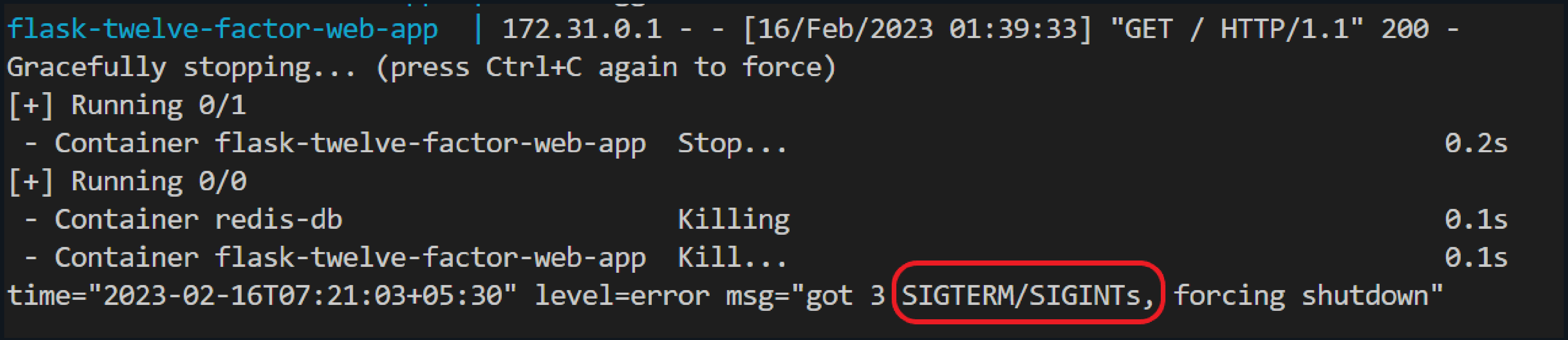


☝ The twelve-factor app's processes are disposable, meaning they can be started or stopped at a moment's notice.

☝ The twelve-factor app's processes should shutdown gracefully when they receive a SIGTERM signal from the process manager.

IX
Disposability





```
flask-twelve-factor-web-app  | 172.31.0.1 - - [16/Feb/2023 01:39:33] "GET / HTTP/1.1" 200 -
Gracefully stopping... (press Ctrl+C again to force)
[+] Running 0/1
 - Container flask-twelve-factor-web-app  Stop...                                          0.2s
[+] Running 0/0
 - Container redis-db                     Killing                                          0.1s
 - Container flask-twelve-factor-web-app  Kill...                                          0.1s
time="2023-02-16T07:21:03+05:30" level=error msg="got 3 SIGTERM/SIGINTs, forcing shutdown"
```

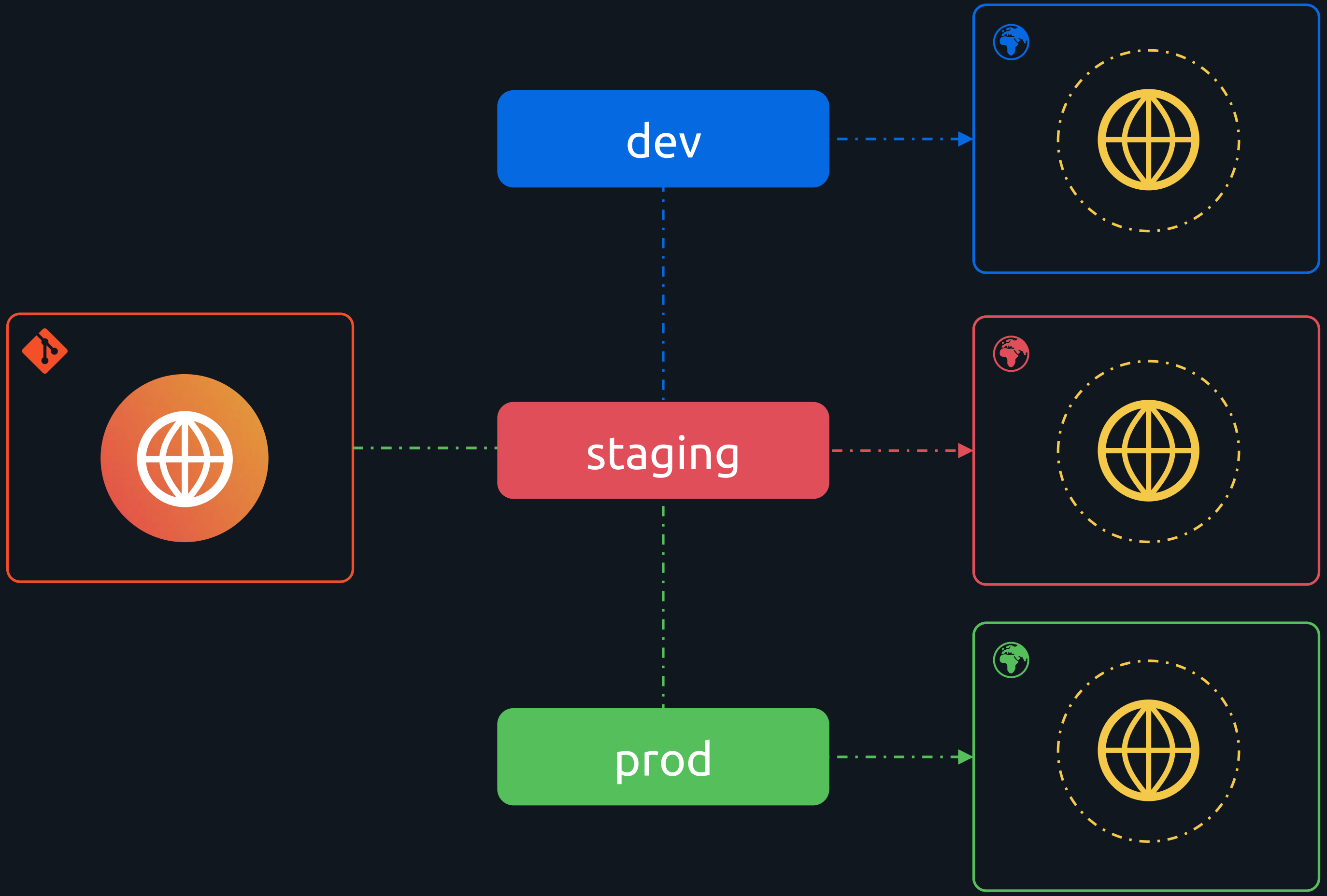
KodeKloud

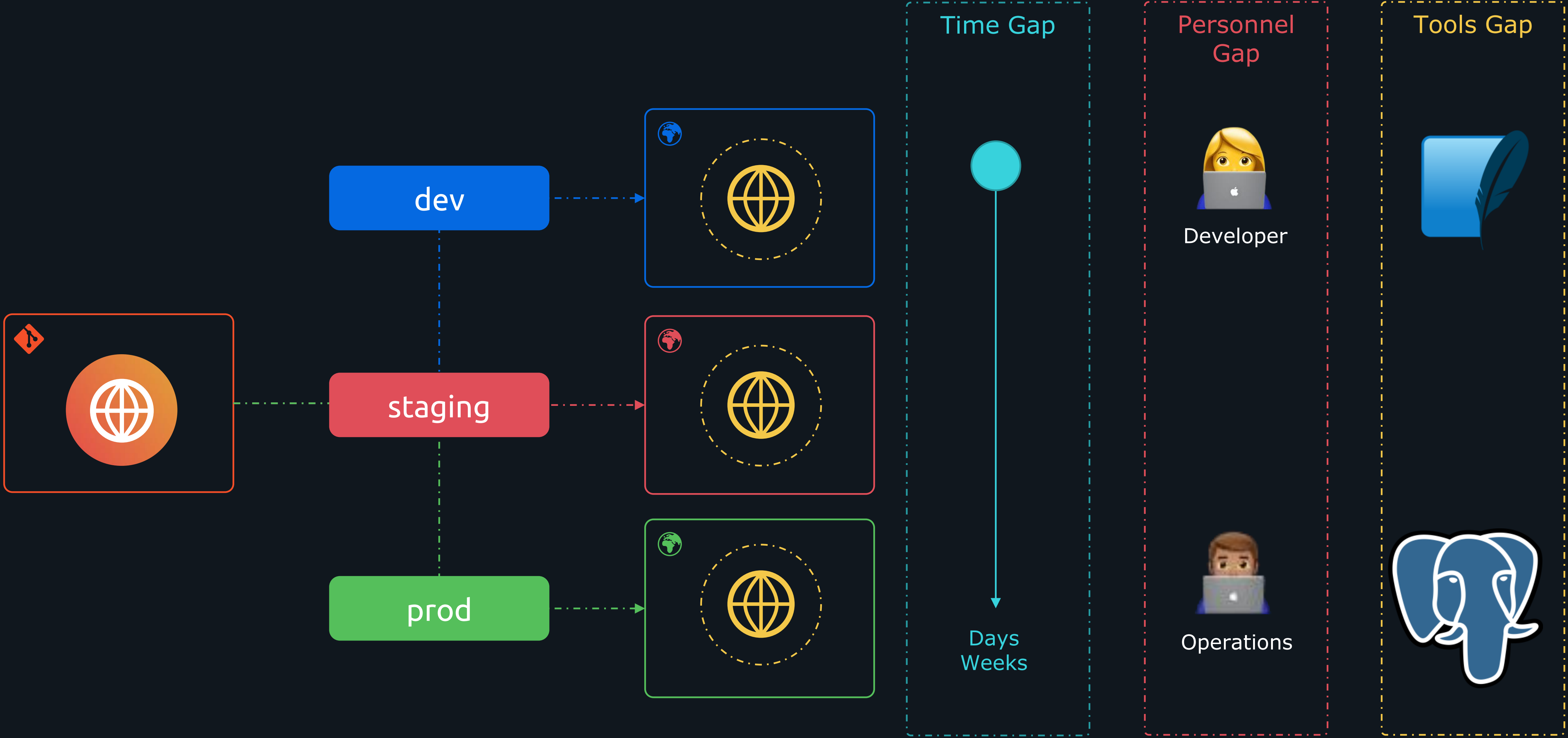
KodeKloud

KodeKloud

X
Dev/prod parity

KodeKloud

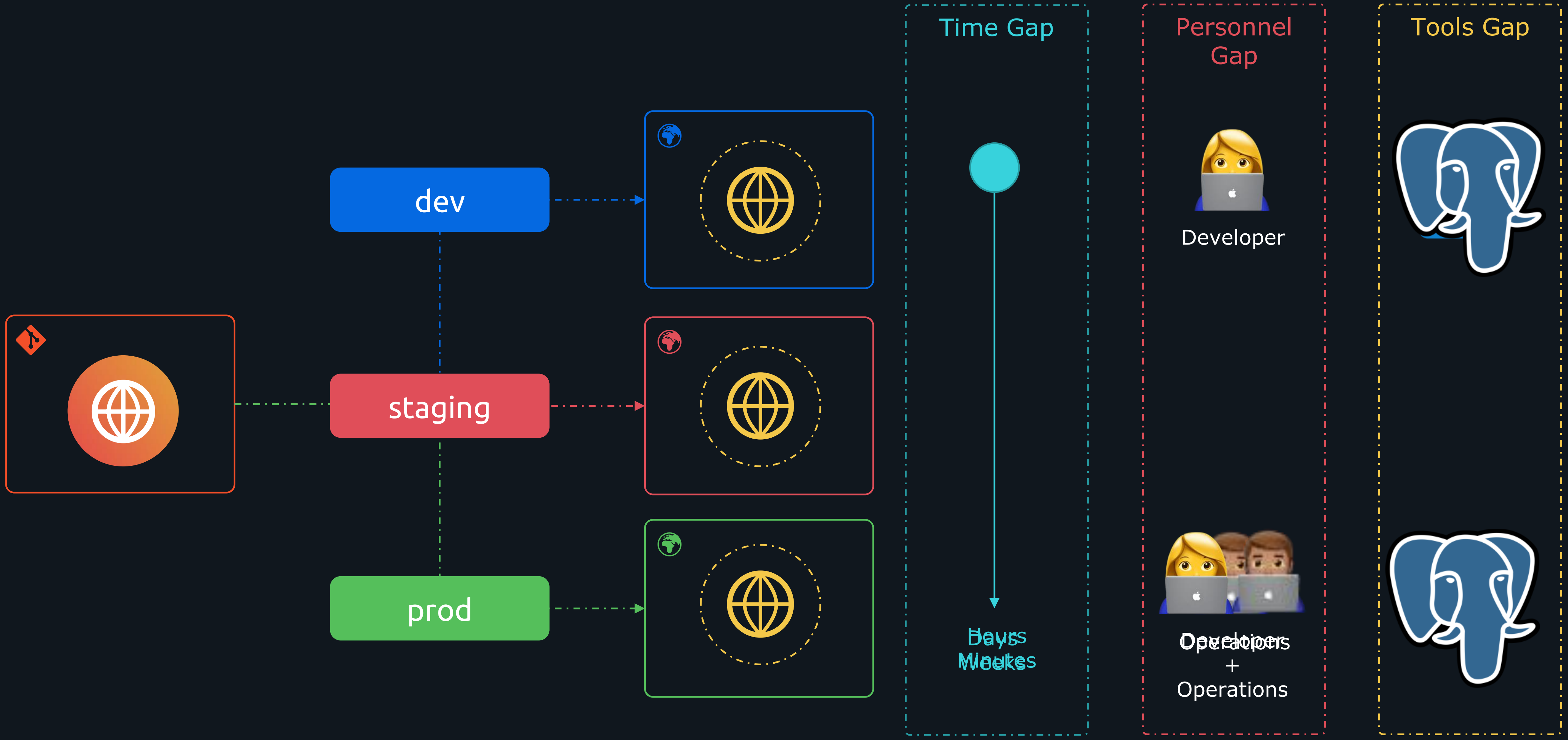

dev
staging
prod


dev
staging
prod
Time Gap
Days
Weeks
Personnel Gap
Developer
Operations
Tools Gap

☝ The twelve-factor app is designed for continuous deployment by keeping the gap between development and production small.

☝ The twelve-factor developer resists the urge to use different backing services between development and production.

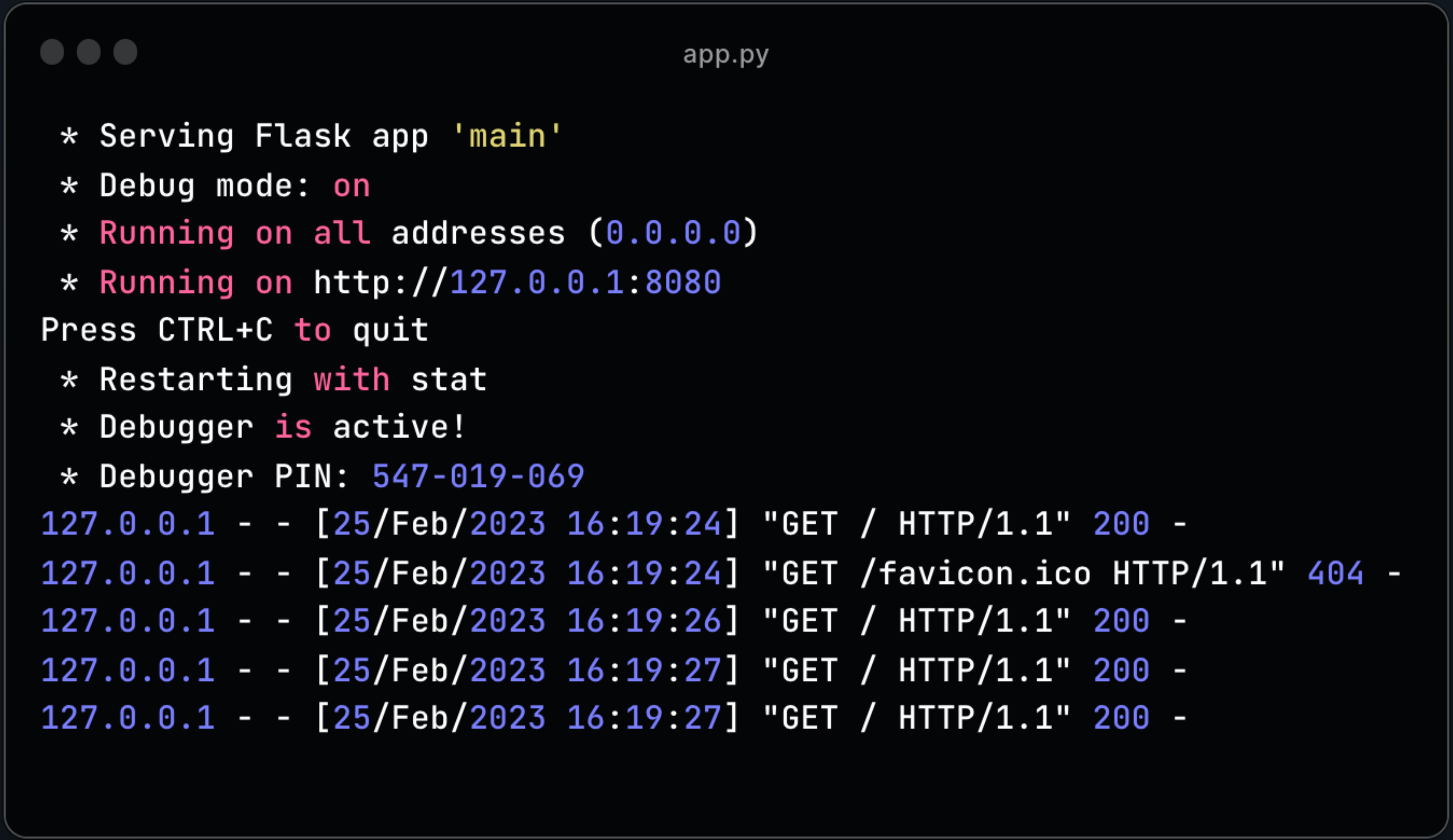
X
Dev/prod parity


dev
staging
prod
Time Gap
Hours
Days
Minutes
Weeks
Personnel Gap
Developer
Developer
Operations
+
Operations
Tools Gap

KodeKloud

KodeKloud

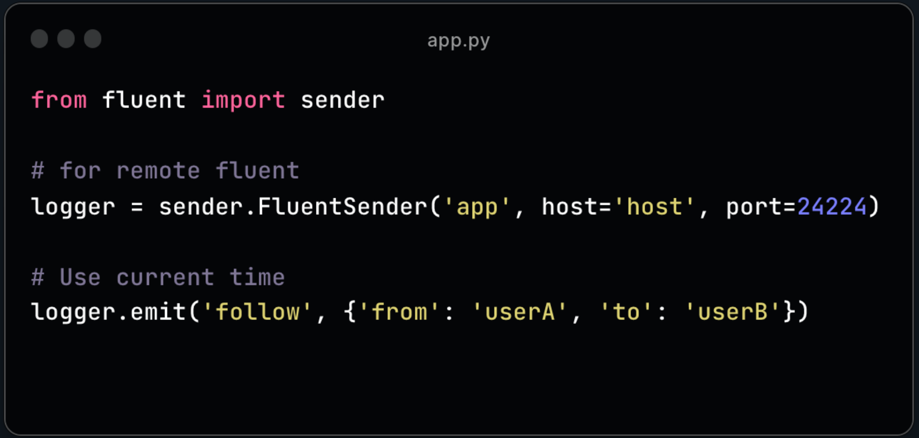
XI
Logs



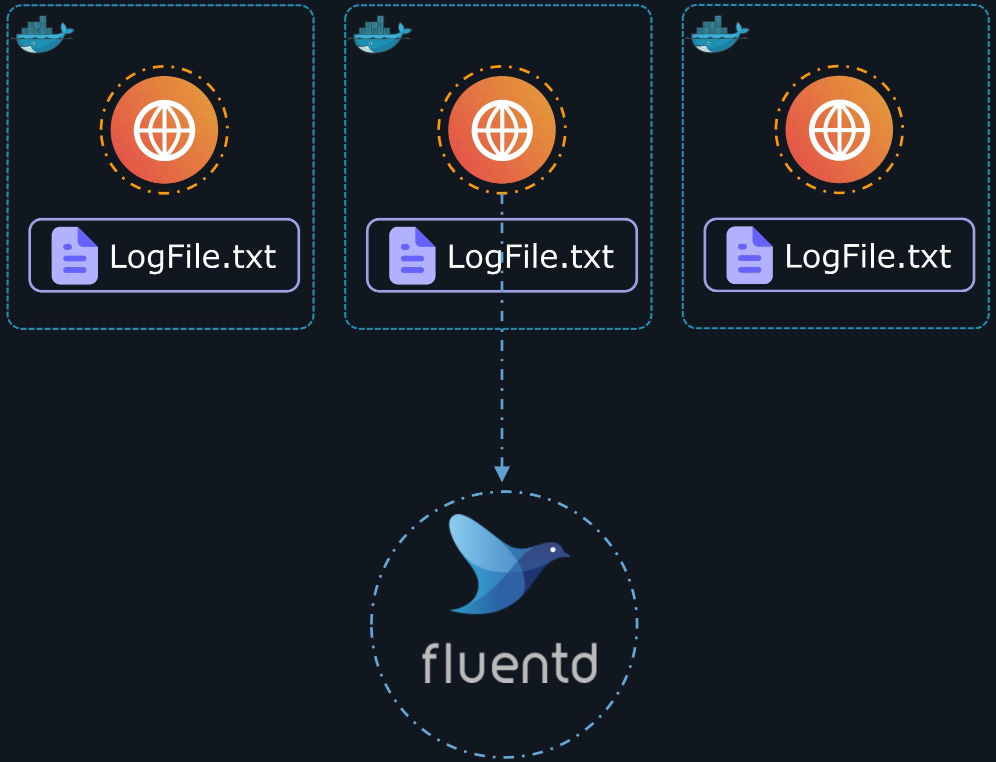
app.py

```
 * Serving Flask app 'main'
 * Debug mode: on
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8080
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 547-019-069
127.0.0.1 - - [25/Feb/2023 16:19:24] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [25/Feb/2023 16:19:24] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [25/Feb/2023 16:19:26] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [25/Feb/2023 16:19:27] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [25/Feb/2023 16:19:27] "GET / HTTP/1.1" 200 -
```

KodeKloud



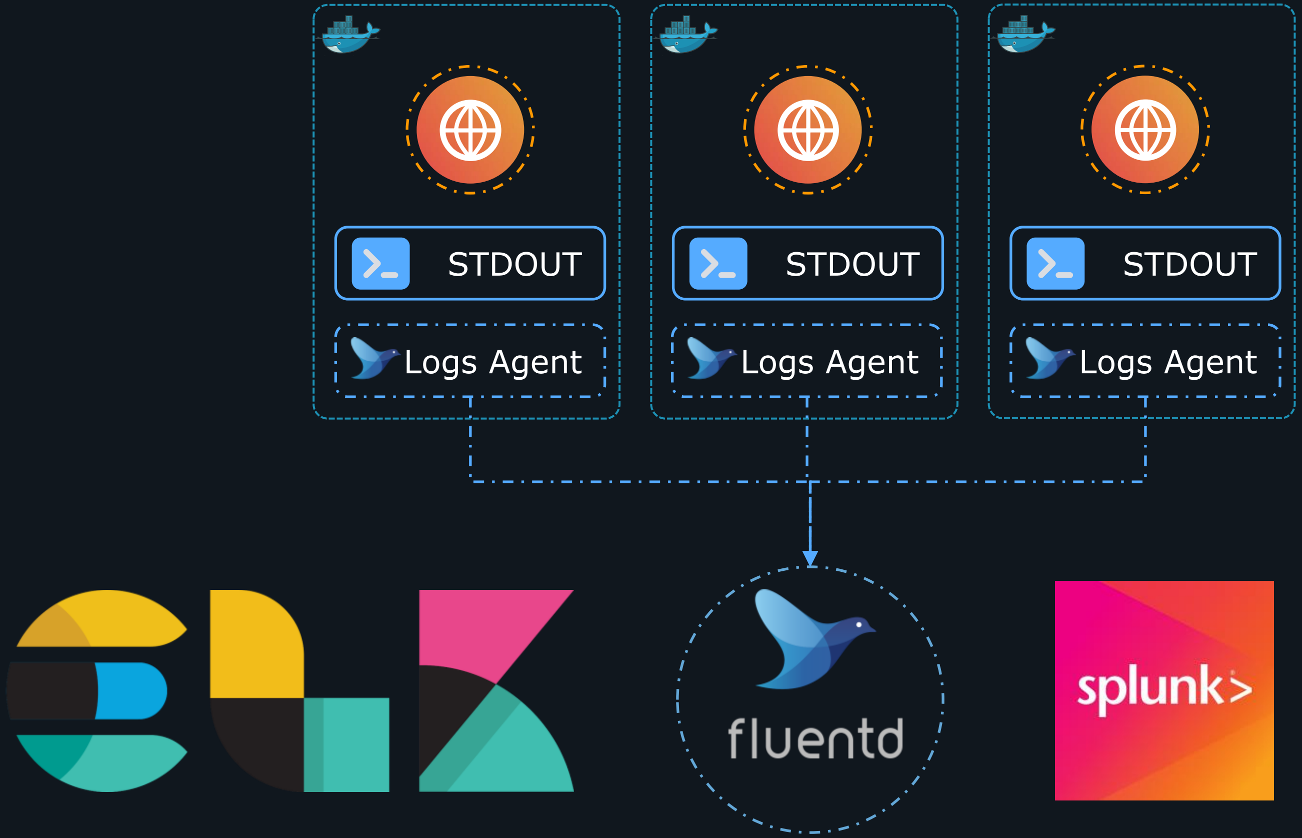
```python
from fluent import sender

# for remote fluent
logger = sender.FluentSender('app', host='host', port=24224)

# Use current time
logger.emit('follow', {'from': 'userA', 'to': 'userB'})
```



☝ A twelve-factor app never concerns itself with routing or storage of its output stream.

☝ Store logs in a centralized location in a structured format.

XI
Logs

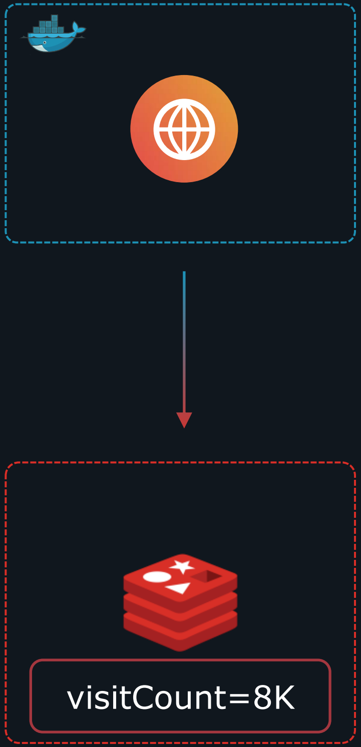

STDOUT
Logs Agent
STDOUT
Logs Agent
STDOUT
Logs Agent
fluentd
splunk>

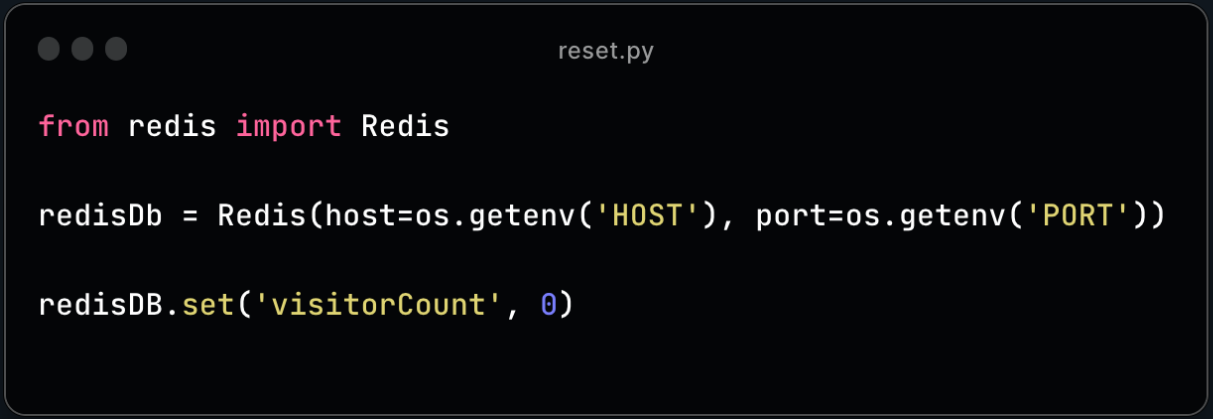
KodeKloud

KodeKloud

XII
Admin Processes

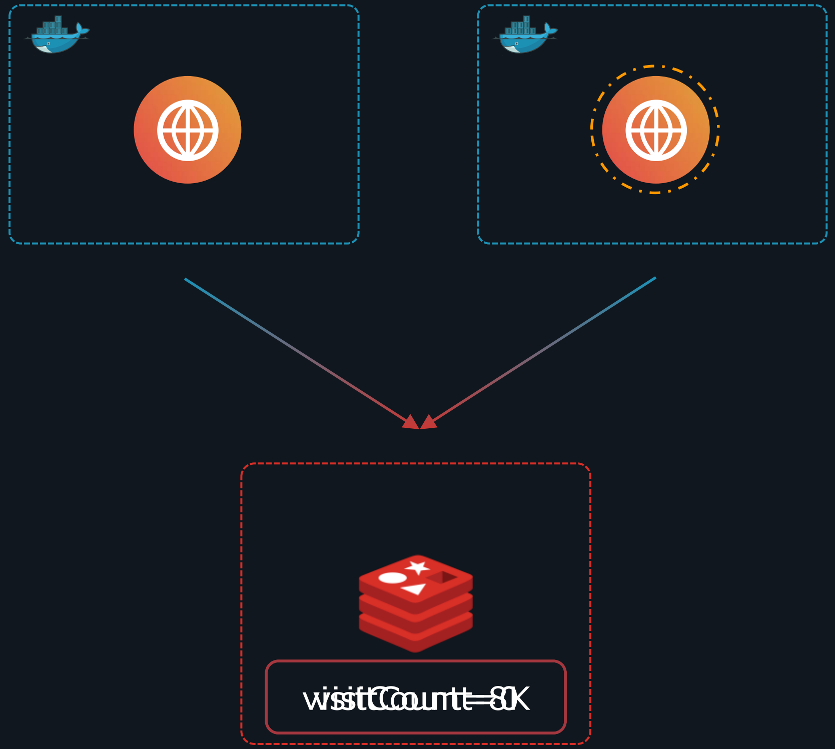
KodeKloud





reset.py

```python
from redis import Redis

redisDb = Redis(host=os.getenv('HOST'), port=os.getenv('PORT'))

redisDB.set('visitorCount', 0)
```



reset.py

```python
from redis import Redis

redisDb = Redis(host=os.getenv('HOST'), port=os.getenv('PORT'))

redisDB.set('visitorCount', 0)
```

KodeKloud

KodeKloud