

# - Swarm

YellowfinYellowfin4GBRAM80808090

Yellowfin YellowfinYellowfinYellowfin Yellowfin

Yellowfin

2Yellowfin

1. Yellowfin ()
2. web.xml (Yellowfin Yellowfin )
3. DockerSwarm
- 4.

```
version: '3'
services:
  yellowfin-multi-instance-prod:
    ports:
      - "8080:8080" # Maps Yellowfin running on port 8080 to Docker Swarm port 8080
    deploy:
      replicas: 1
      placement:
        constraints: [node.role == manager] # (Optional) Places the production instance on a manager node
    environment:
      # Required environment variables
      - JDBC_CLASS_NAME=INSERT_DATABASE_TYPE_1_HERE # Database driver class name
      - JDBC_CONN_URL=jdbc:INSERT_JDBC_CONNECTION_STRING_1_HERE # Database connection string
      - JDBC_CONN_USER=INSERT_DATABASE_USER_1_HERE # Username to use when accessing the database
      - JDBC_CONN_PASS=INSERT_JDBC_PASSWORD_1_HERE # Password for the database user
      - JDBC_CONN_ENCRYPTED=true # Flag for indicating if the database user's password supplied is
encrypted or not.
      - APP_MEMORY=8192 # The amount of memory in megabytes to assign to the Yellowfin Application.
    image: "yellowfinbi/yellowfin-app-only:<RELEASE_VERSION_GOES_HERE>" # Path to the app-only image of
Yellowfin

  yellowfin-multi-instance-dev:
    ports:
      - "8090:8080" # Maps Yellowfin running on port 8080 to Docker Swarm port 8090
    deploy:
      replicas: 1
      placement:
        constraints: [node.role == worker] # (Optional) Places the dev instance on a worker node
    environment:
      # Required environment variables
      - JDBC_CLASS_NAME=INSERT_DATABASE_TYPE_2_HERE # Database driver class name
      - JDBC_CONN_URL=jdbc:INSERT_JDBC_CONNECTION_STRING_2_HERE # Database connection string
      - JDBC_CONN_USER=INSERT_DATABASE_USER_2_HERE # Username to use when accessing the database
      - JDBC_CONN_PASS=INSERT_JDBC_PASSWORD_2_HERE # Password for the database user
      - JDBC_CONN_ENCRYPTED=true # Flag for indicating if the database user's password supplied is
encrypted or not.
      - APP_MEMORY=4096 # The amount of memory in megabytes to assign to the Yellowfin Application.
    image: "yellowfinbi/yellowfin-app-only:<RELEASE_VERSION_GOES_HERE>" # Path to the app-only image of
Yellowfin
```

5. **constraints:** (Yellowfin Docker Swarm Docker Swarm )
6. (Yellowfin web.xml )PostgreSQL

```
# Required environment variables
- JDBC_CLASS_NAME=org.postgresql.Driver # Database driver class name
- JDBC_CONN_URL=jdbc:postgresql://192.168.1.50/docker_multiple_yellowfin_instances # Database
connection string
- JDBC_CONN_USER=postgres # Username to use when accessing the database
- JDBC_CONN_PASS=bXF0oj5gnB1oRB1kZq5 # Password for the database user
- JDBC_CONN_ENCRYPTED=true # Flag for indicating if the database user's password supplied is encrypted
or not.
- APP_MEMORY=4096 # The amount of memory in megabytes to assign to the Yellowfin Application.
image: "yellowfinbi/yellowfin-app-only:9.6.0" # Path to the app-only image of Yellowfin
```

7. **yellowfin-multiple-instances.yml**YAML
8. Yellowfin  
docker stack deploy --compose-file yellowfin-multiple-instances.yml yellowfin
9. URL8080 () Yellowfin
10. Yellowfin (Yellowfin)
11. Yellowfin

-

DockerKubernetes

Docker

- [SwarmDocker](#)
  - [Sandbox](#)
  - 
  - 
  -
- [SwarmDocker](#)
  - [Sandbox - Swarm](#)
  - - Swarm
  - - Swarm
  - - Swarm

Kubernetes

- [Kubernetes](#)
  - [Sandbox -](#)
  - -
- [Kubernetes](#)
  - 
  -

Yellowfin

- 
- 

- [AWSYellowfin](#)

- [AzureYellowfin](#)
- [Google Cloud PlatformYellowfin](#)

- [Docker](#)
- [Kubernetes](#)
- [Yellowfin](#)

**Yellowfin**

[Yellowfin](#)

- 
- [Yellowfin](#)
- 

- 
- [Yellowfin](#)
- [LinuxYellowfin](#)
- [SAML Bridge](#)
-