



# DOCKER 13장 DOCKER HUB 사용하기



# Docker Hub

- <https://hub.docker.com>
- 공개 저장소(Public Repository)
  - *Docker 이미지를 다른 사람들과 공유*
  - *개수 제한 없이 무료 생성*
- 개인 저장소(Private Repository)
  - *Docker 이미지를 다른 사람들과 공유 하지 않음.*
  - *1개까지는 무료, 그 이상은 유료*

# Docker Hub

rocker/rstudio ☆

Last pushed: 4 hours ago

Repo Info

Tags

Dockerfile

Build Details

Short Description

RStudio Server image

Full Description

Using the rocker/rstudio container

Quickstart

```
docker run -d -p 8787:8787 rocker/rstudio
```

Visit `localhost:8787` in your browser and log in with username:password as `rstudio:rstudio`.

Notes:

- The `rocker/rstudio` is now part of the versioned stack. To request the `rstudio` image with a particular version of R, use the tag corresponding to the R version (e.g. `rocker/rstudio:3.4.0`, or `rocker/rstudio:devel`), or omit the tag to always get the latest stable release.
- Consequently, the Dockerfiles for these versions are [on rocker-versioned github repo](#). Dockerfiles in [rocker-org/rocker](#) are just aliases, except for `testing` tag, which builds on `r-base` image from this repo (and thus on `debian:testing`) rather than `rocker/r-ver`.

Docker Pull Command

```
docker pull rocker/rstudio
```


Owner

rocker

Source Repository

[rocker-org/rocker](#)

# Docker Hub 가입하기

 docker hub

Explore Help Sign in

## Docker Hub

Dev-test pipeline automation, 100,000+ free apps, public and private registries

New to Docker?  
Create your free Docker ID to get started.

Choose a Docker ID

Email address

Choose a password

Sign Up

© 2016 Docker Inc.



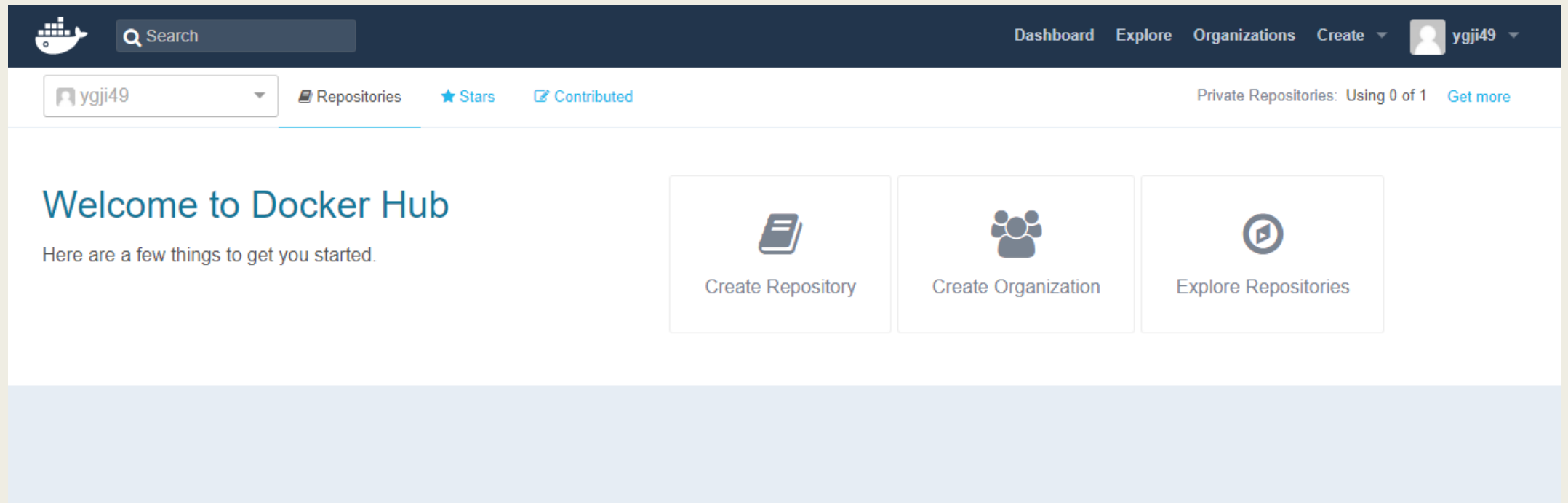
Please confirm your email address

You have created a Docker ID with the username: ygji49

Confirm Your Email

(This link will expire in 2 days.)

# Docker Hub 가입하기



# push 명령으로 이미지 올리기

- Docker Hub 공개 저장소(Public Repository)에 **example-nginx** 디렉토리를 생성
- Dockerfile 작성하고 올리기

```
~$ mkdir example-nginx
```

```
~$ cd example-nginx
```

```
~$ vi Dockerfile
```

```
FROM ubuntu:14.04
```

```
MAINTAINER Foo Bar <exampleuser@example.com>
```

```
RUN apt-get update
```

```
RUN apt-get install -y nginx
```

```
RUN echo "\ndaemon off;" >> /etc/nginx/nginx.conf
```

```
RUN chown -R www-data:www-data /var/lib/nginx
```

```
VOLUME ["/data", "/etc/nginx/site-enabled", "/var/log/nginx"]
```

```
WORKDIR /etc/nginx
```

```
CMD ["nginx"]
```

```
EXPOSE 80
```

```
EXPOSE 443
```

```
~/example-nginx$ docker build --tag ygji49/example-nginx:0.1 .
```

# push 명령으로 이미지 올리기

- Docker Hub에 이미지를 올리려면 이미지 이름을 <Docker Hub 사용자 계정>/<이미지 이름>:<태그> 형식

```
~/example-nginx$ docker login
```

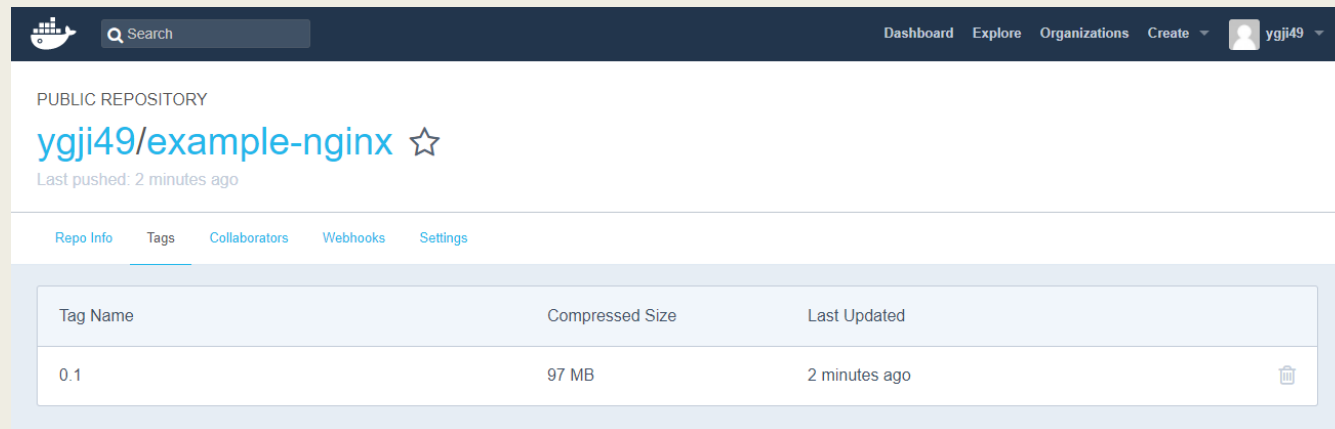
```
Username: ygji49
```

```
Password:
```

```
Email: ygji49@gmail.com
```

```
Login Succeeded
```

```
~/example-nginx$ docker push ygji49/example-nginx:0.1
```



The screenshot shows the Docker Hub interface for a public repository named 'ygji49/example-nginx'. The repository is marked as 'PUBLIC REPOSITORY' and has a star icon. It indicates 'Last pushed: 2 minutes ago'. Below the repository name, there are tabs for 'Repo Info', 'Tags', 'Collaborators', 'Webhooks', and 'Settings'. The 'Tags' tab is selected, displaying a table with the following data:

Tag Name	Compressed Size	Last Updated
0.1	97 MB	2 minutes ago

# Docker Hub Automated Build 활용하기

- Docker Hub는 GitHub과 BitBucket을 연동하여 이미지를 자동 빌드하는 기능을 제공한다

The image shows two screenshots of the Docker Hub interface. The top screenshot displays the 'Create' dropdown menu with 'Create Automated Build' highlighted. Below it, the 'Create Auto-build' card for GitHub is highlighted with a red box. The bottom screenshot shows the 'Users/Organizations' dropdown menu with 'biospin' highlighted, and the 'cloudnomad\_bio' repository selected in the list on the right, also highlighted with a red box.

**Top Screenshot: Docker Hub Dashboard**

- Navigation bar: Dashboard, Explore, Organizations, Create, ygji49
- Search bar: Search
- Link Accounts: GitHub (braveji18), Link Accounts
- Buttons: Create Repository, **Create Automated Build**, Create Organization
- Card: **Create Auto-build** (GitHub icon), Github
- Card: Link Account (Bitbucket icon), Bitbucket

**Bottom Screenshot: Docker Hub Search Results**

- Navigation bar: Dashboard, Explore, Organizations, Create, ygji49
- Search bar: Search
- Link Accounts: GitHub (braveji18), Link Accounts
- Users/Organizations dropdown: biospin, braveji18
- Repository list: all\_bio, angularjs, BigBio, biopy, biopyApp, bioR, biospark, biospin.github.io, **cloudnomad\_bio**



# Docker Hub Automated Build 활용하기

The screenshot shows the Docker Hub interface for the repository `ygji49/cloudnomad_bio`. The page is titled "PUBLIC | AUTOMATED BUILD" and includes a search bar and navigation links (Dashboard, Explore, Organizations, Create) in the top header. The repository name is displayed with a star icon and the text "Last pushed: never". Below the repository name, there are tabs for Repo Info, Tags, Dockerfile, Build Details, Build Settings, Collaborators, Webhooks, and Settings. The main content area is divided into two columns. The left column contains a "Short Description" field with the text "docker hub test" and a "Full Description" field with the text "Full description is empty for this repo.". The right column contains a "Docker Pull Command" field with the text "docker pull ygji49/cloudnomad\_bio", an "Owner" field with the user "ygji49", and a "Source Repository" field with the text "biospin/cloudnomad\_bio". The "Source Repository" field is highlighted with a red border.

PUBLIC | AUTOMATED BUILD

ygji49/cloudnomad\_bio ☆

Last pushed: never

Repo Info Tags Dockerfile Build Details Build Settings Collaborators Webhooks Settings

Short Description

docker hub test

Full Description

Full description is empty for this repo.

Docker Pull Command

docker pull ygji49/cloudnomad\_bio

Owner

ygji49

Source Repository

biospin/cloudnomad\_bio



# DOCKER 14장 DOCKER REMOTE API 사용하기



# Docker Engine SDKs and API

- Docker Remote API가 Docker Engine SDKs and API 로 변경됨.
- <https://docs.docker.com/develop/sdk/>
- Python SDK
  - *yum -y update*
  - *curl "https://bootstrap.pypa.io/get-pip.py" -o "get-pip.py"*
  - *python get-pip.py*
  - *Recommended: Run pip install docker.*
  - *pip install websocket-client*
  - *pip install docker-pycreds*

# Docker Engine SDKs and API

## ■ \$docker run 명령어를 python 또는 HTTP 로 실행

- `import docker`
- `client = docker.from_env()`
- `print client.containers.run("alpine", ["echo", "hello", "world"])`
  
- `$ curl --unix-socket /var/run/docker.sock -H "Content-Type: application/json" \`
- `-d '{"Image": "alpine", "Cmd": ["echo", "hello world"]}' \`
- `-X POST http://v1.24/containers/create`
- `{"Id":"1c6594faf5","Warnings":null}`
- `$ curl --unix-socket /var/run/docker.sock -X POST`  
`http://v1.24/containers/1c6594faf5/start`
- `$ curl --unix-socket /var/run/docker.sock -X POST`  
`http://v1.24/containers/1c6594faf5/wait`
- `{"StatusCode":0}`
- `$ curl --unix-socket /var/run/docker.sock`  
`"http://v1.24/containers/1c6594faf5/logs?stdout=1"`
- `hello world`

# Docker Engine SDKs and API

- Run a container in the background

- *import docker*
- *client = docker.from\_env()*
- *container = client.containers.run("bfirsh/reticulate-splines", detach=True)*
- *print container.id*

- List and manage containers

- *import docker*
- *client = docker.from\_env()*
- *for container in client.containers.list():*
- *print container.id*

# Docker Engine SDKs and API

- Stop all running containers
  - *import docker*
  - *client = docker.from\_env()*
  - *for container in client.containers.list():*
  - *container.stop()*
- Print the logs of a specific container
  - *import docker*
  - *client = docker.from\_env()*
  - *container = client.containers.get('f1064a8a4c82')*
  - *print container.logs()*

# Docker Engine SDKs and API

- List all images

- *import docker*
  - *client = docker.from\_env()*
  - *for image in client.images.list():*
  - *print image.id*

- Pull an image

- *import docker*
  - *client = docker.from\_env()*
  - *image = client.images.pull("alpine")*
  - *print image.id*