

# Install Oracle XE 11g dengan Docker (/~fahrinurul/blog/2014/01/30/install-oracle-xe-11g-dengan-docker/)

by fahrinurul - Thursday, January 30, 2014

Docker (<https://www.docker.io/>) adalah *tool* yang memudahkan dalam pengaturan Linux Containers (LXC). Pada dasarnya, LXC adalah *light weight virtual machines*. Kita dapat menjalankan *operating system* di dalam *host operating system*. Hanya saja, berbeda dengan *full virtual machine* seperti VirtualBox atau VMWare, LXC tidak berbasis infrastruktur sistem yang terpisah (dengan *resource* yang independen) tetapi masih bergantung pada *resource* milik *host* (Linux *kernel*) sehingga penggunaan *resource* lebih ringan. Di sini, dengan Docker, kita akan coba menjalankan *container* yang sudah terpasang Oracle XE 11g.



⬇️ Download *Project Directory* dengan *File Pendukung* : [oracle\\_docker\\_builder.zip](https://bitbucket.org/fahri_work/20140130_oracle_docker_builder/get/58448bee7839.zip) ([https://bitbucket.org/fahri\\_work/20140130\\_oracle\\_docker\\_builder/get/58448bee7839.zip](https://bitbucket.org/fahri_work/20140130_oracle_docker_builder/get/58448bee7839.zip))

## Table of Contents

FAQ

*Introduction*

*Install Docker*

*Project Directory*

*File Pendukung*

Dockerfile

start-oracle-xe.sh

listener.ora

supervisord.conf

*Building Image*

*Jalankan Container*

Tes Akses

🌐 Atur Proxy

## FAQ

1. A: Saya pemakai Windows dan Oracle sudah menyediakan installer Oracle XE ([download \(http://www.oracle.com/technetwork/database/database-technologies/express-edition/downloads/index.html\)](http://www.oracle.com/technetwork/database/database-technologies/express-edition/downloads/index.html)). Kenapa saya harus repot-repot memakai cara ini ?

Jika Anda adalah orang yang cinta dan senang dengan kerapian dan tidak ingin *mengotori* "Program Files" dengan *software* yang berbau pekerjaan sehingga sewaktu-waktu tidak dibutuhkan, aplikasi tinggal dihapus tanpa bekas maka cara ini cocok.

2. *B: Saya juga pemakai Windows seperti si A tapi kalau tujuannya hanya divirtualisasikan, saya bisa kok make pre-built developer vm (VirtualBox) berisi Oracle XE ([download \(http://www.oracle.com/technetwork/community/developer-vm/index.html\)](http://www.oracle.com/technetwork/community/developer-vm/index.html)) yang juga disediakan Oracle. Kenapa saya harus repot-repot memakai cara ini ?*

Seperti yang sudah disebutkan sebelumnya, dibanding *full virtual machine*, penggunaan *resource* LXC lebih kecil.

3. *C: Eh, tapi Docker kan cuman jalan native di Linux. Bagi yang memakai Windows tetap harus pasang full Linux VM (pake VirtualBox atau VMWare) yang isinya Docker. Sama aja donk ada overhead make prebuilt Oracle XE VM dengan memakai Docker di VM. Kenapa saya harus repot-repot memakai cara ini ?*

Betul, untuk pengguna Windows, memang tidak ada bedanya menggunakan *prebuilt* Oracle XE VM dengan Docker karena pemakai Docker di Windows juga perlu menggunakan *full* Linux VM. Hanya saja, cara ini lebih cocok jika Anda peduli dengan prinsip *separation of concern* dan sewaktu-waktu ada kebutuhan dimana satu VM/*container* hanya menyediakan satu *service*. Dengan 'modal' satu Linux VM + Docker tersebut, kita dapat membuat banyak *container* yang notabene lebih ringan. Sedangkan, jika mengandalkan VirtualBox (atau VMWare) tanpa Docker, pembuatan banyak VM perlu mempertimbangkan *resource* yang digunakan karena satu VM lebih haus *resource* dibanding satu *container* dengan menggunakan Docker.

4. *D: Saya pemakai Linux...*

Yak, tutorial ini memang cocok buat Anda.

## Introduction

*Container* pada Docker berbasis pada *pre-built image* yang tersedia di [Central Index Repository \(https://index.docker.io/\)](https://index.docker.io/). *Repository* menyediakan cukup banyak *image* yang sudah terpasang aplikasi/*service* di dalamnya. *Image* yang terpasang Oracle XE 11g sudah tersedia di *repository* pada *image* [kimh/oracledb](https://index.docker.io/u/kimh/oracledb/) (<https://index.docker.io/u/kimh/oracledb/>).

*User* tidak hanya dibatasi dengan menggunakan *image* yang tersedia di *repository* tetapi dapat juga menggunakan Docker sebagai *builder* untuk pembuatan *image* (<http://docs.docker.io/en/latest/use/builder/>). Di tutorial ini, kita tidak akan menjalankan *container* dengan menggunakan *image* [kimh/oracledb](https://index.docker.io/u/kimh/oracledb/) secara langsung tetapi kita akan jadikan *image* tersebut sebagai basis untuk membuat *image* baru dengan kriteria sebagai berikut :

- *Service* Oracle XE dapat diakses secara *remote* sehingga dengan *port forwarding*, *host* dapat mengakses Oracle XE di *container* seolah-olah miliknya (*local*).
- *Container* dapat diakses oleh *host* dengan SSH.



Bagi pengguna Windows, lakukan langkah-langkah di tutorial ini **di dalam Linux VM**. Jika Anda menemui kendala, jangan ragu untuk komplain di bagian komentar.

## Install Docker

- Windows : <http://docs.docker.io/en/latest/installation/windows/> (<http://docs.docker.io/en/latest/installation/windows/>)
- Ubuntu : <http://docs.docker.io/en/latest/installation/ubuntulinux/> (<http://docs.docker.io/en/latest/installation/ubuntulinux/>)

## Project Directory

Buat *folder* untuk menaruh Dockerfile dan *file* pendukung lainnya :

```
$ mkdir oracle_docker_builder
```

*File-file* pendukung yang akan dibuat di dalam folder `oracle_docker_builder` :

```
oracle_docker_builder
├─ Dockerfile
├─ listener.ora
├─ start-oracle-xe.sh
└─ supervisord.conf
```

## File Pendukung

### Dockerfile

- `Dockerfile` adalah 'resep' dalam pembuatan *image* di Docker. Isinya terdiri dari kumpulan instruksi dengan format `INSTRUCTION arguments`.  
(lihat penjelasan masing-masing instruksi yang didukung (<http://docs.docker.io/en/latest/use/builder/#id5>))

#### Dockerfile

```
FROM kimh/oracledb

RUN echo "root:root" | chpasswd

RUN apt-get update
RUN apt-get install -y supervisor

ADD supervisord.conf /etc/supervisor/conf.d/supervisord.conf
ADD listener.ora /u01/app/oracle/product/11.2.0/xe/network/admin/listener.ora

ADD start-oracle-xe.sh /start-oracle-xe.sh
```

*Image* yang akan dibuat berbasis `kimh/oracledb` yang sudah terpasang Oracle XE 11g.

Ubah *default password* untuk user `root` menjadi `root`.

*Override* konfigurasi `supervisord.conf`. ([lihat](#))

*Override* konfigurasi Oracle Database `listener.ora`. ([lihat](#))

*Custom script* untuk menjalankan *service* Oracle XE. ([lihat](#))

## start-oracle-xe.sh

- Sebelum *service* Oracle XE dijalankan, ada baiknya *environment variable* yang dibutuhkan oleh *service* diset terlebih dahulu.

### start-oracle-xe.sh

```
#!/bin/bash

source /u01/app/oracle/product/11.2.0/xe/bin/oracle_env.sh

/etc/init.d/oracle-xe start
```

## listener.ora

- Agar seolah-olah *host* yang menyediakan *service* Oracle, perlu ada *port forwarding* dari port 1521 (*host*) → port 1521 (*container*). *By default*, *service* Oracle dari *image* `kimh/oracledb` hanya bisa diakses *local* (dari sisi *container*). Padahal, agar *service* Oracle dapat diakses secara *port forwarding*, kita harus memastikan *service* Oracle dapat diakses dari luar (dalam hal ini, *private network* antara *host* dan *container*). Oleh karena itu, `listener.ora` perlu diset untuk dapat diakses secara *remote*.

### listener.ora

```
# listener.ora Network Configuration File:

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PLSExtProc)
      (ORACLE_HOME = /u01/app/oracle/product/11.2.0/xe)
    )
  )

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC_FOR_XE))
      (ADDRESS = (PROTOCOL = TCP)(HOST = 0.0.0.0)(PORT = 1521))
    )
  )

DEFAULT_SERVICE_LISTENER = (XE)
```

`0.0.0.0` menandakan *service* tersedia untuk semua *network interface* (dapat diakses oleh siapa saja). Sedangkan, jika menggunakan nilai *default* (`127.0.0.1`), akses hanya terbatas untuk *loopback interface* (hanya dapat diakses *local* dari sisi *container*).



Jika Anda *concern* dengan *security*, batasi *remote access service* Oracle dengan hanya menggunakan *subnet* dari *private network* antara *host* dan *container*.

supervisord.conf

- `supervisor` digunakan untuk mempermudah dalam menjalankan banyak program ketika *container* dijalankan. Dua program yang akan dijalankan : SSH *server* dan `start-oracle-xe.sh`

supervisord.conf

```
[supervisord]
nodaemon=true

[program:sshd]
command=/usr/sbin/sshd -D

[program:oracle-xe]
command=/bin/bash /start-oracle-xe.sh
```

Menjalankan SSH *server* sebagai *foreground process*.

Building Image

*Image* yang akan dibuat kita beri nama `phai/oracle-xe`.

```
$ cd oracle_docker_builder
$ sudo docker build -t phai/oracle-xe .
```

Jalankan Container

```
$ sudo docker run -d -p 0.0.0.0:1521:1521 -p 0.0.0.0:2202:22 -p 0.0.0.0:8082:8080 -t phai/oracle-xe /usr/bin/supervisord
```

*Container* dijalankan dengan *image* `phai/oracle-xe` dan langsung menjalankan `/usr/bin/supervisord`.

*Port forwarding* :

port host	→	port container	service
1521	→	1521	Oracle Net Listener
2202	→	22	SSH

port host	→	port container	service
8082	→	8080	Oracle XML DB HTTP Server

## Tes Akses

- **SSH ke *container*** (username : `root`, password : `root`)

Dari *host* :

```
$ ssh root@localhost -p 2202
```

- **SQL\*Plus**

Setelah masuk ke *container* :

```
# sqlplus SYS/oracle "AS SYSDBA"
```

- **ORACLE Application Express (Web)**

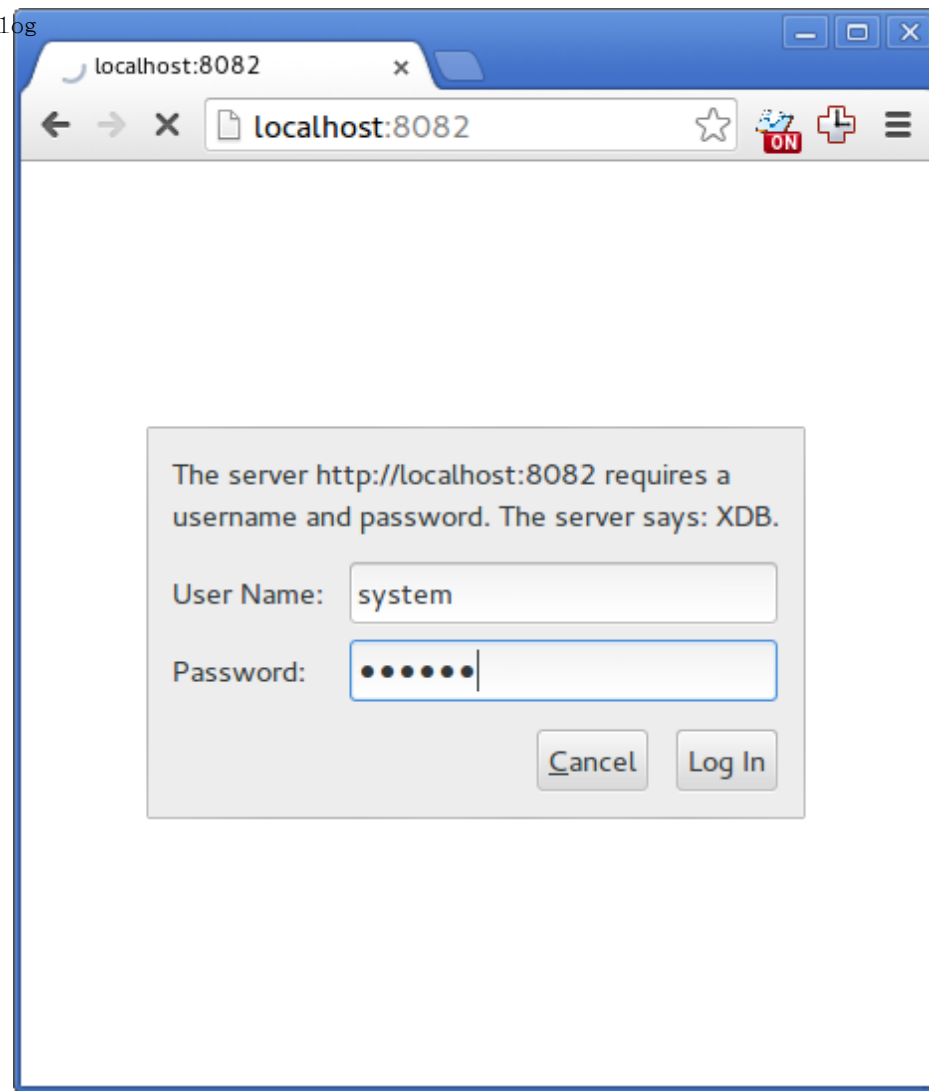
Dari *host*, akses dengan menggunakan *browser* alamat : [http://localhost:8082/apex/apex\\_admin](http://localhost:8082/apex/apex_admin) ([http://localhost:8082/apex/apex\\_admin](http://localhost:8082/apex/apex_admin)).

Untuk APEX :

- Username : `admin`
- Password : `oracle`

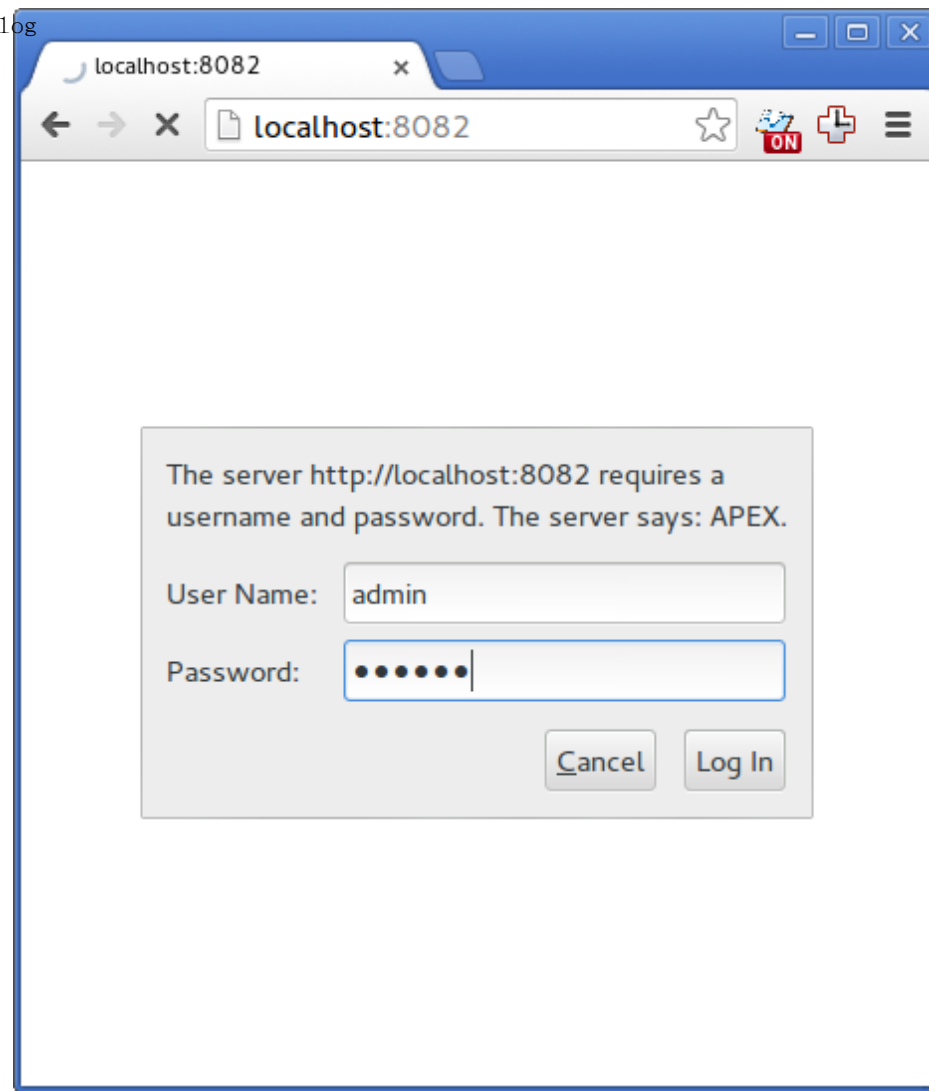
Ketika mengakses <http://localhost:8082> (<http://localhost:8082>), *browser* memunculkan *prompt* autentikasi seperti di bawah ini :

- XDB
  - User Name: `system`
  - Password: `oracle`



- o APEX

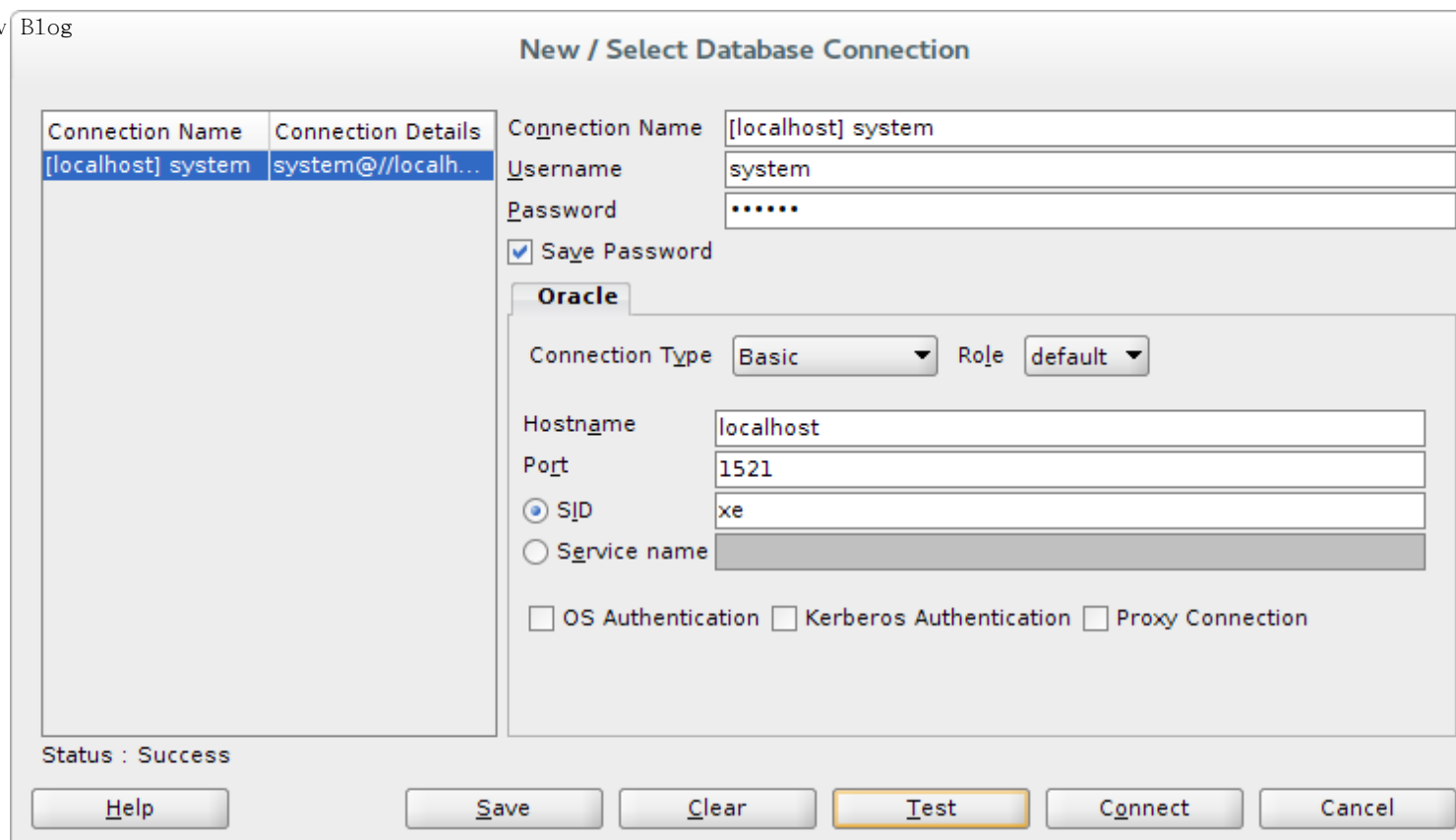
- User Name:
- Password:



- **Oracle SQL Developer**

- Username:
- Password:





## 🌐 Atur Proxy

Jika akses internet komputer perlu menggunakan *proxy*, lakukan langkah-langkah penyesuaian berikut ini :

1. Set *proxy* di `/etc/default/docker`

**`/etc/default/docker`**

```
# Docker Upstart and SysVinit configuration file

# Customize location of Docker binary (especially for development testing).
#DOCKER="/usr/local/bin/docker"

# Use DOCKER_OPTS to modify the daemon startup options.
#DOCKER_OPTS="-dns 8.8.8.8"

# If you need Docker to use an HTTP proxy, it can also be specified here.

export http_proxy=http://152.118.24.10:8080/
export https_proxy=http://152.118.24.10:8080/
```

2. Restart Docker

```
$ sudo service docker restart
```

3. Set `http_proxy` di `Dockerfile` dengan instruksi `ENV`

**Dockerfile**

```
FROM kimh/oracledb

RUN echo "root:root" | chpasswd

ENV http_proxy "http://152.118.24.10:8080/"
ENV https_proxy "http://152.118.24.10:8080/"

RUN apt-get update
RUN apt-get install -y supervisor

ADD supervisord.conf /etc/supervisor/conf.d/supervisord.conf
ADD listener.ora /u01/app/oracle/product/11.2.0/xe/network/admin/listener.ora

ADD start-oracle-xe.sh /start-oracle-xe.sh
```

4. *Unset* `http_proxy` di `start-oracle-xe.sh`

Ternyata, *service* Oracle tidak berjalan (karena SQL\*Plus gagal inisialisasi) jika ada *environment variable* `http_proxy`. Pesan *error* yang muncul seperti ini :

```
Error 46 initializing SQL*Plus
HTTP proxy setting has incorrect value
SP2-1502: The HTTP proxy server specified by http_proxy is not accessible
```

Pastikan *variable* `http_proxy` dan `https_proxy` di-*unset* sebelum menjalankan *service* Oracle :

**start-oracle-xe.sh**

```
#!/bin/bash

source /u01/app/oracle/product/11.2.0/xe/bin/oracle_env.sh

unset http_proxy
unset https_proxy

/etc/init.d/oracle-xe start
```

5. Ulangi langkah *Building Image* & *Jalankan Container*.

👉 [docker \(/~fahrinurul/blog/tag/docker/index.html\)](/~fahrinurul/blog/tag/docker/index.html), [oracle-xe \(/~fahrinurul/blog/tag/oracle-xe/index.html\)](/~fahrinurul/blog/tag/oracle-xe/index.html), [Oracle XE 11g \(/~fahrinurul/blog/tag/Oracle XE 11g/index.html\)](/~fahrinurul/blog/tag/Oracle XE 11g/index.html)

Sort by Best

Share Favorite



Start the discussion...

Be the first to comment.

Subscribe Add Disqus to your site

© Fahri Nurul Hidayat 2014.  
Modification of [Asciidoctor.org Project Site. \(https://github.com/asciidoctor/asciidoctor.org\)](https://github.com/asciidoctor/asciidoctor.org)  
Composed in AsciiDoc. Styled by Foundation. Baked with Awestruct.  
Content licensed under the CC BY 3.0 License.