

GETTING STARTED

login

- 기본 Syntax

```
$ cf login [-a API_URL] [-u USERNAME] [-p PASSWORD] [-o ORG] [-s SPACE]
```

- 설명

PCF 에 로그인 하기 위한 명령어

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|---|---------|
| -a API_URL | CLI 가 접속 하려는 PCF URL Ex) https://api.10.244.0.34.xip.io | X |
| -u USERNAME | PCF 에 접속하는 사용자 id | X |
| -p PASSWORD | PCF 에 접속하는 사용자 password | X |
| -o ORG | PCF 에 접속하는 사용자의 소속조직 명 | X |
| -s SPACE | PCF 에 접속하는 사용자의 소속조직 스페이스직 명 | X |

- **사용예시**

```
# 파라미터 지정한 경우
$ cf login --skip-ssl-validation -a https://api.10.244.0.34.xip.io -u admin -p admin -o pivotal -s development

# 파라미터 지정하지 않을 경우
$ cf login
API endpoint: https://api.10.244.0.34.xip.io

Email> admin

Password>
Authenticating...
OK

Targeted org pivotal

Select a space (or press enter to skip):
1. development
2. staged
3. oper

Space> 1
Targeted space development

API endpoint:  https://api.10.244.0.34.xip.io (API version: 2.29.0)
User:          admin
Org:           pivotal
Space:         development
```

logout

- 기본 Syntax

```
$ cf logout
```

- 설명

cf 에 logout 합니다.

- 파라미터

-없음

- 사용예시

```
$ cf logout
```

passwd

- 기본 Syntax

```
$ cf passwd
```

- 설명

PCF 사용자계정의 패스워드를 변경합니다.

- 파라미터

-없음

- **사용예시**

```
$ cf passwd
Current Password>

New Password>

Verify Password>
Changing password...
OK
Please log in again
```

target

- 기본 Syntax

```
$ cf target [-o ORG] [-s SPACE]
```

- 설명

로그인한 사용자가 사용할 Target 조직 및 스페이스 설정합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------------|---------|
| -o ORG | Target 조직 | X |
| -s SPACE | Target 스페이스 | X |

- 사용예시

```
# 파라미터 지정한 경우
$ cf target -o cf -s development
API endpoint:  https://api.10.244.0.34.xip.io (API version: 2.29.0)
User:          admin
Org:           cf
Space:         development

# 파라미터 지정하지 않은 경우(현재 Target 된 정보가 출력)
$ cf target
API endpoint:  https://api.10.244.0.34.xip.io (API version: 2.29.0)
```

| | |
|--------|-------------|
| User: | admin |
| Org: | cf |
| Space: | development |

api

- 기본 Syntax

```
$ cf api <URL>
```

- 설명

Target api 를 조회하거나 target api URL 을 설정합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------|----------------|---------|
| URL | Api Target URL | O |

- 사용예시

```
$ cf api --skip-ssl-validation api.10.244.0.34.xip.io
```

auth

- 기본 Syntax

```
$ cf auth <USERNAME> <PASSWORD>
```

- 설명

PCF login 시 로그인만 되며 스페이스, 타겟은 지정되지 않습니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|------------------|---------|
| USERNAME | 로그인 사용자 ID | O |
| PASSWORD | 로그인 사용자 PASSWORD | O |

- 사용예시

```
$ cf api --skip-ssl-validation api.10.244.0.34.xip.io
```


APPS

apps

- 기본 Syntax

```
$cf apps
```

- 설명

타겟 스페이스에 App 목록을 조회합니다.

- 파라미터

-없음

- 사용예시

```
$ cf apps
```

app

- 기본 Syntax

```
$cf app <APP_NAME>
```

- 설명

App 의 상태를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------|---------|
| APP_NAME | APP 명 | O |

```
- **사용예시**
```
$ cf app spring-music
```
```

push

- 기본 Syntax

```
$ cf push <APP_NAME> [-b BUILDPACK_NAME] [-c COMMAND] [-d DOMAIN] [-f MANIFEST_PATH] [-i NUM_INSTANCES] [-k DISK] [-m MEMORY] [-n HOST] [-p PATH] [-s STACK] [-t TIMEOUT] [--no-hostname] [--no-manifest] [--no-route] [--no-start]
```

- 설명

App 을 PCF 에 배포 하고 app 을 Start 합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|---|---------|
| APP_NAME | push 하는 App 명(지정하지 않으면 디렉토리명) | O |
| -b BUILDPACK | custom 빌드팩 URL ex) https://github.com/PCFRnD/egov-java-buildpack | X |
| -c COMMAND | App start command | X |
| -d DOMAIN | App 도메인 | X |
| -f MANIFEST_PATH | Manifest 파일 경로 | X |

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|--|---------|
| -i NUM_INSTANCES | App 인스턴스 갯수 | X |
| -m MEMORY | 인스턴스 메모리 용량 | X |
| -k DISK | 디스크 사용 용량 | X |
| -n HOST | 호스트명 ex) my-subdomain) | X |
| -p PATH | App 의 디렉토리 경로 또는 App file(zip,var 등)경로 | X |
| -s STACK | App 이 실행되는 운영체제 파일시스템(default: cflinuxfs2) | X |
| -t TIMEOUT | App 이 실행되는동안 CLI 가 대기하는 timeout 시간 | X |
| --no-hostname | App 에 root 도메인을 매핑 | X |
| --no-manifest | Manifest 파일을 무시합니다. | X |
| --no-route | Push 된 앱에 라우트 정보를 삭제하고 App 에 라우트 정보를 매핑하지 않음 | X |

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|---------------------------|---------|
| --no-start | App 을 push 하고 Start 하지 않음 | X |
| --random-route | App 에게 라우트 정보를 랜덤하게 생성 | X |

- **사용예시**

```
$ cf push spring-music
```

scale

- 기본 Syntax

```
$ cf scale <APP_NAME> [-i INSTANCES] [-k DISK] [-m MEMORY] [-f]
```

- 설명

App 의 메모리,디스크 크기 및 인스턴스 갯수를 조정합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|----------------|---------|
| APP_NAME | APP 명 | O |
| -i INSTANCES | 인스턴스 갯수 | X |
| -k DISK | 디스크 용량 | X |
| -m MEMORY | 메모리 용량 | X |
| -f | App 강제 restart | X |

- 사용예시

```
$ cf scale spring-music -i 2 -m 512m
```

delete

- 기본 Syntax

```
$ cf delete <APP_NAME> [--f] [--r]
```

- 설명

App 을 삭제합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|---------------------|---------|
| APP_NAME | APP 명 | O |
| --f | 확인 없이 App 삭제 | X |
| --r | App 에 매핑된 라우트 정보 삭제 | X |

- 사용예시

```
$ cf delete spring-music
```


rename

- 기본 Syntax

```
$ cf rename <APP_NAME> <NEW_APP_NAME>
```

- 설명

App 명을 변경합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|-------------|---------|
| APP_NAME | APP 명 | O |
| NEW_APP_NAME | 변경하려는 App 명 | O |

- 사용예시

```
$ cf rename spring-music new-spring-music
```

start

- 기본 Syntax

```
$ cf start <APP_NAME>
```

- 설명

App 을 기동 합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------|---------|
| APP_NAME | APP 명 | O |

- 사용예시

```
$ cf start spring-music
```

stop

- 기본 Syntax

```
$ cf stop <APP_NAME>
```

- 설명

App 을 중지 합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------|---------|
| APP_NAME | APP 명 | O |

- 사용예시

```
$ cf stop spring-music
```

restart, rs

- 기본 Syntax

```
$ cf restart <APP_NAME>
```

- 설명

App 을 재기동 합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------|---------|
| APP_NAME | APP 명 | O |

- 사용예시

```
$cf restart spring-music
```

restage, rg

- 기본 Syntax

```
$ cf restage <APP_NAME>
```

- 설명

App 을 restage 합니다.(환경변수 설정 또는 서비스 바인딩시 사용)

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------|---------|
| APP_NAME | APP 명 | O |

- 사용예시

```
$cf restage spring-music
```

restart-app-instance

- 기본 Syntax

```
$ cf restart-app-instance <APP_NAME> <INDEX>
```

- 설명

App 의 인스턴스중 특정 인스턴스를 재기동 합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|----------|---------|
| APP_NAME | APP 명 | O |
| INDEX | 인스턴스 인덱스 | O |

- 사용예시

```
$cf restart-app-instance spring-music 1
```

events

- 기본 Syntax

```
$ cf events <APP_NAME>
```

- 설명

App 에서 발생한 최근 Event 정보를 조회합니다. (start/stop/scale 등의 이력)

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------|---------|
| APP_NAME | APP 명 | O |

- 사용예시

```
$ cf events spring-music
```

files

- 기본 Syntax

```
$ cf files <APP_NAME> [PATH] [-i INSTANCE]
```

- 설명

App 의 file 및 디렉토리 목록을 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|--------------|---------|
| APP_NAME | APP 명 | O |
| PATH | APP 의 디렉토리 | X |
| -i INSTANCE | App 인스턴스 인덱스 | X |

- 사용예시

```
$ cf files spring-music
```


logs

- 기본 Syntax

```
$ cf logs <APP_NAME> [--recent]
```

- 설명

App 에서 발생한 로그를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|--------------|---------|
| APP_NAME | APP 명 | O |
| PATH | APP 의 디렉토리 | X |
| -i INSTANCE | App 인스턴스 인덱스 | X |

- 사용예시

```
$ cf logs spring-music
```

env, e

- 기본 Syntax

```
$ cf env <APP_NAME>
```

- 설명

App 의 환경변수를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------|---------|
| APP_NAME | APP 명 | O |

- 사용예시

```
$ cf env spring-music
```

set-env,se

- 기본 Syntax

```
$ cf set-env <APP_NAME> <ENV_VAR_NAME> <ENV_VAR_VALUE>
```

- 설명

App 의 환경변수를 설정합니다. (적용시 restage 필요)

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------------|------------------|---------|
| APP_NAME | APP 명 | O |
| ENV_VAR_NAME | App 의 환경변수 Key | O |
| ENV_VAR_VALUE | App 의 환경변수 Value | O |

- 사용예시

```
$ cf se spring-music author Jim
```

unset-env

- 기본 Syntax

```
$ cf unset-env <APP_NAME> <ENV_VAR_NAME>
```

- 설명

App 에 설정된 환경변수를 삭제합니다.(적용시 restage 필요)

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|----------------|---------|
| APP_NAME | APP 명 | O |
| ENV_VAR_NAME | App 의 환경변수 Key | O |

- 사용예시

```
$ cf unset-env spring-music author
```

stacks

- 기본 Syntax

```
$ cf stacks
```

- 설명

PCF 의 stack 목록(운영체제 파일시스템) 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf stacks
```

stack

• 기본 Syntax

```
$ cf stack <STACK_NAME> [--guid]
```

• 설명

PCF 의 stack 목록(운영체제 파일시스템) 목록을 조회합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-----------------|---------|
| APP_NAME | APP 명 | O |
| --guid | Stack guid 를 조회 | X |

• 사용예시

```
$ cf stack cflinuxfs2
```

copy-source

기본 Syntax

```
$ cf copy-source <SOURCE-APP> <TARGET-APP> [-o TARGET-ORG] [-s TARGET-SPACE] [--no-restart]
```

설명

App 의 소스를 다른 App 에 복사합니다. 파일이 덮어 쓰이지 않으면 자동 restart 합니다.

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------------|-----------------------|---------|
| SOURCE-APP | 원본 APP 명 | O |
| TARGET-APP | 소스가 복사될 대상 App 명 | X |
| -o TARGET-ORG | 타겟 조직 | O |
| -s TARGET-SPACE | 타겟 스페이스 | X |
| --no-restart | 소스 복사 후 restart 하지 않음 | X |

사용예시

```
$ cf copy-source spring-music another-music
```

create-app-manifest

- 기본 Syntax

```
$ cf create-app-manifest <APP_NAME> [-p /path/<app-name>-manifest.yml]
```

- 설명

App 의 manifest 파일을 생성합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------------|---------------------------------------|---------|
| SOURCE-APP | 원본 APP 명 | O |
| -p /path/.yml | 파일이 생성될 위치와 파일명(-p 를 사용하지 않으면 자동생성된다) | X |

- 사용예시

```
$ cf create-app-manifest spring-music -p ./spring-music-manifest.yml
```


SERVICES

marketplace,m

- 기본 Syntax

```
$ cf marketplace [-s SERVICE_NAME]
```

- 설명

cf 마켓플레이스에서 제공하는 서비스 목록을 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------------|-------------------|---------|
| -s SERVICE_NAME | 서비스의 plan 이 조회된다. | X |

- 사용예시

```
$ cf create-app-manifest spring-music -p ./spring-music-manifest.yml
```

services,s

- 기본 Syntax

```
$ cf services
```

- 설명

타겟 스페이스에 서비스 인스턴스 목록을 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------------|-------------------|---------|
| -s SERVICE_NAME | 서비스의 plan 이 조회된다. | X |

- 사용예시

```
$ cf create-app-manifest spring-music -p ./spring-music-manifest.yml
```

service

- 기본 Syntax

```
$ cf service <SERVICE_INSTANCE> [--guid]
```

- 설명

서비스 인스턴스의 정보를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|-------------------------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| --guid | 서비스 인스턴스의 Guid 를 조회합니다. | X |

- 사용예시

```
$ cf service spring-music-db
```

create-service

기본 Syntax

```
$ cf create-service <SERVICE> <PLAN> <SERVICE_INSTANCE> [-c PARAMETERS_AS_JSON] [-t TAGS]
```

설명

마켓플레이스에서 제공하는 서비스로 서비스 인스턴스를 만든다.

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------------------|--|---------|
| SERVICE | 마켓플레이스에 있는 서비스명 | O |
| PLAN | 서비스 플랜명 | O |
| SERVICE_INSTANCE | 만들 서비스 인스턴스명 | O |
| -c PARAMETERS_AS_JSON | 서비스 설정정보를 json 형태로 입력 Ex) -c '{"ram_gb":4}' | X |
| -t TAGS | 서비스 인스턴스 태그 | X |

사용예시

```
$ cf create-service spring-music-db silver p-mysql
```

update-service

• 기본 Syntax

```
$ cf update-service <SERVICE_INSTANCE> [-p NEW_PLAN] [-c PARAMETERS_AS_JSON] [-t TAGS]
```

• 설명

서비스 인스턴스를 수정합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------------------|--|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| -p NEW_PLAN | 서비스 플랜명 | O |
| -c PARAMETERS_AS_JSON | 서비스 설정정보를 json 형태로 입력 Ex) -c '{"ram_gb":4}' | O |
| -t TAGS | 서비스 인스턴스 태그 | X |

• 사용예시

```
$ cf update-service spring-music-db -p gold_plan
```

delete-service

- 기본 Syntax

```
$ cf delete-service SERVICE_INSTANCE [-f]
```

- 설명

서비스 인스턴스를 삭제합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|------------------------------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| -f | 삭제 확인 메시지 없이 서비스 인스턴스 삭제합니다. | X |

- 사용예시

```
$ cf delete-service spring-music-db
```

rename-service

- 기본 Syntax

```
$ cf rename-service <SERVICE_INSTANCE> <NEW_SERVICE_INSTANCE>
```

- 설명

서비스 인스턴스명을 수정합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------------|-----------------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| NEW_SERVICE_INSTANCE | 변경하려는 서비스 인스턴스명 | O |

- 사용예시

```
$ cf rename-service spring-music-db new_spring-music-db
```

create-service-key,csk

- 기본 Syntax

```
$ cf create-service-key <SERVICE_INSTANCE> <SERVICE_KEY> [-c PARAMETERS_AS_JSON]
```

- 설명

서비스 인스턴스의 key 를 생성합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------------------|-----------------------------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| SERVICE_KEY | 서비스 인스턴스 key 명 | O |
| -c PARAMETERS_AS_JSON | 서비스 인스턴스 설정(JSON Parameter) | X |

- 사용예시

```
$ cf create-service-key spring-music-db mykey -c '{"permissions":"read-only"}'
```


service-keys,sk

- 기본 Syntax

```
$ cf service-keys <SERVICE_INSTANCE>
```

- 설명

서비스 인스턴스의 key 목록을 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|-----------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |

- 사용예시

```
$ cf service-keys spring-music-db
```

service-key

- 기본 Syntax

```
$ cf service-key <SERVICE_INSTANCE> <SERVICE_KEY> [--guid]
```

- 설명

서비스 인스턴스의 key 의 상세정보를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|------------------------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| SERVICE_KEY | 서비스 인스턴스 key 명 | O |
| --guid | 서비스 인스턴스 guid 를 조회합니다. | X |

- 사용예시

```
$ cf service-key spring-music-db mykey
```

delete-service-key,dsk

• 기본 Syntax

```
$ cf delete-service-key <SERVICE_INSTANCE> <SERVICE_KEY> [-f]
```

• 설명

서비스 key 를 삭제합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|------------------------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| SERVICE_KEY | 서비스 인스턴스 key 명 | O |
| --guid | 서비스 인스턴스 guid 를 조회합니다. | X |

• 사용예시

```
$ cf delete-service-key spring-music-db mykey
```

bind-service,bs

- 기본 Syntax

```
$ cf bind-service <APP_NAME> <SERVICE_INSTANCE> [-c PARAMETERS_AS_JSON]
```

- 설명

App 과 서비스 인스턴스를 바인딩합니다.
 - 서비스 인스턴스와 APP 을 바인딩해야 App 에서 서비스 사용가능

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------------------|-----------------------|---------|
| APP_NAME | APP 명 | O |
| SERVICE_INSTANCE | 서비스 인스턴스 명 | O |
| -c PARAMETERS_AS_JSON | 바인딩 설정 파라미터 (json 형태) | X |

- 사용예시

```
$ cf bind-service spring-music spring-music-db -c '{"permissions":"read-only"}'
$ cf bind-service spring-music spring-music-db -c ~/workspace/tmp/instance_config.json
```

unbind-service,us

• 기본 Syntax

```
$ cf unbind-service <APP_NAME> <SERVICE_INSTANCE>
```

• 설명

App 과 서비스 인스턴스를 언바인딩합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|------------|---------|
| APP_NAME | 서비스 인스턴스명 | O |
| SERVICE_INSTANCE | 서비스 인스턴스 명 | O |

• 사용예시

```
$ cf unbind-service spring-music spring-music-db
```

create-user-provided-service,cups

• 기본 Syntax

```
$ cf create-user-provided-service <SERVICE_INSTANCE> [-p CREDENTIALS] [-l SYSLOG-DRAIN-URL]
```

• 설명

Market place 에서 제공하는 서비스를 사용하지 않고 사용자가 별도의 서비스를 구성하여 APP 과 바인딩합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------------------|------------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| -p CREDENTIALS | 서비스 인스턴스 명 | X |
| -l SYSLOG-DRAIN-URL | 서비스 인스턴스 명 | X |

• 사용예시

```
$ cf create-user-provided-service spring-music-db -p '{"username":"admin","password":"pa55woRD"}'
```

update-user-provided-service,uups

• 기본 Syntax

```
$ cf update-user-provided-service <SERVICE_INSTANCE> [-p CREDENTIALS] [-l SYSLOG-DRAIN-URL]
```

• 설명

user-provided service instance 정보를 수정합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------------------|------------|---------|
| SERVICE_INSTANCE | 서비스 인스턴스명 | O |
| -p CREDENTIALS | 서비스 인스턴스 명 | X |
| -l SYSLOG-DRAIN-URL | 서비스 인스턴스 명 | X |

• 사용예시

```
$ cf update-user-provided-service spring-music-db -p '{"username":"admin","password":"pa55woRD"}'
```

ORGS

orgs,o

- 기본 Syntax

```
$ cf orgs
```

- 설명

조직정보 목록을 조회합니다...

- 파라미터
- 없음
- 사용예시

```
$ cf orgs
```


org

- 기본 Syntax

```
$ cf org <ORG_NAME>
```

- 설명

조직 상세 정보를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-------------------|---------|
| ORG_NAME | 조직명 | O |
| --guid | 조직의 guid 를 조회합니다. | X |

- 사용예시

```
$ cf org cf
```

create-org,co

- 기본 Syntax

```
$ cf create-org <ORG_NAME> [-q QUOTA_NAME]
```

- 설명

조직정보를 생성합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------------|----------------|---------|
| ORG_NAME | 조직명 | O |
| -q QUOTA_NAME | 조직에게 할당할 quota | X |

- 사용예시

```
$cf create-org test -q default
```

delete-org

- 기본 Syntax

```
$ cf delete-org <ORG_NAME> [-f]
```

- 설명

조직정보 목록을 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|----------------------|---------|
| ORG_NAME | 조직명 | O |
| -f | 확인메시지 없이 조직정보 삭제합니다. | X |

- 사용예시

```
$ cf delete-org cf -f
```

rename-org

- 기본 Syntax

```
$ cf rename-org <ORG_NAME> <NEW_ORG_NAME>
```

- 설명

조직명을 변경합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|---------|---------|
| ORG_NAME | 조직명 | O |
| NEW_ORG_NAME | 변경할 조직명 | O |

- 사용예시

```
$ cf rename cf new-cf
```

SPACES

spaces

- 기본 Syntax

```
$ cf spaces
```

- 설명

스페이스 목록을 가져온다.

- 파라미터
- 없음
- 사용예시

```
$ cf spaces
```

space

- 기본 Syntax

```
$ cf space <SPACE_NAME>
```

- 설명

스페이스 상세정보를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------|-------|---------|
| SPACE_NAME | 스페이스명 | O |

- 사용예시

```
$ cf space development
```

create-space

- 기본 Syntax

```
$ cf create-space <SPACE_NAME> [-o ORG_NAME] [-q SPACE-QUOTA-NAME]
```

- 설명

스페이스 정보를 생성합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------------------|-------------------|---------|
| SPACE_NAME | 스페이스명 | O |
| -o ORG_NAME | 스페이스에 매핑될 조직명 | X |
| -q SPACE-QUOTA-NAME | 스페이스에 할당될 QUOTA 명 | X |

- 사용예시

```
$ cf create-space -o cf -q cf-space-quota
```

delete-space

- 기본 Syntax

```
$ cf delete-space <SPACE_NAME> [-f]
```

- 설명

스페이스정보를 삭제합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------|-------------------------|---------|
| SPACE_NAME | 스페이스명 | O |
| -f | 삭제 확인메시지 없이 스페이스 삭제합니다. | X |

- 사용예시

```
$ cf delete-space development
```


rename-space

- 기본 Syntax

```
$ cf rename-space <SPACE_NAME> <NEW_SPACE_NAME>
```

- 설명

스페이스 명을 변경합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|-------------------------|---------|
| SPACE_NAME | 스페이스명 | O |
| NEW_SPACE_NAME | 삭제 확인메시지 없이 스페이스 삭제합니다. | O |

- 사용예시

```
$ cf rename-space development new_development
```

DOMAINS

domains

- 기본 Syntax

```
$ cf domains
```

- 설명

도메인 정보 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf domains
```

create-domain

- 기본 Syntax

```
$ cf create-domain <ORG_NAME> <DOMAIN>
```

- 설명

도메인 정보 목록을 생성합니다. 생성된 도메인은 설정된 조직에서 사용가능하다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|------|---------|
| ORG_NAME | 조직명 | O |
| DOMAIN | 도메인명 | O |

- 사용예시

```
$ cf create-domain cf-org cf.or.kr
```

delete-domain

• 기본 Syntax

```
$ cf delete-domain <DOMAIN> [-f]
```

• 설명

도메인 정보를 삭제합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------|-------------------------|---------|
| DOMAIN | 도메인명 | O |
| -f | 삭제 확인메시지 없이 도메인을 삭제합니다. | X |

• 사용예시

```
$ cf delete-domain cf.or.kr
```

create-shared-domain

• 기본 Syntax

```
$ cf create-shared-domain <DOMAIN>
```

• 설명

공유 도메인정보를 생성한다

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------|-----|---------|
| DOMAIN | 조직명 | O |

• 사용예시

```
$ cf create-shared-domain cf.or.kr
```

delete-shared-domain

- 기본 Syntax

```
$ cf delete-shared-domain <DOMAIN> [-f]
```

- 설명

공유 도메인정보를 삭제합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------|-------------------------|---------|
| DOMAIN | 조직명 | O |
| -f | 삭제 확인메시지 없이 도메인을 삭제합니다. | X |

- 사용예시

```
$ cf delete-shared-domain cf.or.kr
```

REOUTES

routes, r

- **기본 Syntax**

```
$ cf routes
```

- **설명**

현재 조직/스페이스에 존재하는 라우트 정보목록을 조회합니다.

- **파라미터**

- 없음

- **사용예시**

```
$ cf routes
```

create-route

• 기본 Syntax

```
$ cf create-route <SPACE_NAME> <DOMAIN> [-n HOSTNAME]
```

• 설명

공유 도메인정보를 삭제합니다...

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|--|---------|
| SPACE_NAME | 스페이스명 | O |
| DOMAIN | 삭제 확인메시지 없이 공유 도메인을 삭제합니다. - 도메인 정보가 입력되어있어야 합니다. | O |
| -n HOSTNAME | 호스트 명 | X |

• 사용예시

```
$ cf create-route development cf.or.kr
```


update-route

- 기본 Syntax

```
$ cf update-route <SPACE_NAME> <DOMAIN> [-n HOSTNAME]
```

- 설명

공유 도메인정보를 삭제합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|--|---------|
| SPACE_NAME | 스페이스명 | O |
| DOMAIN | 삭제 확인메시지 없이 공유 도메인을 삭제합니다. - 도메인 정보가 입력되어있어야 합니다. | O |
| -n HOSTNAME | 호스트 명 | X |

- 사용예시

```
$ cf update-route development cf.or.kr
```

check-route

- 기본 Syntax

```
$ cf check-route <HOST> <DOMAIN>
```

- 설명

라우트 정보가 존재하는지 체크합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------|----------------------------|---------|
| HOST | 호스트 명 | O |
| DOMAIN | 삭제 확인메시지 없이 공유 도메인을 삭제합니다. | O |

- 사용예시

```
$ cf check-route spring-music cf.or.kr
```

map-route

- 기본 Syntax

```
$ cf map-route <APP_NAME> <DOMAIN> [-n HOSTNAME]
```

- 설명

App 에게 URL route 정보를 할당합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|-----------------|---------|
| APP_NAME | App 명 | O |
| DOMAIN | App 에게 할당할 도메인 | O |
| -n HOSTNAME | App 에게 할당할 Host | X |

- 사용예시

```
$ cf map-route spring-music cf.or.kr -n test
```

unmap-route

- 기본 Syntax

```
$ cf unmap-route <APP_NAME> <DOMAIN> [-n HOSTNAME]
```

- 설명

App 에게 URL route 정보를 삭제합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|-----------------|---------|
| APP_NAME | App 명 | O |
| DOMAIN | App 에게 할당할 도메인 | O |
| -n HOSTNAME | App 에게 할당할 Host | X |

- 사용예시

```
$ cf unmap-route spring-music cf.or.kr -n spring-music
```

delete-route

• 기본 Syntax

```
$ cf delete-route <DOMAIN> [-n HOSTNAME] [-f]
```

• 설명

App 에게 URL route 정보를 삭제합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|----------------------------|---------|
| DOMAIN | App 에게 할당할 도메인 | O |
| -n HOSTNAME | App 에게 할당할 Host | X |
| -f | 삭제 확인메시지 없이 라우트 정보를 삭제합니다. | X |

• 사용예시

```
$ cf delete-route spring-music cf.or.kr -n spring-music
```

delete-orphaned-routes

기본 Syntax

```
$ cf delete-orphaned-routes [-f]
```

설명

App 에 매핑되지 않은 라우트 정보를 모두 삭제한다

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------|----------------------------|---------|
| -f | 삭제 확인메시지 없이 라우트 정보를 삭제합니다. | X |

사용예시

```
$ cf delete-orphaned-routes
```

BUILDPACKS

buildpacks

- 기본 Syntax

```
$ cf buildpacks
```

- 설명

빌드팩 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf buildpacks
```

create-buildpack

• 기본 Syntax

```
$ cf create-buildpack <BUILDPACK> <-p PATH> <-i POSITION> [--enable|--disable]
```

• 설명

빌드팩을 생성합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|---|---------|
| BUILDPACK | 빌드팩명 | O |
| -p PATH | 빌드팩 경로 | O |
| -i POSITIONE | 빌드팩 auto-detection 동안 빌드팩 체크 순서 ex)1.2.3 | O |
| --enable | 스태이징시 사용 | X |
| --disable | 스태이징시 미사용 | X |

• 사용예시

```
$ cf create-buildpack egov-buildpack ~/workspace/buildpack/egov -i 1
```


update-buildpack

- 기본 Syntax

```
$ cf update-buildpack <BUILDPACK> [-p PATH] [-i POSITION] [--enable|--disable] [--lock|--unlock]
```

- 설명

빌드팩 정보를 수정합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|---|---------|
| BUILDPACK | 빌드팩명 | O |
| -p PATH | 빌드팩 경로 | O |
| -i POSITIONE | 빌드팩 auto-detection 동안 빌드팩 체크 순서 ex)1.2.3 | O |
| --enable | 스태이징시 사용 | X |
| --disable | 스태이징시 미사용 | X |

- 사용예시

```
$ cf create-buildpack egov-buildpack ~/workspace/buildpack/egov -i 1
```

delete-buildpack

- 기본 Syntax

```
$ cf delete-buildpack <BUILDPACK_NAME> [-f]
```

- 설명

빌드팩을 삭제합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------|------------------------|---------|
| BUILDPACK | 빌드팩명 | O |
| -f | 삭제 확인메시지 없이 빌드팩 정보를 삭제 | X |

- 사용예시

```
$ cf delete-buildpack egov-buildpack
```

USER ADMIN

create-user

- 기본 Syntax

```
$ cf create-user <USERNAME> <PASSWORD>
```

- 설명

새로운 사용자 계정을 생성합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|--------|---------|
| USERNAME | 사용자 ID | O |
| PASSWORD | 패스워드 | O |

- 사용예시

```
$ cf create-user cfuser userpassword
```

delete-user

• 기본 Syntax

```
$ cf delete-user <USERNAME> [-f]
```

• 설명

새로운 사용자 계정을 삭제합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|------------------------|---------|
| USERNAME | 사용자 ID | O |
| -f | 삭제 확인메시지 없이 사용자 정보를 삭제 | X |

• 사용예시

```
$ cf delete-user cfuser
```

org-users

- 기본 Syntax

```
$ cf org-users <ORG_NAME>
```

- 설명

조직에 소속된 사용자를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-----|---------|
| ORG_NAME | 조직명 | O |

- 사용예시

```
$ cf org-users cforg
```

set-org-role

• 기본 Syntax

```
$ cf set-org-role <USERNAME> <ORG> <ROLE>
```

• 설명

사용자에게 특정조직의 role 을 설정합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|---|---------|
| USERNAME | 사용자명 | O |
| ORG | 조직명 | O |
| ROLE | 역할명 - OrgManager : 사용자 관리 및 plan 설정/변경 권한 - BillingManager : 빌링계정 및 과금정보 생성 및 관리 - OrgAuditor : 조직 quota 사용률 및 사용자 role 을 조회 | O |

• 사용예시

```
$ cf set-org-role cfuser cforg OrgManager
```

unset-org-role

• 기본 Syntax

```
$ cf unset-org-role <USERNAME> <ORG> <ROLE>
```

• 설명

사용자에게 특정조직의 role 을 설정을 해제합니다..

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|---|---------|
| USERNAME | 사용자명 | O |
| ORG | 조직명 | O |
| ROLE | 역할명 - OrgManager : 사용자 관리 및 plan 설정/변경 권한 - BillingManager : 빌링계정 및 과금정보 생성 및 관리 - OrgAuditor : 조직 quota 사용률 및 사용자 role 을 조회 | O |

• 사용예시

```
$ cf unset-org-role cfuser cforg OrgManager
```

space-users

- 기본 Syntax

```
$ cf space-users <ORG> <SPACE>
```

- 설명

조직의 스페이스에 할당된 사용자 목록정보를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------|-------|---------|
| ORG | 조직명 | O |
| SPACE | 스페이스명 | O |

- 사용예시

```
$ cf space-users development
```


set-space-role

• 기본 Syntax

```
$ cf set-space-role <USERNAME> <ORG> <SPACE> <ROLE>
```

• 설명

사용자에게 조직의 스페이스에 role 을 할당합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|--|---------|
| USERNAME | 사용자명 | O |
| ORG | 조직명 | O |
| SPACE | 스페이스명 | O |
| ROLE | 역할명 - SpaceManager: 스페이스의 관리자로 스페이스 내의 사용자 계정 관리 및 인스턴스 수, 서비스 바인딩 상태 및 스페이스 내의 리소스 상태를 조회 및 변경 - SpaceDeveloper: 서비스 관리로 App 배포 - SpaceAuditor: 서비스 관리로 App 을 배포 | O |

• 사용예시

```
$ cf set-space-role cfuser cforg development OrgManager
```

unset-space-role

• 기본 Syntax

```
$ cf unset-space-role <USERNAME> <ORG> <SPACE> <ROLE>
```

• 설명

사용자에게 조직의 스페이스에 role 을 회수합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|---|---------|
| USERNAME | 사용자명 | O |
| ORG | 조직명 | O |
| SPACE | 스페이스명 | O |
| ROLE | 역할명 - SpaceManager: 스페이스의 관리자로 스페이스 내의 사용자 계정 관리 및 인스턴스 수, 서비스 바인딩 상태 및 스페이스 내의 리소스 상태를 조회. - SpaceDeveloper: 서비스 관리로 App 배포 - SpaceAuditor: 스페이스 내의 서비스 바인딩, 인스턴스 수, app 사용률등을 조회 | O |

• 사용예시

```
$ cf unset-space-role cfuser cforg development OrgManager
```

ORG ADMIN

quotas

- 기본 Syntax

```
$ cf quotas
```

- 설명

Quota 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf quotas
```

quota

- 기본 Syntax

```
$ cf quota <QUOTA>
```

- 설명

Quota 의 상세정보를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|---------|---------|
| USERNAME | QUOTA 명 | O |

- 사용예시

```
$ cf quota cf-quota
```

set-quota

- 기본 Syntax

```
$ cf set-quota <ORG> <QUOTA>
```

- 설명

조직에게 QUOTA 를 할당합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------|---------|---------|
| ORG | 직명 | O |
| QUOTA | QUOTA 명 | O |

- 사용예시

```
$ cf set-quota cf-quota
```

create-quota

- 기본 Syntax

```
$ cf create-quota <QUOTA> [-m TOTAL_MEMORY] [-i INSTANCE_MEMORY] [-r ROUTES] [-s SERVICE_INSTANCES] [--allow-paid-service-plans]
```

- 설명

Quota 정보를 생성합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------------------|--|---------|
| QUOTA | QUOTA 명 | O |
| -m TOTAL_MEMORY | 메모리 할당량 Ex) 1024M, 1G, 10G | X |
| -i INSTANCE_MEMORY | App instance 가 가질수 있는 최대할당량 (-1 은 무한대) Ex) 1024M, 1G, 10G | X |
| -r ROUTES | 최대 라우트 수 | X |
| -s SERVICE_INSTANCES | 최대 서비스 인스턴스 수 | X |
| --allow-paid-service-plans | 과금 서비스 plan 사용가능 | X |

- **사용예시**

```
$ cf create-quota cf-quota -m 500m -i 256m -r 2000 -s 500
```

delete-quota

- 기본 Syntax

```
$ cf delete-quota <QUOTA> [-f]
```

- 설명

Quota 정보를 삭제합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------|--------------------------|---------|
| QUOTA | QUOTA 명 | O |
| -f | 삭제 확인메시지 없이 QUOTA 정보를 삭제 | X |

- 사용예시

```
$ cf delete-quota cf-quota
```

update-quota

• 기본 Syntax

```
$ cf update-quota <QUOTA> [-m TOTAL_MEMORY] [-i INSTANCE_MEMORY][-n NEW_NAME] [-r ROUTES] [-s SERVICE_INSTANCES] [--allow-paid-service-plans | --disallow-paid-service-plans]
```

• 설명

Quota 정보를 수정합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------------|--|---------|
| QUOTA | QUOTA 명 | O |
| -m TOTAL_MEMORY | 메모리 할당량 Ex) 1024M, 1G, 10G | X |
| -i INSTANCE_MEMORY | App instance 가 가질수 있는 최대할당량 (-1 은 무한대) Ex) 1024M, 1G, 10G | X |
| -n NEW_NAME | QUOTA 명 변경시 변경할 이름 | X |
| -r ROUTES | 최대 라우트 수 | X |

| 파라미터명 | 설명 | 필수(O/X) |
|-------------------------------|-------------------|---------|
| -s SERVICE_INSTANCES | 최대 서비스 인스턴스 수 | X |
| --allow-paid-service-plans | 과금 서비스 plan 사용가능 | X |
| --disallow-paid-service-plans | 과금 서비스 plan 사용 불가 | X |

- **사용예시**

```
$ cf update-quota cf-quota -m 500m -i 256m -r 2000 -s 500
```

shared-private-domain

• 기본 Syntax

```
$ cf shared-private-domain <ORG> <DOMAIN>
```

• 설명

private 도메인을 다른 조직과 공유합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------|---------|---------|
| QUOTA | QUOTA 명 | O |
| DOMAIN | 도메인명 | O |

• 사용예시

```
$ cf shared-private-domain cf-org sharedomain.or.kr
```

unshared-private-domain

- 기본 Syntax

```
$ cf unshared-private-domain <ORG> <DOMAIN>
```

- 설명

다른 조직과 share 한 도메인 정보를 unshare 합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------|------|---------|
| ORG | 도메인명 | O |
| DOMAIN | 도메인명 | O |

- 사용예시

```
$ cf unshared-private-domain cf-org sharedomain.or.kr
```

SPACE ADMIN

space-quotas

- 기본 Syntax

```
$ cf space-quotas
```

- 설명

Space-quota 정보 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf space-quotas
```

space-quota

- 기본 Syntax

```
$ cf space-quota <SPACE_QUOTA_NAME>
```

- 설명

Space quota 상세정보를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|--------------|---------|
| SPACE_QUOTA_NAME | 스페이스 QUOTA 명 | O |

- 사용예시

```
$ cf space-quota cf-space-quota
```


create-space-quota

기본 Syntax

```
$ cf create-space-quota <QUOTA> [-i INSTANCE_MEMORY] [-m MEMORY] [-r ROUTES] [-s SERVICE_INSTANCES] [--allow-paid-service-plans]
```

설명

스페이스 Quota 정보를 생성합니다.

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------------------|--|---------|
| QUOTA | QUOTA 명 | O |
| -m TOTAL_MEMORY | 메모리 할당량 Ex) 1024M, 1G, 10G | X |
| -i INSTANCE_MEMORY | App instance 가 가질수 있는 최대할당량 (-1 은 무한대) Ex) 1024M, 1G, 10G | X |
| -r ROUTES | 최대 라우트 수 | X |
| -s SERVICE_INSTANCES | 최대 서비스 인스턴스 수 | X |
| --allow-paid-service-plans | 과금 서비스 plan 사용가능 | X |

- **사용예시**

```
$ cf create-space-quota cf-space-quota -i 2G -m 10G -r 3000 -s 200
```

update-space-quota

기본 Syntax

```
$ cf update-space-quota <SPACE-QUOTA-NAME> [-i MAX-INSTANCE-MEMORY] [-m MEMORY] [-n NEW_NAME] [-r ROUTES] [-s SERVICES] [--allow-paid-service-plans | --disallow-paid-service-plans]
```

설명

스페이스 Quota 정보를 수정합니다.

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------------|--|---------|
| SPACE-QUOTA-NAME | 스페이스 QUOTA 명 | O |
| -i MAX-INSTANCE-MEMORY | App instance 가 가질수 있는 최대할당량 (-1 은 무한대) Ex) 1024M, 1G, 10G | X |
| -m MEMORY | 스페이스가 가질수 있는 최대 메모리 | X |
| -n NEW_NAME | 변경하려는 SPACE-QUOTA 명 | X |
| -r ROUTES | 스페이스가 가지는 최대 route 갯수 | X |
| -s SERVICES | 스페이스가 가지는 최대 서비스 인스턴스 갯수 | X |

| 파라미터명 | 설명 | 필수(O/X) |
|-------------------------------|-------------------|---------|
| --allow-paid-service-plans | 과금 서비스 plan 사용가능 | X |
| --disallow-paid-service-plans | 과금 서비스 plan 사용 불가 | X |

- **사용예시**

```
$ cf update-space-quota cf-space-quota -i 2G -m 10G -r 3000 -s 200
```

delete-space-quota

기본 Syntax

```
$ cf delete-space-quota <SPACE-QUOTA-NAME>
```

설명

스페이스 Quota 정보를 삭제합니다.

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|--------------------------------|---------|
| SPACE-QUOTA-NAME | 스페이스 QUOTA 명 | O |
| -f | 삭제 확인메시지 없이 SPACE-QUOTA 정보를 삭제 | X |

사용예시

```
$ cf delete-space-quota cf-space-quota
```

set-space-quota

- 기본 Syntax

```
$ cf set-space-quota <SPACE-NAME> <SPACE-QUOTA-NAME>
```

- 설명

스페이스에 quota 를 할당합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|------------------|--------------|---------|
| SPACE-NAME | 스페이스명 | O |
| SPACE-QUOTA-NAME | 스페이스 Quota 명 | O |

- 사용예시

```
$ cf set-space-quota development cf-space-quota
```

unset-space-quota

- 기본 Syntax

```
$ cf unset-space-quota SPACE QUOTA
```

- 설명

스페이스에 할당된 quota 를 회수합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------|--------------|---------|
| SPACE | 스페이스명 | O |
| QUOTA | 스페이스 Quota 명 | O |

- 사용예시

```
$ cf unset-space-quota development cf-space-quota
```

SERVICE ADMIN

service-auth-tokens

- 기본 Syntax

```
$ cf service-auth-tokens
```

- 설명

서비스 인증 토큰 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf service-auth-token
```


create-service-auth-token

• 기본 Syntax

```
$ cf create-service-auth-token <LABEL> <PROVIDER> <TOKEN>
```

• 설명

스페이스에 할당된 quota 를 회수합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-----------|---------|
| LABEL | 서비스 토큰 라벨 | O |
| PROVIDER | 서비스 제공자 | O |
| TOKEN | 토큰명 | O |

• 사용예시

```
$ cf create-service-auth-token token-label mysql token
```

update-service-auth-token

• 기본 Syntax

```
$ cf update-service-auth-token <LABEL> <PROVIDER> <TOKEN>
```

• 설명

Service auth token 정보를 수정합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|-----------|---------|
| LABEL | 서비스 토큰 라벨 | O |
| PROVIDER | 서비스 제공자 | O |
| TOKEN | 토큰명 | O |

• 사용예시

```
$ cf update-service-auth-token token-label mysql token
```

delete-service-auth-token

• 기본 Syntax

```
$ cf delete-service-auth-token <LABEL> <PROVIDER> [-f]
```

• 설명

Service auth token 정보를 삭제합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------|----------------------------------|---------|
| LABEL | 서비스 토큰 라벨 | O |
| PROVIDER | 서비스 제공자 | O |
| -f | 삭제 확인메시지 없이 SERVICE TOKEN 정보를 삭제 | X |

• 사용예시

```
$ cf delete-service-auth-token token-label mysql
```

service-brokers

- 기본 Syntax

```
$ cf delete-service-auth-token <LABEL> <PROVIDER> [-f]
```

- 설명

Service Broker 정보 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf service-brokers
```

create-service-broker

• 기본 Syntax

```
$ cf create-service-broker <SERVICE_BROKER> <USERNAME> <PASSWORD> <URL>
```

• 설명

Service Broker 정보를 등록합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------------------|-------------|---------|
| SERVICE_BROKERLABEL | 서비스 브로커명 | O |
| USERNAME | 사용자명 | O |
| PASSWORD | 패스워드 | O |
| URL | 서비스 브로커 URL | O |

• 사용예시

```
$ cf create-service-broker mysql-service-broker admin password http://p-mysql.10.244.0.34.xip.io
```

update-service-broker

• 기본 Syntax

```
$ cf update-service-broker <SERVICE_BROKER> <USERNAME> <PASSWORD> <URL>
```

• 설명

Service Broker 정보를 등록합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------------------|-------------|---------|
| SERVICE_BROKERLABEL | 서비스 브로커명 | O |
| USERNAME | 사용자명 | O |
| PASSWORD | 패스워드 | O |
| URL | 서비스 브로커 URL | O |

• 사용예시

```
$ cf update-service-broker mysql-service-broker admin password http://p-mysql.10.244.0.34.xip.io
```

delete-service-broker

기본 Syntax

```
$ cf delete-service-broker <SERVICE_BROKER> [-f]
```

설명

Service Broker 정보를 삭제합니다.

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|-----------------------------------|---------|
| SERVICE_BROKER | 서비스 브로커명 | O |
| -f | 삭제 확인메시지 없이 SERVICE BROKER 정보를 삭제 | X |

사용예시

```
$ cf delete-service-broker mysql-service-broker
```

rename-service-broker

- 기본 Syntax

```
$ cf rename-service-broker <SERVICE_BROKER> <NEW_SERVICE_BROKER>
```

- 설명

Service Broker 명을 수정합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------------|--------------|---------|
| SERVICE_BROKER | 서비스 브로커명 | O |
| NEW_SERVICE_BROKER | 변경할 서비스 브로커명 | O |

- 사용예시

```
$ cf rename-service-broker mysql-service-broker new_mysql-service-broker
```


migrate-service-broker

• 기본 Syntax

```
$ cf migrate-service-instances <v1_SERVICE> <v1_PROVIDER> <v1_PLAN> <v2_SERVICE> <v2_PLAN>
```

• 설명

서비스 인스턴스에서 사용하는 서비스 및 플랜을 다른 플랜으로 변경합니다.
 - App 이 사용하는 서비스를 다른 서비스로 변경하려 할때 사용합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|-----------------------|---------|
| v1_SERVICE | 기존 서비스 명 | O |
| v1_PROVIDER | 기존 서비스를 제공하는 제공자 | O |
| v1_PLAN | 기존 서비스 인스턴스에서 사용하는 플랜 | O |
| v2_SERVICE | 신규 서비스 명 | O |
| v2_PLAN | 신규 서비스에서 사용하는 플랜 | O |

• 사용예시

```
$ cf migrate-service-instances p-mysql mysql-provider silver postgres silver
```

purge-service-offering

기본 Syntax

```
$ cf purge-service-offering <SERVICE> [-p PROVIDER]
```

설명

cf 와 서비스 브로커간의 정보 불일치를 해결할때 사용합니다.
 (migrate-service-instances 명령 이후 사용)

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------|--------------------------|---------|
| SERVICE | 서비스 명 | O |
| -p PROVIDER | 서비스 제공자 | O |
| -f | 삭제 확인메시지 없이 서비스 정보를 삭제한다 | O |

사용예시

```
$ cf purge-service-offering mysql
```

service-access

- 기본 Syntax

```
$ cf service-access
```

- 설명

```
서비스 access 될 서비스 목록 조회합니다..
```

- 파라미터

- 없음

- 사용예시

```
$ cf service-access
```

enable-service-access

• 기본 Syntax

```
$ cf enable-service-access <SERVICE> [-p PLAN] [-o ORG]
```

• 설명

조직 또는 서비스 plan 을 서비스에 접근 가능하도록 설정합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------|--------|---------|
| SERVICE | 서비스 명 | O |
| -p PLAN | PLAN 명 | O |
| -o ORG | 조직명 | O |

• 사용예시

```
$ cf enable-service-access mysql -p silver -o cf-org
```

disable-service-access

• 기본 Syntax

```
$ cf disable-service-access <SERVICE> [-p PLAN] [-o ORG]
```

• 설명

조직 또는 서비스 plan 을 서비스에 접근 불가 하도록 설정합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|---------|--------|---------|
| SERVICE | 서비스 명 | O |
| -p PLAN | PLAN 명 | O |
| -o ORG | 조직명 | O |

• 사용예시

```
$ cf disable-service-access mysql -p silver -o cf-org
```

SECURITY GROUP

security-group

- 기본 Syntax

```
$ cf security-group <SECURITY_GROUP>
```

- 설명

서큐리티 그룹 상세정보를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|----------|---------|
| SECURITY_GROUP | 서큐리티 그룹명 | O |

- 사용예시

```
$ cf security-group cf-security-group
```

security-groups

- 기본 Syntax

```
$ cf security-groups
```

- 설명

시큐리티 그룹 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf security-groups
```

create-security-group

- 기본 Syntax

```
$ cf create-security-group <SECURITY_GROUP> <PATH_TO_JSON_RULES_FILE>
```

- 설명

시큐리티 그룹정보를 생성합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------------------|--|---------|
| SECURITY_GROUP | 서큐리티 그룹명 | O |
| PATH_TO_JSON_RULES_FILE | 시큐리티 룰을 명시한 JSON 파일의 경로 및 파일명 ex) rule 파일 작성 예제 [{ "protocol": "tcp", "destination": "10.244.1.18", "ports": "3306" }] | O |

- 사용예시

```
$ cf create-security-group cf-security-group ./rule.json
```


update-security-group

- 기본 Syntax

```
$ cf update-security-group <SECURITY_GROUP> <PATH_TO_JSON_RULES_FILE>
```

- 설명

시큐리티 그룹정보를 수정합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-------------------------|--|---------|
| SECURITY_GROUP | 서큐리티 그룹명 | O |
| PATH_TO_JSON_RULES_FILE | 시큐리티 룰을 명시한 JSON 파일의 경로 및 파일명 ex) rule 파일 작성 예제 [{ "protocol": "tcp", "destination": "10.244.1.18", "ports": "3306" }] | O |

- 사용예시

```
$ cf update-security-group cf-security-group ./rule.json
```

delete-security-group

• 기본 Syntax

```
$ cf delete-security-group <SECURITY_GROUP> [-f]
```

• 설명

시큐리티 그룹정보를 삭제합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|--------------------------------|---------|
| SECURITY_GROUP | 서큐리티 그룹명 | O |
| -f | 삭제 확인메시지 없이 시큐리지 그룹 정보를 삭제합니다. | X |

• 사용예시

```
$ cf update-security-group cf-security-group ./rule.json
```

bind-security-group

• 기본 Syntax

```
$ cf bind-security-group <SECURITY_GROUP> <ORG> <SPACE>
```

• 설명

시큐리티 그룹 정보와 스페이스를 바인드 합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|----------|---------|
| SECURITY_GROUP | 서큐리티 그룹명 | O |
| ORG | 조직명 | O |
| SPACE | 스페이스명 | O |

• 사용예시

```
$ cf update-security-group cf-security-group ./rule.json
```

unbind-security-group

• 기본 Syntax

```
$ cf unbind-security-group <SECURITY_GROUP> <ORG> <SPACE>
```

• 설명

시큐리티 그룹 정보와 스페이스를 언바인드 합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|----------|---------|
| SECURITY_GROUP | 서큐리티 그룹명 | O |
| ORG | 조직명 | O |
| SPACE | 스페이스명 | O |

• 사용예시

```
$ cf unbind-security-group cf-security-group cf-group development
```

bind-staging-security-group

• 기본 Syntax

```
$ cf bind-staging-security-group <SECURITY_GROUP>
```

• 설명

App staging 처리를 하기 위해 시큐리티 그룹을 설정합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|----------|---------|
| SECURITY_GROUP | 서큐리티 그룹명 | O |

• 사용예시

```
$ cf bind-staging-security-group cf-security-group
```

staging-security-groups

- 기본 Syntax

```
$ cf staging-security-groups
```

- 설명

Staging security group 정보 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf staging-security-groups
```

unbind-staging-security-group

- 기본 Syntax

```
$ cf unbind-staging-security-group <SECURITY_GROUP>
```

- 설명

App staging 처리를 하기 위한 시큐리티 그룹을 설정을 해제 합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|----------------|----------|---------|
| SECURITY_GROUP | 서큐리티 그룹명 | O |

- 사용예시

```
$ cf unbind-staging-security-group cf-security-group
```

running-security-groups

- 기본 Syntax

```
$ cf unbind-staging-security-group <SECURITY_GROUP>
```

- 설명

실행중인 시큐리티 그룹 목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf unbind-staging-security-group cf-security-group
```


ENVIRONMENT VARIABLE GROUPS

running-environment-variable-group, revg

- 기본 Syntax

```
$ cf running-environment-variable-group
```

- 설명

실환경 변수 내용을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf running-environment-variable-group
```

staging-environment-variable-group, sevg

- 기본 Syntax

```
$ cf staging-environment-variable-group
```

- 설명

스테이징시 사용되는 환경변수 내용을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$ cf staging-environment-variable-group
```

set-staging-environment-variable-group, ssevg

- 기본 Syntax

```
$ cf set-staging-environment-variable-group <ENV_VARIABLE>
```

- 설명

스테이징시 사용되는 환경변수 내용을 설정한다

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|--------------------------|---------|
| ENV_VARIABLE | 환경변수 내용으로 KEY/VALUE 로 구성 | O |

- 사용예시

```
$ cf set-staging-environment-variable-group '{"name":"value","name":"value"}'
```

set-running-environment-variable-group, ssevg

• 기본 Syntax

```
$ cf set-running-environment-variable-group <ENV_VARIABLE>
```

• 설명

환경변수 내용을 설정 합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|-----------------------------|---------|
| ENV_VARIABLE | 환경변수 내용으로 KEY/VALUE 로 구성된다. | O |

• 사용예시

```
$ cf set-running-environment-variable-group '{"name":"value","name":"value"}'
```

FEATURE FLAGS

feature-flags

- 기본 Syntax

```
$ cf feature-flags
```

- 설명

```
feature flags 목록을 조회합니다.
```

- 파라미터

- 없음

- 사용예시

```
$ cf feature-flags
```

feature-flag

- 기본 Syntax

```
$ cf feature-flag <FEATURE_NAME>
```

- 설명

특정 Feature flag 의 상태를 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|--|---------|
| FEATURE_NAME | Feature flag 명. - feature flag 에는 6 가지가 있다. 1)user_org_creation 2) private_domain_creation 3) app_bits_upload 4) app_scaling 5) route_creation 6) service_instance_creation | O |

- 사용예시

```
$ cf feature-flag app_bits_upload
```

enable-feature-flag

• 기본 Syntax

```
$ cf enable-feature-flag <FEATURE_NAME>
```

• 설명

특정 Feature flag 의 상태를 enable 로 변경합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|--|---------|
| FEATURE_NAME | Feature flag 명. - feature flag 에는 6 가지가 있다. 1)user_org_creation 2) private_domain_creation 3) app_bits_upload 4) app_scaling 5) route_creation 6) service_instance_creation | O |

• 사용예시

```
$ cf enable-feature-flag app_bits_upload
```

disable-feature-flag

기본 Syntax

```
$ cf disable-feature-flag <FEATURE_NAME>
```

설명

특정 Feature flag 의 상태를 disable 로 변경합니다.

파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--------------|--|---------|
| FEATURE_NAME | Feature flag 명. - feature flag 에는 6 가지가 있다. 1)user_org_creation 2) private_domain_creation 3) app_bits_upload 4) app_scaling 5) route_creation 6) service_instance_creation | O |

사용예시

```
$ cf disable-feature-flag app_bits_upload
```


ADVANCE

curl

- 기본 Syntax

```
$ cf curl <PATH> [-i] [-v] [-X METHOD] [-H HEADER] [-d DATA] [--output FILE]
```

- 설명

PCF CLI 명령어가 아닌 PCF API 를 호출합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------|--|---------|
| PATH | Cf api path Ex) /v2/spaces/2d94e7ee-9805-408d-a1eb-ceac319e603b/summary | O |
| -i | Response header 포함한 결과 | X |
| -v | Request/response 에 CF_TRACE enable 된 내용 포함 | X |
| -X METHOD | HTTP method((GET,POST,PUT,DELETE,etc) | X |
| -H HEADER | Request 에 Custom Header 를 포함합니다. | X |
| -d DATA | Request 에 Http data 를 포함합니다. | X |

| 파라미터명 | 설명 | 필수(O/X) |
|---------------|-------------------------------------|---------|
| --output FILE | Response 결과를 stdout 대신 FILE 로 결과 저장 | X |

- **사용예시**

```
$ cf curl /v2/spaces/2d94e7ee-9805-408d-a1eb-ceac319e603b/summar
```

config

• 기본 Syntax

```
$ cf config [--async-timeout TIMEOUT_IN_MINUTES] [--trace true | false | path/to/file] [--color true | false] [--locale (LOCALE | CLEAR)]
```

• 설명

CF CLI 에 대한 설정.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|--|---------------------------------------|---------|
| --async-timeout TIMEOUT_IN_MINUTES | CLI 명령 전송시 async timeout 설정 | X |
| --trace (true / false / path/to/file) | CLI 명령 수행시 실행되는 cf api 의 내용 출력 설정 | X |
| --color true / false | CLI 명령 수행시 실행되는 cf api 의 내용 color 설정 | X |
| --locale (LOCALE / CLEAR) | CLI 명령 수행시 실행되는 cf api 의 내용 locale 설정 | X |

• 사용예시

```
$ cf curl /v2/spaces/2d94e7ee-9805-408d-a1eb-ceac319e603b/summar
```

oauth-token

- 기본 Syntax

```
$ cf oauth-token
```

- 설명

사용자가 cf login 후 CF 에서 받은 token 값 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$cf oauth-token
```

ADD/REMOVE PLUGIN REPOSITORY

add-plugin-repo

- 기본 Syntax

```
$ cf add-plugin-repo <REPO_NAME> <URL>
```

- 설명

PCF CLI plugin repository(저장소)를 추가합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------|----------------|---------|
| REPO_NAME | Repository 명 | X |
| URL | Repository URL | X |

- 사용예시

```
cf add-plugin-repo Diego-SSH http://plugins.cloudfoundry.org
```

remove-plugin-repo

- 기본 Syntax

```
$ cf remove-plugin-repo <REPO_NAME> <URL>
```

- **설명**

CLI plugin repository(저장소)를 삭제합니다.

- **파라미터**

| 파라미터명 | 설명 | 필수(O/X) |
|-----------|----------------|---------|
| REPO_NAME | Repository 명 | O |
| URL | Repository URL | O |

- **사용예시**

```
cf remove-plugin-repo Diego-SSH http://plugins.cloudfoundry.org
```

list-plugin-repos

- 기본 Syntax

```
$ cf list-plugin-repos
```

- 설명

CLI 에 추가된 plugin repository(저장소)목록을 조회합니다.

- 파라미터

- 없음

- 사용예시

```
$cf list-plugin-repos
```

repo-plugins

- 기본 Syntax

```
$ cf repo-plugins [-r REPO_NAME]
```

- 설명

Repository 에 있는 플러그인 목록을 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------|--------------|---------|
| REPO_NAME | Repository 명 | X |

- 사용예시

```
$ cf repo-plugins
```


ADD/REMOVE PLUGIN

plugins

- 기본 Syntax

```
$ cf plugins
```

- 설명

추가된 plugin 의 사용가능한 명령어 목록을 조회합니다.

- 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------|--------------|---------|
| REPO_NAME | Repository 명 | X |

- 사용예시

```
$ cf repo-plugins
```

install-plugin

• 기본 Syntax

```
$ cf install-plugin < URL or LOCAL-PATH/TO/PLUGIN> [-r REPO_NAME]
```

• 설명

추가된 plugin 의 사용가능한 명령어 목록을 조회합니다.

• 파라미터

| 파라미터명 | 설명 | 필수(O/X) |
|-----------------------------|---|---------|
| URL or LOCAL-PATH/TO/PLUGIN | Plugin URL 또는 로컬경로 또는 repository 에 있는 플러그인명 | X |
| -r REPO_NAME | Plugin repository 명 | X |

• 사용예시

```
$cf install-plugin 'Usage Report' -r CF-Community
```