

-docker



v2.5gf dockerMakefilegf build, gf gen enums, gf dockermake build, make enums, make docker

Content Menu

•
•
•

```
$ gf docker -h
USAGE
    gf docker [MAIN] [OPTION]

ARGUMENT
    MAIN    main file path for "gf build", it's "main.go" in default.
            empty string for no binary build

OPTION
    -f, --file           file path of the Dockerfile. it's "manifest/docker
/Dockerfile" in default
    -s, --shell          path of the shell file which is executed before
docker build
    -b, --build          binary build options before docker image build,
it's "-a amd64 -s linux" in default
    -tn, --tagName       tag name for this docker, pattern like "image:
tag". this option is required with TagPrefixes
    -tp, --tagPrefixes  tag prefixes for this docker, which are used for
docker push. this option is required with
                        TagName
    -p, --push           auto push the docker image to docker registry if
"-t" option passed
    -e, --extra          extra build options passed to "docker image"
    -h, --help           more information about this command

EXAMPLE
    gf docker
    gf docker -t hub.docker.com/john/image:tag
    gf docker -p -t hub.docker.com/john/image:tag
    gf docker main.go
    gf docker main.go -t hub.docker.com/john/image:tag
    gf docker main.go -t hub.docker.com/john/image:tag
    gf docker main.go -p -t hub.docker.com/john/image:tag

DESCRIPTION
    The "docker" command builds the GF project to a docker images.
    It runs "gf build" firstly to compile the project to binary file.
    It then runs "docker build" command automatically to generate the
docker image.
    You should have docker installed, and there must be a Dockerfile in
the root of the project.
```

dockerMAINmain.goOPTIONSDocker build

```

$ gf docker main.go -p -tn loads/gf-demos:test
2020-12-31 00:47:28.207 start building...
2020-12-31 00:47:28.207 go build -o ./bin/linux_amd64/main main.go
2020-12-31 00:47:35.894 done!
Sending build context to Docker daemon 37.63MB
Step 1/10 : FROM loads/alpine:3.8
--> f9fb622e6db2
Step 2/10 : LABEL maintainer="john@goframe.org"
--> Using cache
--> da238418d031
Step 3/10 : ENV WORKDIR /var/www/gf-demos
--> Using cache
--> 3e7129c087c9
Step 4/10 : ADD ./bin/linux_amd64/main $WORKDIR/main
--> 3661a9dea494
Step 5/10 : RUN chmod +x $WORKDIR/main
--> Running in 1d49d5d91080
Removing intermediate container 1d49d5d91080
--> a03ee04e3380
Step 6/10 : ADD public $WORKDIR/public
--> 63dd06d0e1a3
Step 7/10 : ADD config $WORKDIR/config
--> fa7a57eba577
Step 8/10 : ADD template $WORKDIR/template
--> 7075609b0447
Step 9/10 : WORKDIR $WORKDIR
--> Running in a34ef38e1031
Removing intermediate container a34ef38e1031
--> 580077998eaf
Step 10/10 : CMD ./main
--> Running in ed286b518ad9
Removing intermediate container ed286b518ad9
--> fbbc05842901
Successfully built fbbc05842901
Successfully tagged loads/gf-demos:test
The push refers to repository [docker.io/loads/gf-demos]
b4025b95a79f: Preparing
9e0369a57507: Preparing
46c68dcc8e12: Preparing
59adbc083ee5: Preparing
10e0b999ba57: Preparing
8e850d7b086e: Waiting
d5e057db20a2: Waiting
92e898fd7f84: Waiting
d9ff549177a9: Waiting
...

```

hack/config.yamldocker

```

gfcli:
  docker:
    build: "-a amd64 -s linux -p temp -ew"
    tagPrefixes:
      - ccr.ccs.tencentyun.com/cdb.khaos.eros

```