

# Cypress Docker Tutorial

*A Step by Step Guide*



# Running headless Cypress tests locally using Docker Image

```
docker run -it -v $PWD:/e2e -w /e2e cypress/included:12.8.1
```

# Running Cypress tests on Chrome and Firefox browsers using Docker Image

```
docker run -it -v $PWD:/e2e -w /e2e cypress/included:12.8.1 --browser firefox
```

```
docker run -it -v $PWD:/e2e -w /e2e cypress/included:12.8.1 --browser chrome
```

# Running Cypress tests on Chrome and Firefox browsers using Dockerfile

```
ARG CHROME_VERSION='107.0.5304.121-1'  
ARG EDGE_VERSION='100.0.1185.29-1'  
ARG FIREFOX_VERSION='107.0'
```

```
FROM cypress/factory
```

```
WORKDIR /app  
COPY . .
```

```
RUN npm install --save-dev cypress
```

# Running Cypress tests in headless mode using Dockerfile



```
FROM cypress/included:12.8.1
```

```
WORKDIR /app
```

```
COPY . .
```

```
RUN npm install
```

```
CMD ["npm", "run", "npx cypress run"]
```

# Running Cypress tests on multiple browsers using Docker Compose

```
version: 'cypress-docker-v1'
services:

  cypress-chrome:
    image: cypress/included:12.8.1
    command: "--browser chrome"
    working_dir: /cypress
    volumes:
      - ./:/cypress

  cypress-firefox:
    image: cypress/included:12.8.1
    command: "--browser firefox"
    working_dir: /cypress
    volumes:
      - ./:/cypress

  cypress-electron:
    image: cypress/included:12.8.1
    working_dir: /cypress
    volumes:
      - ./:/cypress
```

# Running Cypress tests on Cloud Grid using Dockerfile

```
FROM cypress/included:12.8.1
WORKDIR /app
COPY . .
RUN npm install
ENTRYPOINT ["npm", "run", "cypress:test:cloud"]
```

# Running Cypress tests in Docker Container on CI/CD (GitHub Actions)



```
name: Cypress Test in custom container
on: [push]
jobs:
  cypress-test:
    runs-on: ubuntu-20.04
    name: Cypress Tests on docker container
    container: cypress/browsers:node12.13.0-chrome78-ff7
    steps:
      - uses: actions/checkout@v3
      - uses: cypress-io/github-action@v5
    with:
      browser: chrome
```



# Bonus: Top Docker Commands

Command	Explanation
<code>docker build -t &lt;image-name&gt;</code>	Build Docker Image
<code>docker pull &lt;image-name&gt;</code>	Pull Docker Image
<code>docker push &lt;image-name&gt;:tag-name</code>	Push Docker Image
<code>docker run -t &lt;image-name&gt;</code>	Run Docker Image
<code>docker images</code>	Show all docker images
<code>docker system prune -a</code>	Stop all containers and delete all the images
<code>docker kill &lt;container-id&gt;</code>	Kill docker container
<code>docker run &lt;imag_name&gt;</code>	Start Docker
<code>docker login</code>	Login In
<code>docker ps -a</code>	List all running docker containers
<code>docker stop</code>	Stop container
<code>docker logs containerId</code>	View the logs
<code>docker-compose up</code>	Start a group of containers defined in a Docker Compose fil

# For more detailed understanding of Cypress Docker Tutorial

Blog : <https://www.lambdatest.com/learning-hub/cypress-docker>

