Docker

- Install docker

Install docker

This is a simple guide on how to install docker on a raspberry pi

[Make sure you have sudo access to the PI!!]

To install docker itself i recommend using the

documentation from docker the documentation has a convenience script seen bellow

```
curl -fsSL https://get.docker.com -o get-docker.sh
sudo sh ./get-docker.sh
```

Then you need to allow your user to use docker commands without sudo

```
sudo usermod -aG docker $USER
```

Then you can either reboot or reload you server with the command bellow

```
newgrp docker
```

Example containers and configuration

In my favorite way to do docker compose files is a structure like this

/docker/[Projectname]/compose.yml

so we create the docker directory, allow the user to read and write in the directory and also creating a symbolic link to the directory in the home directory

```
sudo mkdir /docker
sudo chown $USER:$USER /docker
sudo ln -s /docker
```

Breakdown of the commands

- `sudo mkdir /docker`: Creates a new directory called `docker` in the root directory. (Root directory is `/`)
- `sudo chown $USER:$USER /docker`: Changes the ownership of the `/docker` directory to the current user, allowing read and write access.

- `sudo ln -s /docker`: Creates a symbolic link to the `/docker` directory in the current user's home directory, making it easier to access.

Other important commands

- `docker compose up -d`: Starts the Docker containers defined in the `compose.yml` file in detached mode, allowing them to run in the background. The `-d` flag stands for "detached mode".
- `docker compose down`: Stops and removes the containers defined in the `compose.yml` file, cleaning up resources.

Creating a pihole installation

> This is just a exerpt from the pihole documentation

folowing the structure we make a new directory in /docker

```
mkdir /docker/pihole
cd /docker/pihole
nano compose.yml
```

then copy the pihole example compose file from below

```
# More info at https://github.com/pi-hole/docker-pi-hole/ and https://docs.pi-hole.net/
services:
  pihole:
    container_name: pihole
    image: pihole/pihole:latest
    ports:
      # DNS ports
      - "53:53/tcp"
      - "53:53/udp"
      # WEB GUI
      - "80:80/tcp"
      - "443:443/tcp"
      # DHCP
      - "67:67/udp"
    environment:
      TZ: 'Europe/Oslo'
      FTLCONF_webserver_api_password: '(change me)'
      FTLCONF_dns_listeningMode: 'all'
    volumes:
      - './etc-pihole:/etc/pihole'
```

```
    cap_add:
      - NET_ADMIN
      - SYS_TIME
      - SYS_NICE
    restart: unless-stopped
```

Save and exit and now we are ready to start pihole with this will create a configuration directory

```
docker compose up -d
```

Creating a Minecraft Server

> This is just a exerpt from the [itzg docker minecraft server documentation]

folowing the structure we make a new directory in /docker

```
mkdir /docker/minecraft
cd /docker/minecraft
nano compose.yml
```

then copy the pihole example compose file from below

```
services:
  mc:
    image: itzg/minecraft-server
    tty: true
    stdin_open: true
    ports:
      - "25565:25565"
    environment:
      EULA: "TRUE"
    volumes:
      # attach the relative directory 'data' to the container's /data path
      - ./data:/data
```

Save and exit and now we are ready to start pihole with this will create a configuration directory

```
docker compose up -d
```

And then you can access your minecraft server with the ip specified by the command

```
ip a
```

Specificaly this ip address