systemd

In order to have a command or program run when the Pi boots, you can add it as a service. Once this is done, you can start/stop enable/disable from the linux prompt.

Creating a service

On your Pi, create a .service file for your service, for example:

myscript.service

```
[Unit]
Description=My service
After=network.target

[Service]
ExecStart=/usr/bin/python3 -u main.py
WorkingDirectory=/home/pi/myscript
StandardOutput=inherit
StandardError=inherit
Restart=always
User=pi

[Install]
WantedBy=multi-user.target
```

So in this instance, the service would run Python 3 from our working directory `/home/pi/myscript` which contains our python program to run `main.py`. But you are not limited to Python programs: simply change the ExecStart line to be the command to start any program/script that you want running from booting.

Copy this file into `/etc/systemd/system` as root, for example:

```
sudo cp myscript.service /etc/systemd/system/myscript.service
```

Once this has been copied, you can attempt to start the service using the following command:

```
sudo systemctl start myscript.service
```

Stop it using following command:

```
sudo systemctl stop myscript.service
```

When you are happy that this starts and stops your app, you can have it start automatically on reboot by using this command:

```
sudo systemctl enable myscript.service
```

The `systemctl` command can also be used to restart the service or disable it from boot up!

Some things to be aware of:

- The order in which things are started is based on their dependencies — this particular script should start fairly late in the boot process, after a network is available (see the After section).
- You can configure different dependencies and orders based on your requirements.

You can get more information from: `man systemctl` or here: https://fedoramagazine.org/what-is-an-init-system/