This is a collection of tutorials for the FIWARE system. Each tutorial consists of a series of exercises to demonstrate the correct use of individual FIWARE components and shows the flow of context data within a simple Smart Solution either by connecting to a series of dummy IoT devices or manipulating the context directly or programmatically.

# FIWARE Step-by-Step

## How to Use

Each tutorial is a self contained learning exercise designed to teach the developer about a single aspect of FIWARE. A summary of the goal of the tutorial can be found in the description at the head of each page. Every tutorial is associated with a GitHub repository holding the configuration files needed to run the examples. Most of the tutorials build upon concepts or enablers described in previous exercises the to create a complex smart solution which is *“powered by FIWARE”*.

The tutorials are split according to the chapters defined within the [FIWARE catalog](https://www.fiware.org/developers/catalogue/) and are numbered in order of difficulty within each chapter hence the an introduction to a given enabler will occur before the full capabilities of that element are explored in more depth.

It is recommended to start with reading the full **Core Context Management: Fundamentals** Chapter before moving on to other subjects, as this will give you an fuller understanding of the role of context data in general. However it is not necessary to follow all the subsequent tutorials sequentially - as FIWARE is a modular system, you can choose which enablers are of interest to you.

## Prerequisites

### Docker and Docker Compose

To keep things simple all components will be run using [Docker](https://www.docker.com). **Docker** is a container technology which allows to different components isolated into their respective environments.

* To install Docker on Windows follow the instructions [here](https://docs.docker.com/docker-for-windows/)
* To install Docker on Mac follow the instructions [here](https://docs.docker.com/docker-for-mac/)
* To install Docker on Linux follow the instructions [here](https://docs.docker.com/install/)

**Docker Compose** is a tool for defining and running multi-container Docker applications. A series of \*.yaml files are used configure the required services for the application. This means all container services can be brought up in a single command. Docker Compose is installed by default as part of Docker for Windows and Docker for Mac, however Linux users will need to follow the instructions found [here](https://docs.docker.com/compose/install/)

You can check your current **Docker** and **Docker Compose** versions using the following commands:

docker-compose -v  
docker version

Please ensure that you are using Docker version 18.03 or higher and Docker Compose 1.21 or higher and upgrade if necessary.

### Cygwin for Windows

We will start up our services using a simple Bash script. Windows users should download [cygwin](http://www.cygwin.com/) to provide a command-line functionality similar to a Linux distribution on Windows.