Inventor2040W *SMART LAMP*

Hello everybody!

This is my **Fundamentals of Programming 1** project.
It is based on the *Inventor2040W* board and the *bmx680* breakout both from Pimoroni.
The code is running in Micropython.
I really enjoyed coding this project, I hope you'll have fun reading it!
I've put a lot of effort into writing the Markdown file, it would be a pity not viewing it from github

Follow the next steps to get the project running:

1. Download the project folder

To clone the git repository:

- Open a new terminal window
- Move to the path you want `cd /thepathyouwant`
- Clone the git repository `git clone https://github.com/01ski/inventor_py_project.git`

Or you can just unzip the folder somewhere in your filesystem .

2. Hardware and environment

Make sure you have all the needed hardware and install the environment **(tutorials are linked)**

1. Hardware:
    - Inventor2040W
    - BME680 connected to the board via the i2c soldered port
    - MicroUSB cable to connect the Inventor2040W to the PC
2. Software:
    - Pimoroni Micropython version
    - Thonny or mpremote

Once you have set up the environment you can decide whether to use Thonny or mpremote. I will just explain how to run the project in mpremote, because I liked it better while I was using it - it's fast and it runs directly from the command line.

3. Connect to the device, copy files on it

If you haven't done it yet, change directory to the place where you stored the project folder.

```
cd /projectpath
```

Plug in your device, then to connect your device to mpremote, typing this into the command line.

```
mpremote
```

You should get an output like this:

```
Connected to COM5
```

Exit this message with Ctrl-X.

Copy the modules on to your device with the following lines

```
mpremote cp global_module.py :global_module.py
mpremote cp temperature_module.py :temperature_module.py
mpremote cp humidity_module.py :humidity_module.py
mpremote cp main.py :main.py
```

4. Run the program

There are two ways to run the program.
Since you have copied the main.py file to the device, the program can run by itself without the need for a connection to mpremote.

1. Run the program without mpremote:

```
To run on its own the device just needs to be connected to the power.
Because of the interference with mpremote, you first have to disconnect
the device from mpremote.
```

Enter `mpremote disconnect` , press Enter, then press Ctrl-X.
Hit the **Reset** button on the Inventor board --- TADAAA!

2. Run the program with mpremote:

```
If you run the program via mpremote you'll see some interesting output containing
```

Enter `mpremote run :main.py` or `mpremote run main.py` (the second will run the *main.py* from the PC filesystem, but since you copied the same file to the device it will make no difference)

Now you should see some output containing **air temperature, humidity and air pollution**

5. The program: explained

The program has two modes:

1. First mode (default)

It displays on the LEDs the room temperature:

Colour	Temperature
Blue	< 23 °C
Green	23 - 25 °C
Orange	25 - 27 °C
Red	27 - 30 °C
Purple	> 30 °C

The blinking of the LEDs shows how polluted the air is. There are four speeds: the faster the blinking, the more polluted is the air.

Try altering the sensor readings by breathing on it or spraying parfume next to it

2. The USER - Switch

- Press ONCE: Turns on/off more LEDs
- Press TWICE in a row (fast): Switch to *humidity-mode*
- Press THREE TIMES in a row (fast): Turn off

3. Second mode (humidity-mode)

It displays how much humidity the sensor is measuring, every turned on LED corresponds to 10% relative humidity.

e.g. if 4 LEDs are blinking, the humidity is between 40-50%