* MicroPython follows Python 3
* MicroPython is only a programming language interpreter and does not include an editor. This interpreter resides on the embedded device and can interpret python code
* You have to write your code in any editor that you like and then use small tools to upload and run the code on board.

Some board supports web based editor

* Runs on pyboard, bbc micro , esp8266, wiPy, teensy openWrt uses micropython
* There Is a space program (with ESA) which is trying to embed micropython in their leon mcu which has spark processor and a real time OS . idea is to use micropython in their OS so that py scripts can be sent to the space from earth and see the output

This is because python is so much easy for data manipulation and tests and is much better that matlab according to some geeks

* Something is preinstalled in the flash so when you connect the board to the pc it acts as a python shell so you can write code in it and it will run on the embedded hardware >> this prompt Is on the board
* microPython is not a full python
* some modules that Damien George - Creator of MicroPython imported in the shell running on esp8266 are:

1. micropython
2. network
3. socket
4. json
5. machine (gives control access to GPIO)
6. time

* so I guess there is a python interpreter + a shell like serial interface in compiled binary resides in the flash so when the mcu gets on/reset first a shell like interface is provided by the mcu to the serial and it waits for the incoming characters, then if any statement comes then the cpu takes the code interprets it through the interpreter and output is shown at the serial.

My second guess is that there are some txt files (which we call py modules) which resides in the flash with the compiled interpreter and the shell so when any module is imported what happens is that now cpu can go to that file if interpreter wants

* ctrl+C also works :D [keyboard interrupt]
* try catch works
* the embedded shell supports copy paste options also
* micropython has list ,dict, hash tables
* the creator also showed a cool demo where the esp8266 acts as a server and you can send on/off command in http protocol and toggle the LED
* microPython looking forward to support async IO and already supporting multithreading
* micropython needs atleast 256KB of flash but one thing that adafruit said that you can insert sd card and save your scripts there . this feature is present in the pyBoard
* There is a coffee machine product made by microPython

############################### Adafruit explanation ####################################

* The shell is interactive REPL [read evaluate print Loop] that is why it works
* There are built in libraries for micro Python which I think resides in the flash
* You can mix high level micropython code with low c/c++ code
* Micropython interpreter have to interpret every instruction out of a file and convert it in CPU code