SoC-Day10.2

July 25, 2018

1 SoC Day 10.2

Some of needed libraryes: install them using pip3 if needed. This RPi-Python document is intended for Python3 **only**

document version 0.1a

```
In [ ]: import time # to handle some time tasks
        # import RPi.GPIO as GPIO
        import gpiozero


        import RPi.GPIO as GPIO # well that's too old for me...
```

the PIN number you should be looking at is one OUTSIDE the circle on given scheme.

DO NOT connect **ANY** external device to 5V, 3.3V, Ground pins directly before wiring check. Ask the assistant to help you if you are unsure in pin numbering.

TODO: add scheme picture

1.1 LED:

```
In [ ]: my_first_led = gpiozero.LED(21) # pin configuration argument
```

```
In [ ]: my_first_led.on()
```

```
In [ ]: my_first_led.off()
```

```
In [ ]: dir(my_first_led)
```

```
In [ ]: # find blink function, get it's help document

        # relevant methods:

        # on, off, blink, is_lit, toggle
```

1.2 exercise no. 1:

Blik Using 2 LED's. Make it like this:

While LED no.1 is on, LED no.2 should be off. While LED no.2 is on, LED no.1 should be off. Make it periodically. Use sleep or blink functions.

use these resources:

a) two GPIO ports

b) a single GPIO port - be careful about what you're doing

1.3 exercise no. 2: Semaphore

1.4 PWMLED

```
In [1]: cool_led = PWMLED(21)
        cool_led.value = 0.0
        cool_led.value = 0.5
        cool_led.value = 1.0

        # arguments:

        # value
        # pulse(fade_in_time, fade_out_time, n=None, baskground)
        # blink(on_time, off_time, fade_in_time, fade_out_time, n=None, baskground)
```

```
In [ ]: ## Button (tipkalo zapravo):

        read gpiozero.button documentation using help() function. Connect button to available GPIO pin
```

```
In [ ]: import signal #(pause)

        button = Button(20)

        # wait_for_press
        # wait_for_release
        # .pin
        # is_pressed
        # pressed_time
        # when_pressed
        # when_released
        # close
        # bounce_time argument
        # when_held
```

1.5 exercise no. 3: Smart Semaphore

make a semaphore with sensoring. Make it asynchronous.

1.6 exercise no. 4: A game (again)

```
In [3]: # use random.randint(), exit()...
        # samo jedna funkcija!
```

1.7 Buzzer

```
In [ ]: p = gpiozero.Buzzer(PIN)
        p.on()
        p.off()
        p.toggle()
        beep(on_time=1, off_time=1, n=None, background=True)
```

1.8 Servo:

```
In [ ]: s = gpiozero.Servo(PIN)
        s.min()
        s.mid()
        s.max()
        s.value # -1 to 1
```

1.9 exercise no. 5: HC-SR04

just make it work