Инструменты. Python. Библиотека trikRuntime

Широколобов Илья Юрьевич

ООО «КиберТех»
Санкт-Петербург, 2020

Инструменты

1. Python
   a. Официальный сайт Python, где можно скачать интерпретатор (Python 3): https://www.python.org/
   b. Официальная документация по Python: https://docs.python.org/3/
2. TRIK Studio
   a. trikset.com/downloads#trikstudio
   b. Справочник help.trikset.com
   c. Репозиторий github.com/trikset/trikRuntime
3. Контроллер ТРИК, набор датчиков

Python

Python - высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода.

- Портируемость
- Динамическая типизация
- Объектно-ориентированное программирование
- Функциональное программирование
- Модульность
- Интроспекция

Python - операторы

```
if (self.x > 0):
      brick.motor("M3").setPower(100)
      brick.motor("M4").setPower(100)


while i < 5:
   script.wait(100)
   i+=1


for i in range(5):
   print(i)
   script.wait(100)
```

Python - числа

```
>>> a = 5
>>> print(a, "is of type", type(a))
5 is of type <class 'int'>
>>> a = 2.0
>>> print(a, "is of type", type(a))
2.0 is of type <class 'float'>
>>> a = 1+2j
>>> print(a, "is complex number?", isinstance(1+2j,
complex))
(1+2j) is complex number? True
```

Python – списки, кортежи

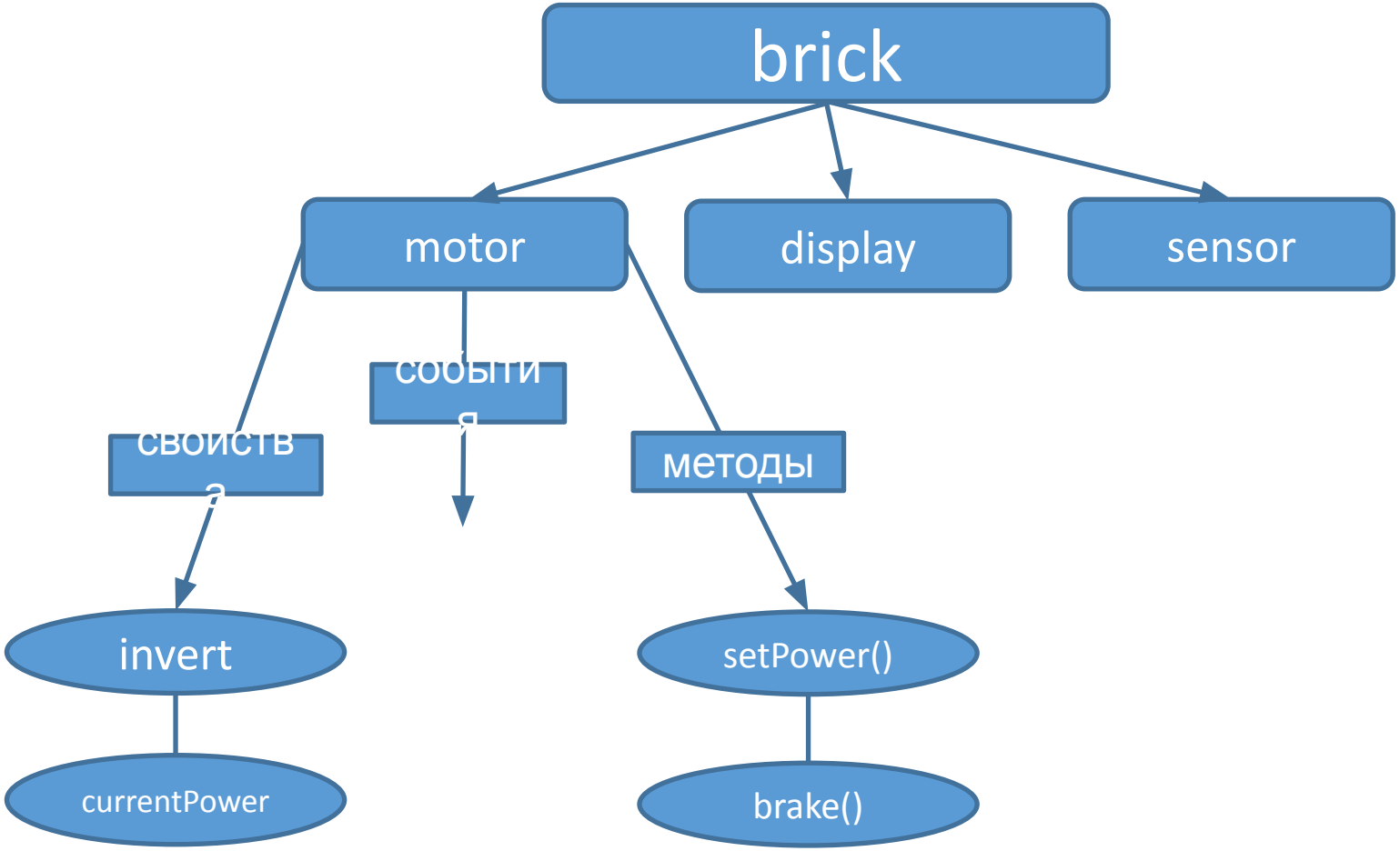
Список — упорядоченная последовательность элементов

```
a = [1, 2.2, 'python']
a[1] = 4.5
print(a)
# 1, 4.5, python
```

Кортежи, как и списки — это упорядоченные последовательности элементов, только они неизменяемы.

```
a = (1, 2.2, 'python')
a[1] = 4.5
# error
```

Объекты класса



Объекты trikRuntime

- brick
- script
- mailbox
- gamepad

Отладка

1. Вывод в консоль

```
print(“Привет, мир”) # Привет, мир!
var b = 70
var sens = brick.sensor(A1).readRawData()
print(“b=“, b, “ sens=“, sens)
# например, b=70 sens=30
```

2. Вывод на экран робота

```
brick.display().addLabel(“b=” + b, 1, 20)
brick.display().redraw() # перерисовать
```

3. Запись в файл

```
script.removeFile(“output.csv”)
script.writeToFile(“output.csv”, b + “;” + sens + “\n”)
```

Задачи

Задача 1.2.1. Вывести в консоль и на экран робота «Привет, ТРИК!»
Решение:

```
print("Привет, мир") # Привет, мир!
brick.display().addLabel("Привет, ТРИК!", 50, 50)
brick.display().redraw() # перерисовать экран
script.wait(1000)
```

Задача 1.2.2: Есть две группы роботов. Количество роботов первой группы определено в переменной **a**, а второй – **b**. Вывести на экран робота разницу групп, а также текстовую информацию, в какой группе роботов больше. Например, «Больше состав в группе A на 5 роботов»

Моторы

Управление моторами в JavaScript

```
brick.motor(M4).setPower(80)
# подать мощность 80% на M4

mRight = brick.motor(M3).setPower # переопределение метода
mRight(60)
# Подать мощность 60% на M3

brick.motor(M3).brake() # резкая остановка мотора (стопор на 500мс)
brick.motor(M3).setPower(0) # плавная остановка мотора
```

Движение по таймеру

```
mRight(80)
mLeft(80)
script.wait(3000)
# движение вперед в течение 3 секунд
# «-» зависимость от заряда батареи на реальном роботе
```

Энкодеры

Основная характеристика **cpr** - кол-во сигналов на оборот

Сбросить и считать показания энкодера

```
brick.encoder(E3).reset() # сбросить показания энкодера E3
en =brick.encoder(E3).read() # считать значение показаний энкодера E3
```

или

```
eLeft = brick.encoder(E3)
eLeft.reset() # сбросить показания энкодера E3
el = eLeft.read() # считать значение показаний энкодера E3

# считать сырое значение показаний энкодера E3
el = eLeft.readRawData()
```

Иногда удобнее работать с сырыми показаниями энкодеров.

Энкодеры

Движение по энкодерам

```
#моторы
var mLeft = brick.motor(M3).setPower
var mRight = brick.motor(M4).setPower

//энкодеры
var eLeft = brick.encoder(E3)
var eRight = brick.encoder(E4)

eLeft.reset()
eRight.reset()

var v = 80
mLeft(v)
mRight(v)

var el = 0

while(el < limit){
    el = Math.abs(eLeft.readRawData())
    script.wait(10)
# задержка в цикле необходима, чтобы считать новое показание с датчиков
}
```

Задачи

Задача 1.2.3: Реализовать функции

```
# движение вперед-назад на расстояние s со скоростью v.
# При этом знаком скорости задается направление движения
cpr = 360 # количество сигналов за оборот вала
d = 5.6 # диаметр колес робота
def run(v, s)

# поворот робота на месте на угол alpha со скоростью v.
# направление вращения может задаваться знаком v или alpha
b = 15.4 # колея робота
def rotate(v, alpha)
```

Контакты

Широколобов Илья Юрьевич

 ilya.shirokolobov@gmail.com

 +79313820455