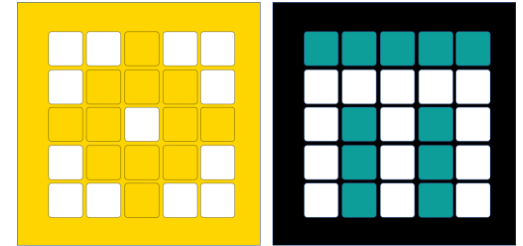


# PRIME LESSONS

By the Makers of EV3Lessons



## TIPURILE DE DATE, OPERAȚIUNILE , ȘI VARIABILELE

BY SANJAY AND ARVIND SESHAN

This lesson uses SPIKE 3 software

# OBIECTIVELE LECȚIEI

- Învățăm bazele sintaxei Python syntax (code)
- Învățăm tipurile de bază de date
- Învățăm să utilizăm operații de bază
- Învățăm variabilele de bază

# TIPURI DE BAZĂ

- Numere întregi
  - Stochează numerele întregi
- Numere raționale
  - Stochează zecimalele
- Boolean
  - Stochează **Adevărat** sau **Fals**
- Șiruri de caractere
  - Stochează șiruri de caractere, text
- Aceste tipuri sunt deja preinstalate în limbajul de programare Python

```
>> type(42)
int
>> type(42.1)
float
>> type(True)
bool
>> type(False)
bool
>> type("Hello World")
str
```

# TIPURI SPECIFICE SPIKE PRIME/MINDSTORMS

- SPIKE Prime/Mindstorms furnizează librării care definesc clase adiționale

- Aceste tipuri sunt desemnate/inițializate pentru variabile care să acceseze date sau să controleze senzori sau motoare

- Poți încărca aceste variabile pentru comenzi ca:

- Spike Prime 3:

- ```
from hub import button, light, light_matrix, motion_sensor, port, sound

import runloop, color, color_matrix, color_sensor, device, distance_sensor,
force_sensor, motor, motor_pair, orientation
```

- Mindstorms:

- ```
from mindstorms import MSHub, Motor, MotorPair, ColorSensor, DistanceSensor, A
pp
```

- Aceste tipuri de date sunt puțin diferite față de numerele întregi, șirurile de caractere, etc. dar au proprietăți similare

- Aceste tipuri specifice pentru SPIKE/MINDSTORMS și utilizarea lor vor fi explicate mai târziu în lecții.

# UTILIZAREA FUNCȚIEI DE PRINTARE

- Vom acoperi funcțiile în general într-o altă lecție. Aici doar descriem cum să utilizăm funcția de printare pentru a afișa informații pe consolă.
- „Print data” pe consolă “console”/output screen

```
>> print("Hello World")
Hello World
>> print(253.5)
253.5
```

- Notă ajutătoare: Plasarea unui # în fața unui text creează un comentariu. Codul nu va executa ceea ce e scris după acest semn.

```
>> # Comment
>> print("Hello World")
Hello World
```

# UTILIZAREA MATRICEI LED A HUB-ULUI

- Vom acoperi cum vom utiliza diferite componente prin metode specifice mai târziu în alte lecții. Aici doar descriem cum să utilizăm display-ul hub-ului pentru a arăta anumite valori.
- Acest cod afișează numărul și cuvântul „hello” pe hub-ul SPIKE

```
from hub import light_matrix
import runloop

async def main():
    # write your code here
    await light_matrix.write("Hi!")

runloop.run(main())
```

- Acest cod face același lucru în hub-ul MINDSTORMS

```
from mindstorms import MSHub
hub = MSHub()
hub.light_matrix.write(5.3)
hub.light_matrix.write("hello")
```

- Notă: Liniile “from” și “hub =” trebuie incluse doar o singură dată la începutul codului în programare. Pentru a utiliza matricea LED, este de ajuns să scriem comanda “write” mai târziu în cod.

# VARIABLELE

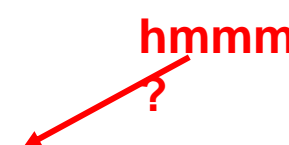
- Variabilele stocază datele
  - Acestea sunt ca variabilele de la algebră
- Datele sunt de un anumit tip
- Conținutul stocat în variabile poate fi schimbat cu o valoare diferită sau i se poate schimba chiar și tipul.
- Poți denumi o variabilă oricum dorești (în acest caz este “x”). Cu toate acestea, numele variabilei trebuie să înceapă cu o literă (în general o literă mică).

```
>> x = 7
>> print(x)
7
>> x = "hi"
>> print(x)
"hi"
>> x = "bye"
>> print(x)
"bye"
```

# OPERAȚIILE

- Poți scrie expresii matematice utilizând semnele operațiilor comune:
  - adunare (+), scădere (-), împărțire (/), înmulțire (\*), modul (%) (procent), exponent (\*\*)
  - Operatorul “//” va lua ajusta numărul la întreg. Va înlătura toate zecimalele.
- Poți aduna/adăuga numere, numere raționale, șiruri de caractere și multe altele
- Nu poți schimba diferitele tipuri de date în operații (cu excepția numerelor raționale, numerelor întregi și valorilor de tip boolean).
- Avansat: plasează “import math” la începutul programului tău pentru a avea acces la mai multe funcții; e.g. “`math.sqrt(n)`” (rădăcină pătrată)

```
>> print(5+10)
15
>> print(10/3)
3.3333333333333335
>> print(10//3)
3
>> print("ab"+"cd")
abcd
>> print(7+"ab")
TypeError: unsupported operand
type(s) for +: 'int' and 'str'
>> print(7,"ab")
7 ab
```



*Pentru cei care sunt curioși, 10/3 are un rezultat care se termină în 5 datorită a ceva numit “floating point approximation”. De fapt, calculatorul estimează când sunt implicate decimale, și aici apare o inadvertență*



# Operații cu Variabile

- Operațiile cu variabile nu sunt chiar ca la algebră
  - Expresiile sunt evaluate de la dreapta la stânga
  - Expresiile din dreapta = sunt evaluate primele, apoi alocate variabilei din partea stângă
- În exemplul din dreapta,  $x+10$  este evaluat la 20 mai întâi, apoi  $x$  este setat la valoarea de 20, stergându-se valoarea anterioară

```
>> x = 10
>> print(x)
10
>> x = x+10
20
>> # Shorthand:
>> x+=10
30
```

# PROVOCARE

- Creează o variabilă  $x$  și desemnează o valoare
- Creează o variabilă  $y$  și desemnează valoarea egală cu rădăcină pătrată din  $x$
- Afișați  $Y$  pe Hub

# SOLUȚIA PROVOCĂRII (SPIKE PRIME)

```
from hub import light_matrix
import runloop

# Importă librăria de matematică standard Python pentru a ajuta cu funcțiile
matematice

import math

async def main():
    # Aceasta creează variabila x și îi alocă valoarea 2
    x = 2

    # Aceasta creează y și îl setează ca rădăcină pătrată la x (rădăcina pătrată
    este
    # aceeași cu puterea exponențială a lui 0.5)
    y = math.pow(x, 0.5)

    # Aceasta afișează y. Trebuie mai întâi să realizezi conversia la un șir
    # Utilizăm funcția str()
    await light_matrix.write(str(y))

runloop.run(main())
```

# CREDITS

- Această lecție a fost creată de Sanjay Seshan și Arvind Seshan for SPIKE Prime Lessons
- La această lecție au contribuit membrii comunității FLL Share & Learn.
- Mai multe lecții sunt disponibile pe [www.primelessons.org](http://www.primelessons.org)
- Această lecție a fost tradusă în limba română de echipa de robotică FTC – ROSOPHIA #21455 RO20



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-nc-sa/4.0/).