

**Temat:** Hello World! Hello Codey!

Kl 7-8 Szkoła Podstawowa

**Przedmiot:** informatyka

**Autor:** Sylwester Zasoński

**Czas trwania:** 1h lekcyjna

**Cele ogólne:**

- Rozwijanie kompetencji miękkich (umiejętność pracy zespołowej, logiczne, algorytmiczne myślenie)
- Wprowadzenie języka Python

**Cele operacyjne:**

Uczeń:

- posługuje się komputerem lub innym urządzeniem cyfrowym oraz urządzeniami zewnętrznymi przy wykonywaniu zadania
- uczeń zapoznaje się z zasadami składni języka Python
- uczeń potrafi pisać proste skrypty w języku Python

**Metody:**

praca indywidualna/zespołowa, wykład

**Środki dydaktyczne:**

1. Robot Codey Rocky + kabel/adapter do połączenia
2. Komputer z zainstalowaną aplikacją mBlock

<http://www.mblock.cc/mblock-software/>

**Przebieg zajęć:**

***Celem lekcji jest poznanie języka tekstowego Python.***

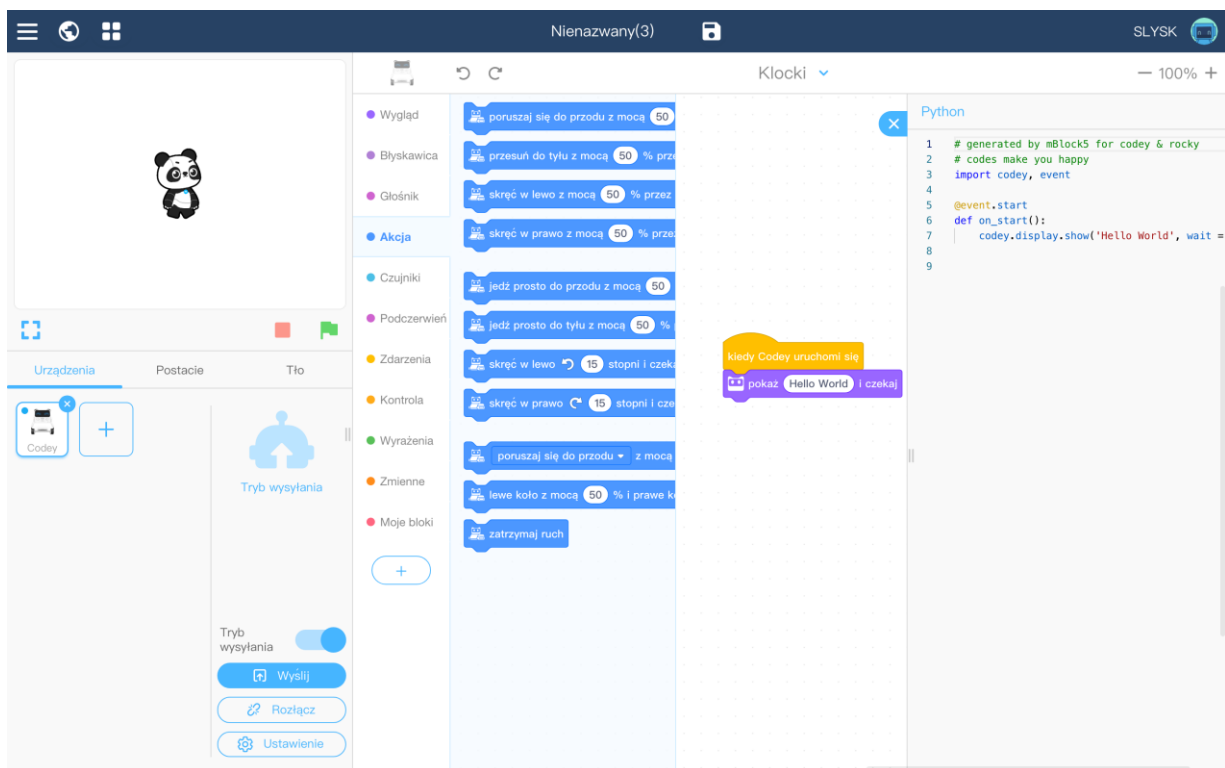
Python – język programowania wysokiego poziomu ogólnego przeznaczenia, o rozbudowanym pakiecie bibliotek standardowych, którego ideą przewodnią jest

czytelność i klarowność kodu źródłowego. Jego składnia cechuje się przejrzystością i zwięzłością

*Źródło definicji Wikipedia*

MBlock5 oparty na Scratch3.0 dostępny na komputer posiada unikalną funkcję podglądu kodu zapisanego w Scratch do języka Python (opcja dostępna w przypadku programowania robota Codey Rocky oraz duszków-obiektów na ekranie).

Chcąc podejrzeć kod dla Codey Rocky, tryb wysyłania kodu musi być włączony.



Podgląd kodu będzie widoczny po kliknięciu zakładki w prawym górnym rogu.

Kiedy poznasz swój pierwszy język tekstowy, pierwszą linijką kodu powinno być "Hello World".

W przypadku Codey Rocky tekst pokażemy na wyświetlaczu.

Taki kod będzie wyglądał następująco:

```
import codey, event
```

```
@event.start
```

```
def on_start():
```

```
    codey.display.show('Hello World')
```

**Pierwsza linijka** to importowanie bibliotek:

Codey – to głowa robota

Event – to zdarzenia, czyli kody inicjujące skrypty.

**Druga linijka** to dosłownie zapis:

@kiedy robot uruchomiony, rodzaj zdarzenia definiowany jest właściwą komendą (w tym przypadku `.start`)

Jeżeli kod miałby się uruchomić po naciśnięciu przycisku A wtedy kod wyglądał by następująco `@event.button_a_pressed`

**Trzecia linijka** rozpoczyna skrypt i jest nierozłączną częścią kodu z drugiej linijki.

Poniżej **czwarta linijka** zaczyna się od wcięcia, robi się je tabulatorem. Oznacza to, że kod poniżej będzie wykonywany w tym skrypcie i będzie inicjowany `@event.start`

```
1  import codey, event
2
3  @event.start
4  def on_start():
5      codey.display.show('Hello World')
```

Spróbujmy teraz dodać dźwięk:

```
codey.speaker.play_melody('hello.wav')
```

**Hello.wav** - to nazwa odtwarzanego pliku dźwiękowego. Pozostałe wbudowane dźwięki poznasz używając podglądu scratch->python, np.:

**hi.wav**

**bye.wav**

**yeah.wav**

ltd.

Codey na spodzie wyświetlacza posiada również diodę LED RGB, kolor diody ustawisz podając parametry RGB.

```
codey.led.show(255, 0, 0)
```

**RGB** – jeden z modeli przestrzeni barw, opisywanej współrzędnymi **RGB**. Jego nazwa powstała ze złożenia pierwszych liter angielskich nazw barw: R – red (czerwonej), G – green (zielonej) i B – blue (niebieskiej), z których model ten się składa.

Zródło wikipedia

Istnieje też możliwość zaprogramowania diody LED RGB, aby świeciła przez określony czas, wówczas ostatni parametr będzie odpowiadać za czas wyświetlania, na poniższym przykładzie wynosi ona 1.

```
codey.led.show(255, 0, 0, 1)
```

Ważne:  
Dodając czas musisz również na początku kodu dodać:  
**Time**  
Ta linijka będzie wyglądać następująco:  

```
import codey, time, event
```

Jeżeli dioda nie została ustawiona na wyłączenie czasowe (np. Po 5 sekundach) możesz użyć komendy wyłączającej.

```
codey.led.off()
```

Pamiętaj! Do pracy z robotem wystarczy sama głowa, która jest w pełni funkcjonalna bez podwozia. Jeżeli jednak zdecydujesz się dodawać kod odpowiedzialny za ruch lub programujący czujniki znajdujące się w podwoziu o nazwie Rocky, musisz wyedytować kod i dopisać na początku **rocky**,

Cała linia wygląda tak:

`import codey, rocky, time, event`

Podstawowe kody odpowiedzialne za ruch to:

`rocky.forward(50, 1)` - jedź do przodu z mocą 50% przez 1 sekundę

`rocky.backward(50, 1)` - jedź do przodu z mocą 50% przez 1 sekundę

`rocky.turn_left(50, 1)` - jedź do przodu z mocą 50% przez 1 sekundę

`rocky.turn_right(50, 1)` - jedź do przodu z mocą 50% przez 1 sekundę

Pierwsza wartość w nawiasie np.(50) to moc silników wyrażana w procentach, druga np.(1) to czas wykonywania polecenia.

Znasz już podstawowe komendy dla Codey Rocky w Pythonie. Spróbuj zatem napisać własny prosty kod, w którym robot wyświetli przywitanie, odtworzy dźwięk czy przejedzie kawałek.