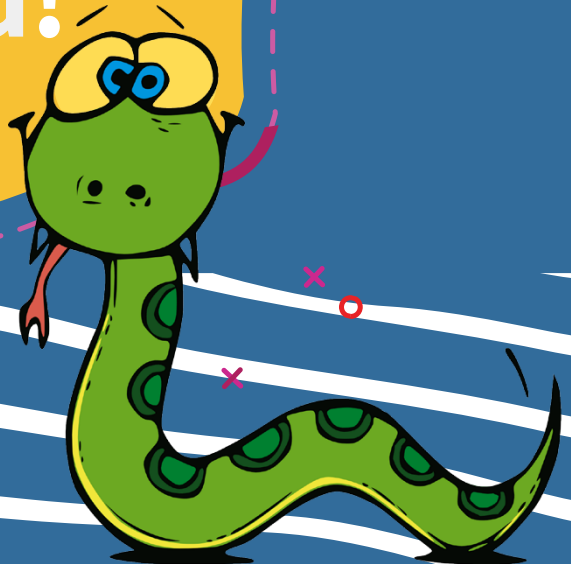


PROGRAMMING WITH python

'Hello,
world!'



Lekcja # 2

Grafika żółwia Proste figury

Spis treści

Wstęp.....	3
Twój pierwszy program	3
Rysowanie prostokąta.....	5
Rysowanie okręgu	11

Wstęp

Python posiada ciekawą funkcję rysowania - żółw.

Za pomocą komend możesz nim sterować: na przód, w tył, lewo, prawo. Domyślnie kursor jest reprezentowany jako 🐢. Możemy to łatwo zmienić za pomocą jednej linijki kodu i teraz się tego nauczymy.

Zaczynajmy!



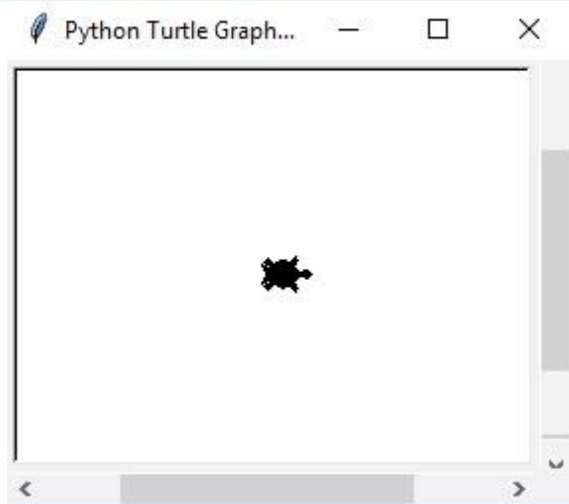
Twój pierwszy program

Każda figura geometryczna może zawierać punkty, linie i kształty, na przykład kwadrat, trójkąt, prostokąt, sześciokąt. W tej lekcji nauczymy się jak narysować każdy z nich z wykorzystaniem biblioteki turtle. Możesz poczytać o tym module w dokumentacji: [link](#).

Python posiada wiele ogólnodostępnych bibliotek i jest niemożliwe zapamiętać do czego służy każda z nich, dlatego stworzono ich dokumentację. Jest kilka najczęściej używanych modułów i poznamy je na tym kursie. Obecnie jesteśmy zainteresowani biblioteką turtle.

Początkowo utwórz nowe okno i wyświetl żółwia (Rysunek 1):

```
from turtle import*  
window = Screen()  
Pen()  
shape("turtle")  
window.exitonclick()
```



Rysunek 1

W kodzie zaimportowaliśmy **żółwia** (**from turtle import***), więc możemy teraz na nim pracować. Następnie stworzyliśmy nowe okno `window = Screen()`, więc wszystkie akcje będą wykonywane w nim. Wyświetliliśmy wskaźnik

`Pen()` i ustawiliśmy jako żółw `shape("turtle")`.

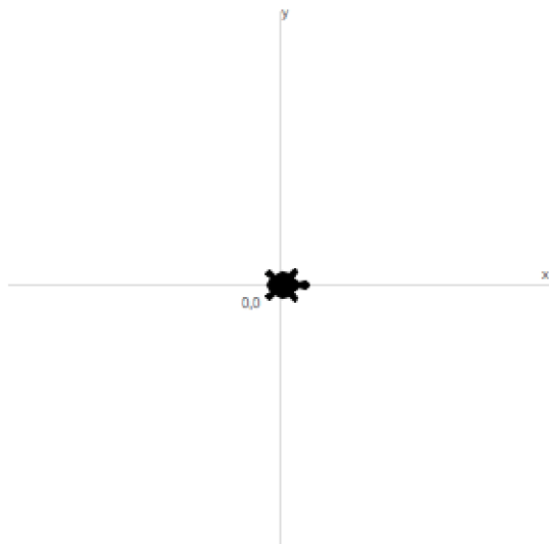
Możesz zmienić te preferencje poprzez zmianę wartości `"turtle"` na jedną z sugerowanych: `"arrow"`, `"circle"`, `"circle"`, `"triangle"`, `"classic"`. Spróbuj to zrobić i wybierz najbardziej Ci odpowiadającą.

Dodaliśmy również `window.exitonclick()` więc nowe okno zamknie się po kliknięciu na nie.

Rysowanie prostokąta

Nasz żółw może się poruszać i tworzyć figury!

Proces rysowania jest bardzo prosty: obiekt się porusza i zostawia za sobą ślad. Wyobraź sobie że żółw jest w punkcie `0`, reprezentacji w osi `x` i `y`, czyli jesteśmy w środku układu współrzędnych (Rysunek 2).



Rysunek 2

Nauczmy naszego żółwia rysować prostokąt
(zobacz Rysunek 3 na stronie 7):

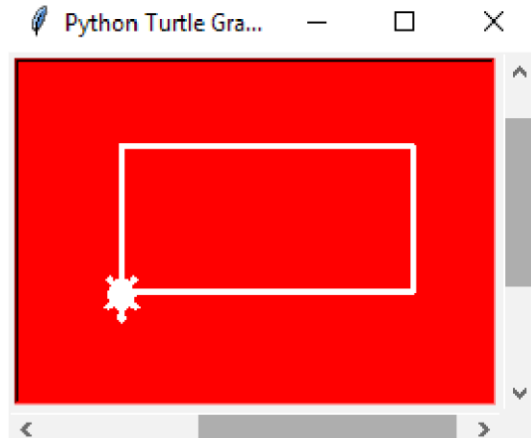
```
from turtle import*

window = Screen()

reset()
shape("turtle")
bgcolor("red")
color("white")

speed(2)
pensize(3)
forward(150)
left(90)
forward(75)
left(90)
forward(150)
left(90)
forward(75)

window.exitonclick()
```



Rysunek 3

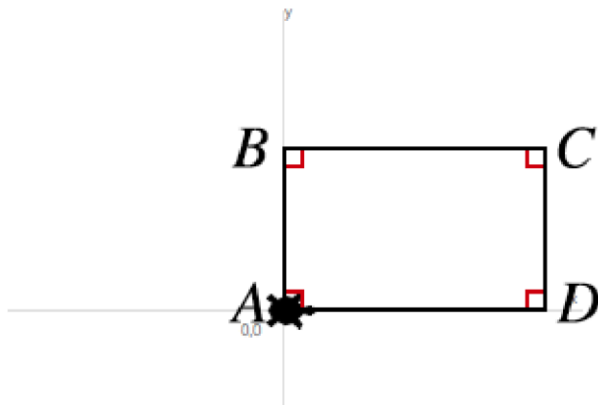
Rozważmy co robi nasz kod.

Pierwszą rzeczą jest wyczyszczenie ekranu przed rysowaniem `reset()`. Dodaliśmy również kolor tła `bgcolor("red")` i zmieniliśmy kolor linii `color("white")`.

Aby zmienić prędkość żółwia używamy komendy `speed(2)`. Im mniejsza liczba tym wolniej się porusza. Spróbuj zmienić wartości na `0.5` do `10` i zobacz co się zmieni.

Aby linia nie była za cienka, ustawiliśmy jej grubość na 3: `pensize(3)`. Możesz zmienić również ten parametr.

Prostokąt to figura której dwa przeciwległe boki są sobie równe, np. $AB=CD$ i $BC=DA$ i wszystkie kąty są proste (Rysunek 4).



Rysunek 4

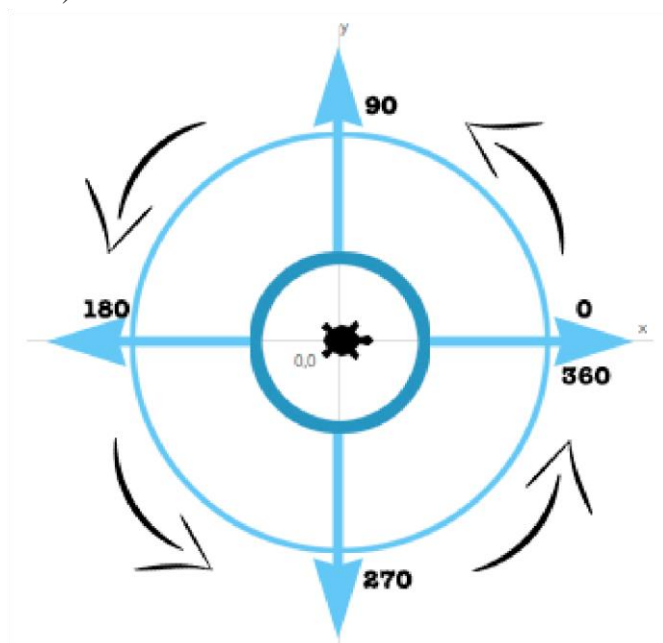
Dystans w jakim porusza się żółw jest mierzony w pikselach, `forward(75)` oznacza ruch na przód o 75 pikseli. **Pixel jest to najmniejsza część rysunku.** To co widzimy na monitorze, to piksele która składają się w jeden rysunek (Rysunek 5).



Rysunek 5

Ruch żółwia jest uzależniony od komend `forward()` i `back()`, gdzie odległość wyrażamy w pikselach, jak wspomnieliśmy wyżej.

Żółw narysował prostą linię, lecz teraz potrzebuje się obrócić. Możemy użyć funkcji `left()` lub `right()` aby określić kierunek obrotu (lewo lub prawo). Parametr obrotu określamy w stopniach. **Stopnie są miarą kąta i mogą przyjmować wartości od 0° do 360° .** (Rysunek 6).



Rysunek 6

Sprawmy aby żółw poruszał się w prawo za pomocą prostej zmiany funkcji z `left(90)` na `right(90)` w kodzie.

Nauczyliśmy się zatem rysować proste figury używając **Pythona** i biblioteki turtle

Rysowanie okręgu

W Pythonie jest również możliwość narysowania okręgu (zobacz Rysunek 7 na stronie 11). Aby to zrobić potrzebujesz tylko jednej funkcji `circle()`.

Zróbmy to za pomocą kodu:

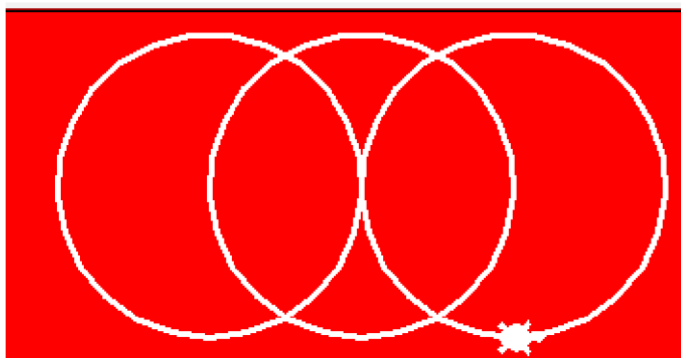
```
from turtle import*

window = Screen()
reset()

shape("turtle")
bgcolor("white")
color("red")

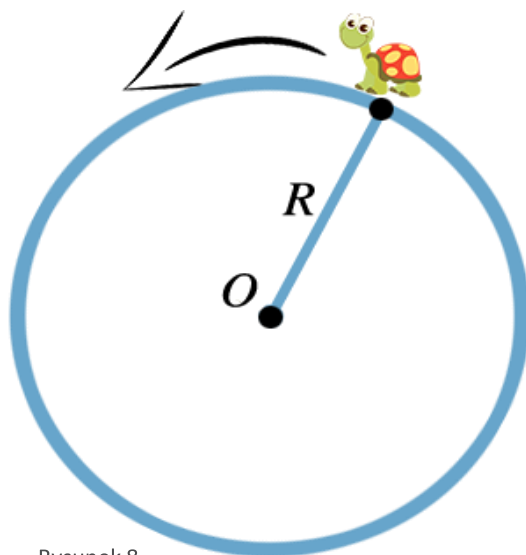
speed(2)
pensize(3)
circle(80)
```

```
penup()  
forward(80)  
pendown()  
circle(80)  
penup()  
forward(80)  
pendown()  
circle(80)  
  
window.exitonclick()
```



Rysunek 7

Użyliśmy funkcji `circle(80)`, gdzie `80` jest promieniem okręgu (`R`), na poniższym przykładzie jest to odległość łącząca punkt (`O`) i dowolny punkt leżący na obwodzie (Rysunek 8).



Rysunek 8

Zobaczmy jak zadziałało użycie funkcji `penup()` i `pendown()`.

Nasz żółw jest długopisem, zatem jeśli podniesiemy go do góry, nie będzie rysował. Dzięki temu możemy przesunąć żółwia w dowolnym kierunku, nie zostawiając śladu.

Skomentuj linie poleceń z tymi funkcjami: `#penup()` i `#pendown()` i uruchom kod.



Lekcja # 2

Grafika żółwia

Proste figury

© STEP IT Academy

www.itstep.org

Wszelkie prawa do chronionych zdjęć, audio i wideo należą do ich autorów lub prawnych właścicieli. Fragmenty prac są wykorzystywane wyłącznie w celach ilustracyjnych w zakresie uzasadnionym celem w ramach procesu edukacyjnego oraz w celach edukacyjnych zgodnie z art. 1273 ust. 4 Kodeksu cywilnego Federacji Rosyjskiej oraz art. 21 i 23 Ustawy Ukrainy "O prawie autorskim i prawach pokrewnych". Zakres i metoda cytowanych prac są zgodne z normami, nie kolidują z normalnym wykorzystaniem utworu i nie naruszają uzasadnionych interesów autorów i podmiotów praw autorskich. Cytowane fragmenty utworów można zastąpić alternatywnymi, niechronionymi analogami i jako takie odpowiadają kryteriom dozwolonego użytku. Wszelkie prawa zastrzeżone. Wszelkie powielanie, w całości lub w części, jest zabronione. Zgoda na wykorzystanie utworów i ich fragmentów jest dokonywana z autorami i innymi właścicielami praw. Materiały z tego dokumentu mogą być używane tylko z linkiem do zasobów. Odpowiedzialność za nieuprawnione kopiowanie i komercyjne wykorzystanie materiałów określa się zgodnie z obowiązującym ustawodawstwem Ukrainy.