



# **Zaszyfruj się z Pythonem!**

inż. Laura Dymarczyk  
inż. Adrian Rupala



## **Dlaczego Python?**

- Język wysokiego poziomu, ogólnego zastosowania
- Prosta i czytelna składnia
- Darmowe biblioteki, menedżer pakietów, Open Source



## **Trochę teorii...**

- Nie wymusza jednego stylu programowania
- Pozwala programować obiektowo, strukturalnie oraz funkcyjnie
- Język interpretowany

The image features a light gray background with a large, dark green, textured brushstroke that runs horizontally across the center. Overlaid on this brushstroke is a white rectangular frame. Inside the frame, the text "Napiszmy trochę kodu!" is written in a bold, white, sans-serif font, centered both horizontally and vertically.

**Napiszmy trochę  
kodu!**



**Powitajmy  
nowy świat**

Witaj świecie! A może coś innego?

```
>>> print("Hello world!")
```



## **Zdefiniujmy coś...**

- Przypisanie do określonej wartości czegoś innego

```
>>> dzban = "słowo roku 2019"
```

```
>>> print(dzban)
```



# Funkcje

- Stwórzmy coś bardziej zaawansowanego.

```
>>> def nazwa_funkcji(parms):  
    <napisz tu fajny kod>  
    <więcej fajnego kodu>  
    return <fajna wartość>
```

```
>>> print(nazwa_funkcji(parms))
```



## Pętla for

- >>> for <zmienna> in <wartość>:  
    <twój fajny kod który coś robi>
- >>> ... in range(start, stop, krok):  
    ...





## Pętla while

- >>> i = 1
- >>> while i = <wartość>:  
    <dalej fajny kod>  
    i += 1

The image features a light gray background with two broad, horizontal blue brushstrokes. A white rectangular frame is centered over the intersection of these strokes. Inside the frame, the text "Zaczniemy szyfrowanie!" is written in a bold, white, sans-serif font, split across two lines.

**Zaczniemy  
szyfrowanie!**



## **Szyfr Cezara**

- Przykład szyfru podstawieniowego
- Nazwa pochodzi od Juliusza Cezara, który używał tej techniki do komunikowania się z przyjaciółmi
- Każda litera tekstu zaszyfrowana jest inną, oddaloną od niej o stałą liczbę pozycji w alfabecie

## Przykład...

- Parametr przesunięcia 3:

Alfabet:

AĄBCĆDEĘFGHIJKLŁMNŃOÓPRSŚTUWYZŻŻ

Szyfr:

CĆDEĘFGHIJKLŁMNŃOÓPRSŚTUWYZŻŻAĄB

Tekst jawny:

MĘŻNY BĄDŹ, CHROŃ PUŁK TWÓJ I SZEŚĆ FLAG

Tekst zaszyfrowany:

OH BÓŹ DĆFA, EKTRP ŚZŃM YŹSŁ L UAGWĘ INCJ



## **Jak się za to zabrać?**

- Musimy zdefiniować nasz alfabet
- Musimy określić ilość liter w alfabecie
- Odczytać i zapisać plik tekstowy z tekstem
- Napisać funkcję odpowiedzialną za szyfrowanie tekstu (odpowiednie przesuwanie liter)

A large, expressive blue brushstroke graphic that sweeps across the left side of the slide, partially framing the title.

# Zabawa z plikami

- ```
>>> plik = open(nazwa_pliku, 'parametr')
>>> plik.read()
>>> plik.write()
>>> plik.close()
```
- Parametr może przyjmować różne wartości, np:
  - r - dla odczytu
  - w - dla zapisu
  - b- w postaci binarnej



## Co jeszcze się przyda?

- `>>> len(dane)` - zwraca ilość (długość) obiektów w strukturze
- `>>> wartosc.find(litera)` - zwraca indeks pierwszej szukanej wartości w strukturze





“

*Stay Hungry  
Stay Foolish*



**Partnerzy**

 ŚLĄSKI  
3. FESTIWAL  
NAUKI  
KATOWICE

 WHITEASTER

 | kóło  
naukowe  
informatyków