



# Pystart.pl

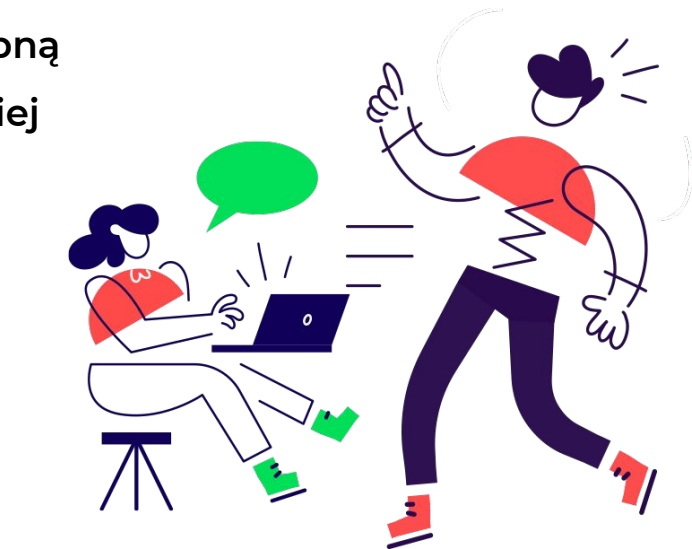
## Czas na generatory!

lekcja dwudziesta pierwsza

# PyStart #21 Czas na generatory

## Wprowadzenie

- Generator pozwala funkcji zwracać wielokrotnie
- Dzięki generatorowi funkcja może zwracać nie czekając na pełen wynik
- W ten sposób znacznie optymalizujemy pamięć potrzebną do obsłużenia naszego żądania + program działa szybciej
- Nowe słówko **yield**, to taki return



# PyStart #22 Czas na generatory

## Jak to wygląda bez generatora?

```
1 def get_doubles(numbers: range):
2     results = []
3     for i in numbers:
4         results.append(i+i)
5
6     return results
7
8
9 for n in get_doubles(range(1, 11)):
10     print(f'Wynik wynosi {n}')
11
```

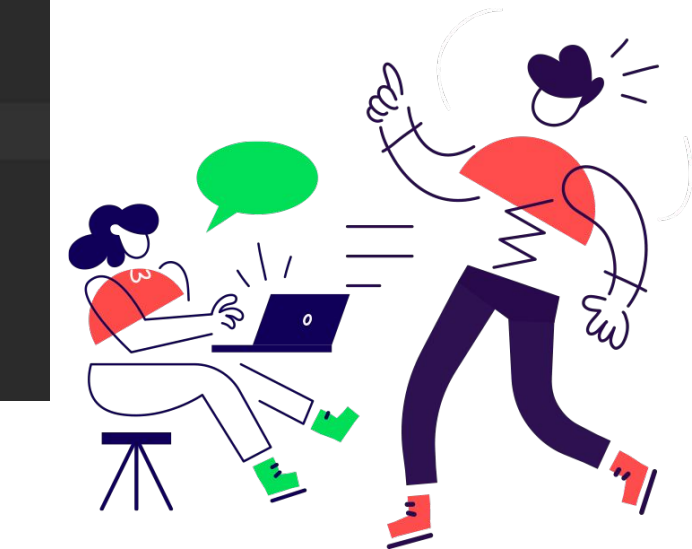
Trzeba czekać aż wszystkie wartości zostaną “przemielone”  
Trzeba stworzyć wielką listę results.. tylko po to by ją zwrócić



# PyStart #22 Czas na generatory

## Przykład jeden generator

```
1  def get_doubles(numbers: range):
2      for i in numbers:
3          print(f'Iteruje.. liczę dla {i}')
4          yield i + i
5
6
7  results = get_doubles(range(111, 222))
8  print(next(results))
9  print(next(results))
10 print(next(results))
```



# PyStart #22 Czas na generatory

## Przykład jeden generator

```
1 def get_doubles(numbers: range):
2     for i in numbers:
3         print(f'Iteruje.. liczę dla {i}')
4         yield i + i
5
6
7 for n in get_doubles(range(1, 11)):
8     print(f'Wynik wynosi {n}')
9
```



# PyStart #22 Czas na generatory

## Generator liczb pierwszych

```
1  from math import sqrt, floor
2
3
4  def is_prime(n):|...
10
11
12  def generate_prime_numbers():
13      n = 2
14      while True:
15          if is_prime(n):
16              yield n
17
18          n += 1
19
20
21  for number in generate_prime_numbers():
22      print(f'To liczba pierwsza {number}')
23
```



# PyStart #22 Czas na generatory

## Generator liczb pierwszych



A jak go przerwać?



# PyStart #22 Czas na generatory

## Generator liczb pierwszych

```
21 primes = generate_prime_numbers()
22 for number in primes:
23     print(f'To liczba pierwsza {number}')
24     if len(str(number)) >= 6:
25         print('Tyle starczy')
26         primes.close()
27
```

