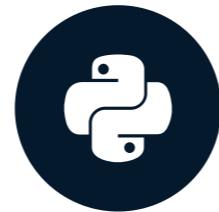


What is Python?

INTRODUCTION TO PYTHON FOR DEVELOPERS



Jasmin Ludolf

Senior Data Science Content Developer

What to expect

- Learn all about Python
- Think like a developer
- Why is Python popular?
- Write and run code



Why learn Python?

- General-purpose programming language
- Used for apps, automation, websites, and more
 - Facebook, Netflix, and Spotify
- Clear, readable syntax

```
print("Hello!")
```

- Large developer community
- Free and open-source
- Anyone can use it!



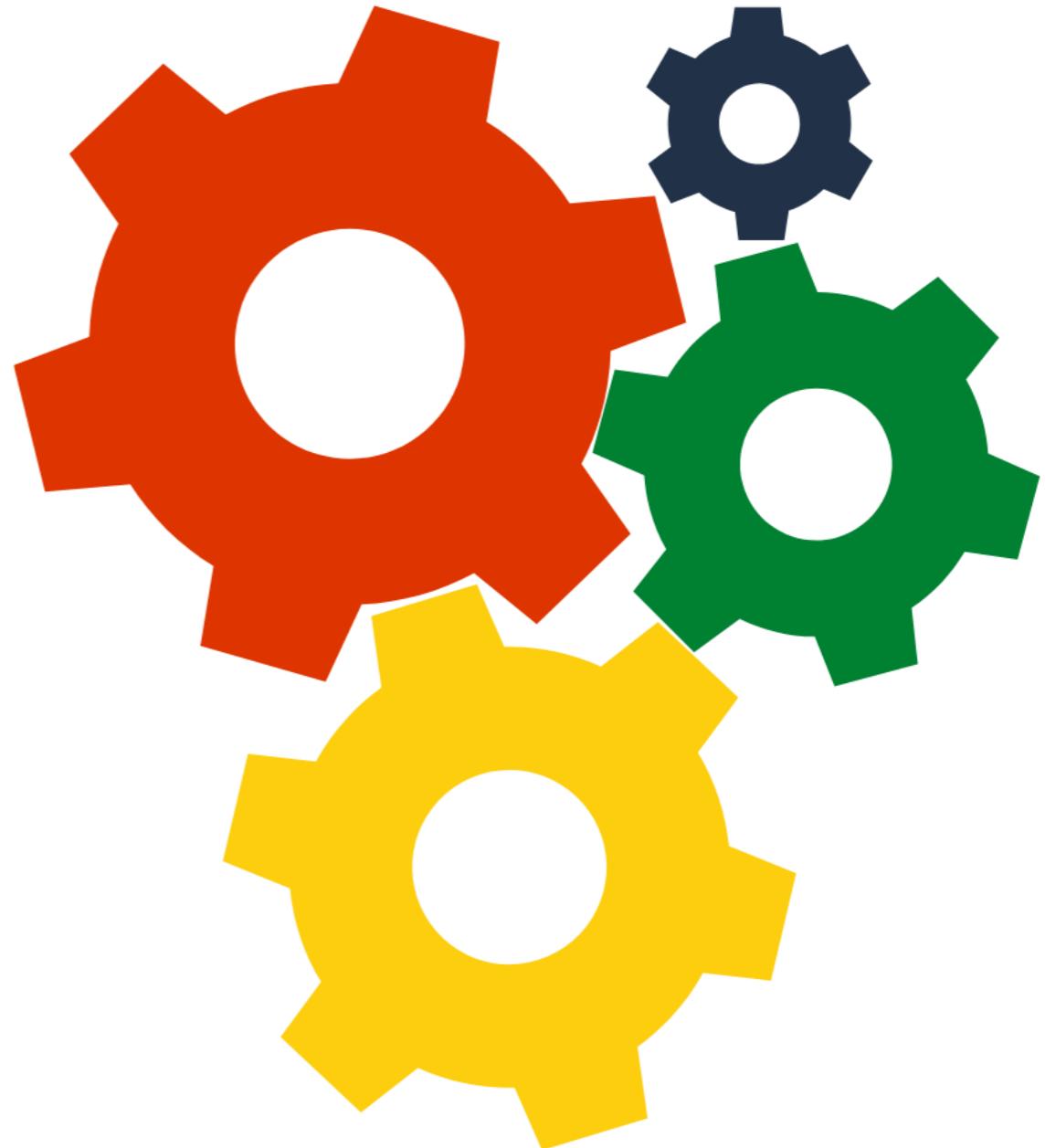
¹ Image by Godfrey Atima, Pexels

Build a recipe scaler

Python script: file with python code we run

Development practices:

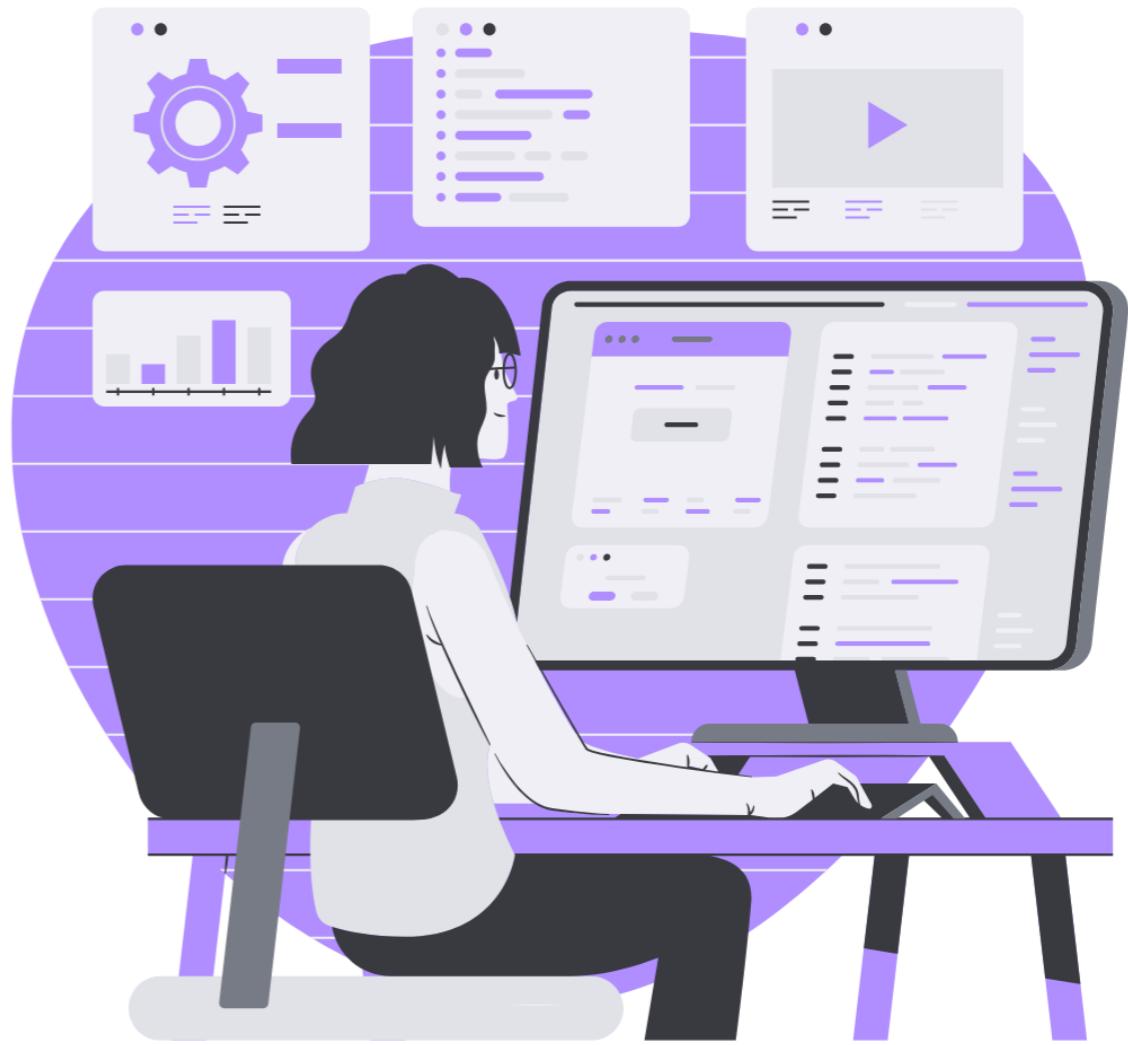
- Start small
- Iterating
- Testing
- Documenting



By the end of the course

Learn about:

- Variables
- Data types
- Control flow
- Code structure



The print function

- Built-in Python function
- Displays code results
- Used for testing

```
print()
```

The print function

- Built-in Python function
- Displays code results
- Used for testing

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

```
Hello, world!
```

```
print('Hello, world!')
```

```
Hello, world!
```

Documenting with comments

- Comments explain code
- Start with `#` symbol
- Appear in different color
- Do not affect code

```
# This is a code comment  
# The below is a print function  
print("Hello, world!")
```

Let's practice!

INTRODUCTION TO PYTHON FOR DEVELOPERS

Python variables, strings, and integers

INTRODUCTION TO PYTHON FOR DEVELOPERS



Jasmin Ludolf

Senior Data Science Content Developer

Understanding variables

- Store and reuse information
- Easy to read and update
- Avoid typing things over and over



¹ Image by reginafosterphotos

Data types

- The kind of value
 - Number
 - Text
 - True/False
- Python assigns data types automatically

Text

0123456789

True or False

String data types

- **Strings:** text values
- Can contain letters, numbers, and punctuation

"Hello, world!"

"20-minute pasta recipe"

String data types

- **Strings:** text values
- Can contain letters, numbers, and punctuation
- Use single or double quotes
- Double quotes for words with apostrophes

```
"Chef's secret seasoning"
```

```
'Chef's secret seasoning'
```

String variable

```
ingredient_name
```

- Use lowercase and underscores for spaces

String variable

```
ingredient_name =
```

String variable

```
ingredient_name = "Tomatoes"
```

Integer data types and variables

- **Integer:** whole numbers
- No quotation marks are needed

```
ingredient_quantity = 2
```

Printing variables

```
print(ingredient_name)
```

Tomatoes

```
print(ingredient_quantity)
```

2

Adjusting variables

```
print(ingredient_quantity)
```

2

```
ingredient_quantity = 1  
print(ingredient_quantity)
```

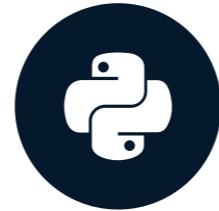
1

Let's practice!

INTRODUCTION TO PYTHON FOR DEVELOPERS

Common data types

INTRODUCTION TO PYTHON FOR DEVELOPERS



Jasmin Ludolf

Senior Data Science Content Developer

Decimal data types

- **Integers:** whole numbers
- **Floats:** numbers with decimals, e.g. 1.5
- No quotation marks needed

```
new_ingredient_quantity = 1.5
```

Boolean data types

- Boolean: True or False
- Great for storing yes/no information
- Capitalized
- No quotation marks

```
is_in_stock = True
```

```
is_in_stock = False
```

The type function

- Data types:
 - Strings
 - Integers
 - Floats
 - Booleans

```
type()
```

- Tells us the data type

The type function

String

```
ingredient_name = "tomatoes"  
print(type(ingredient_name))
```

```
<class 'str'>
```

Float

```
new_ingredient_quantity = 1.5  
print(type(new_ingredient_quantity))
```

```
<class 'float'>
```

Integer

```
ingredient_quantity = 2  
print(type(ingredient_quantity))
```

```
<class 'int'>
```

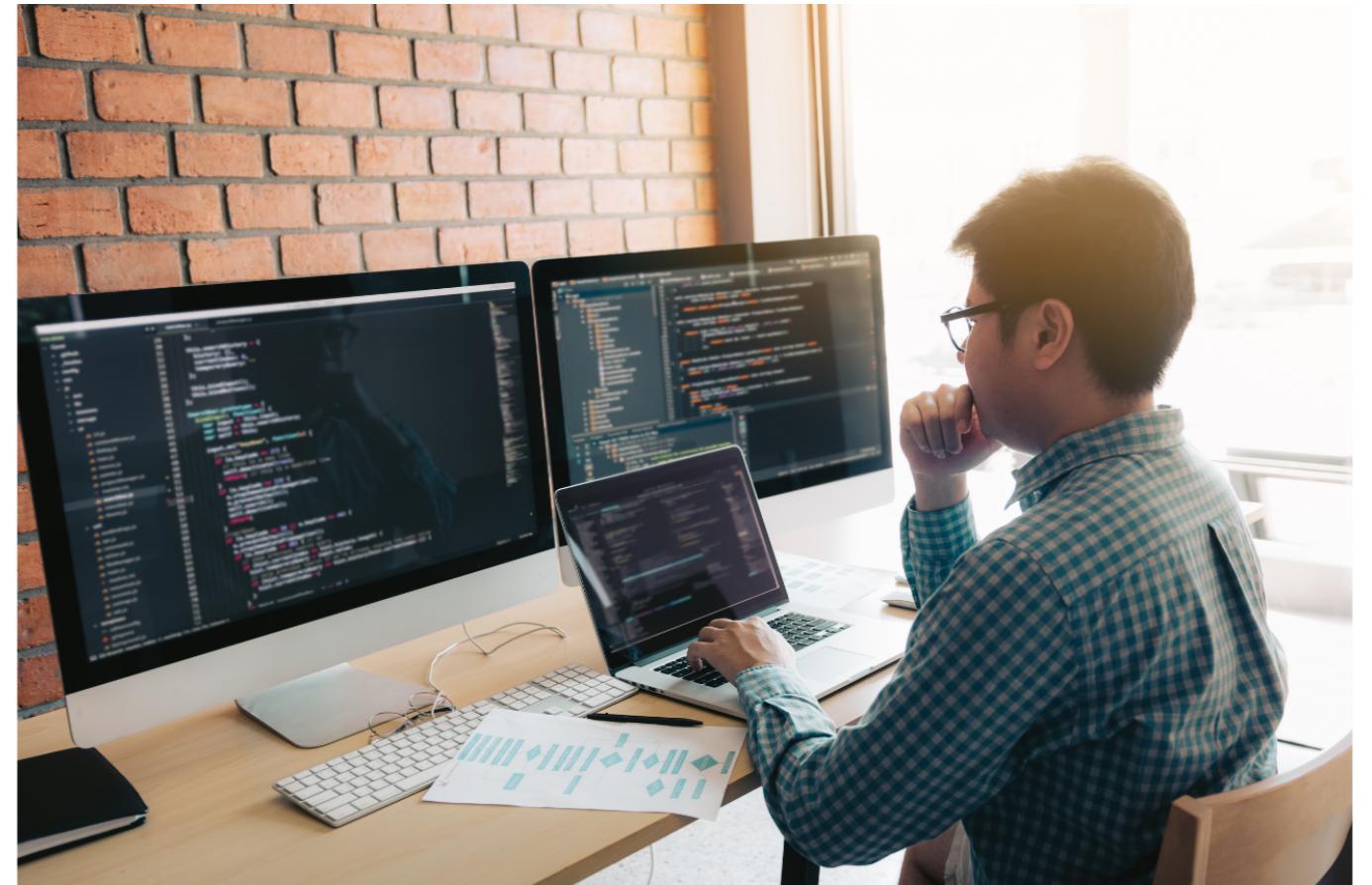
Boolean

```
is_in_stock = True  
print(type(is_in_stock))
```

```
<class 'bool'>
```

Understanding data types

- Understand what code is doing and why
- Use `type()` to identify the type
- Determine what operations we can perform



¹ Image by wutzkoh

Operators

- Symbols called operators to perform computations

Arithmetic operators:

- `+` for addition
- `-` for subtraction
- `*` for multiplication
- `/` for division

Numbers:

- Work like regular math

```
print(2 + 1.5)
```

```
3.5
```

Strings:

- Only `+` and `*` work

```
"Hi" + "There" = "HiThere"
```

```
"Hi" * 3 = "HiHiHi"
```

Let's practice!

INTRODUCTION TO PYTHON FOR DEVELOPERS