De cero a ciencia de DATOS

PYTHON

WHAT IS PYTHON?

PYTHON IS AN INTERPRETED LANGUAGE WHOSE PHILOSOPHY FOCUSES ON A SYNTAX THAT FAVORS READABLE CODE.

-OBJECT ORIENTED

-IMPERATIVE PROGRAMMING

-DYNAMIC PROGRAMMING

-MULTIPLATFORM

WHAT IS PYTHON?

IT IS WIDELY USED FOR RAPID APPLICATION DEVELOPMENT AND AUTOMATION.

PYTHON'S SIMPLE, EASY-TO-LEARN SYNTAX EMPHASIZES READABILITY AND REDUCES THE COST OF PROGRAM MAINTENANCE.

IT IS OPEN-SOURCE AND MANAGED BY THE PYTHON SOFTWARE FOUNDATION.

ORIGINALLY CONCEIVED IN THE LATE 80S BY GUIDO VAN ROSSUM IN THE

NETHERLANDS.

WHY PYTHON?

- MOST POPULAR PROGRAMMING LANGUAGE, ACCORDING TO MULTIPLE RANKINGS (IEEE, TIOBE, STACK OVERFLOW).
- THE DEFAULT LANGUAGE FOR MACHINE LEARNING, DATA SCIENCE, AND AI.
- INCREASED PRODUCTIVITY DUE TO ITS EASE OF USE
- NO COMPILATION STEP
  EDIT-TEST-DEBUG CYCLE IS FAST.

WHO USES IT AND HOW?

PYTHON IS NOT JUST EASY TO LEARN—IT'S INCREDIBLY POWERFUL.

MAJOR COMPANIES LIKE GOOGLE, META, MICROSOFT, OPENAI, NETFLIX, SPOTIFY, AND NASA RELY ON PYTHON.

WHO USES IT AND HOW?

DROPBOX'S BACKEND AND SYNCHRONIZATION LOGIC HEAVILY RELY ON PYTHON.

WHO USES IT AND HOW?

GOOGLE STILL FOLLOWS THE EARLY ENGINEERING PRINCIPLE:

"USE PYTHON WHERE WE CAN, C++ WHERE WE MUST."

IT POWERS AI MODELS, AUTOMATION, AND BACKEND SERVICES.

Google

WHO USES IT AND HOW?

SPOTIFY USES PYTHON FOR DATA ANALYTICS, ML MODELS, AND API SERVICES.

Spotify®

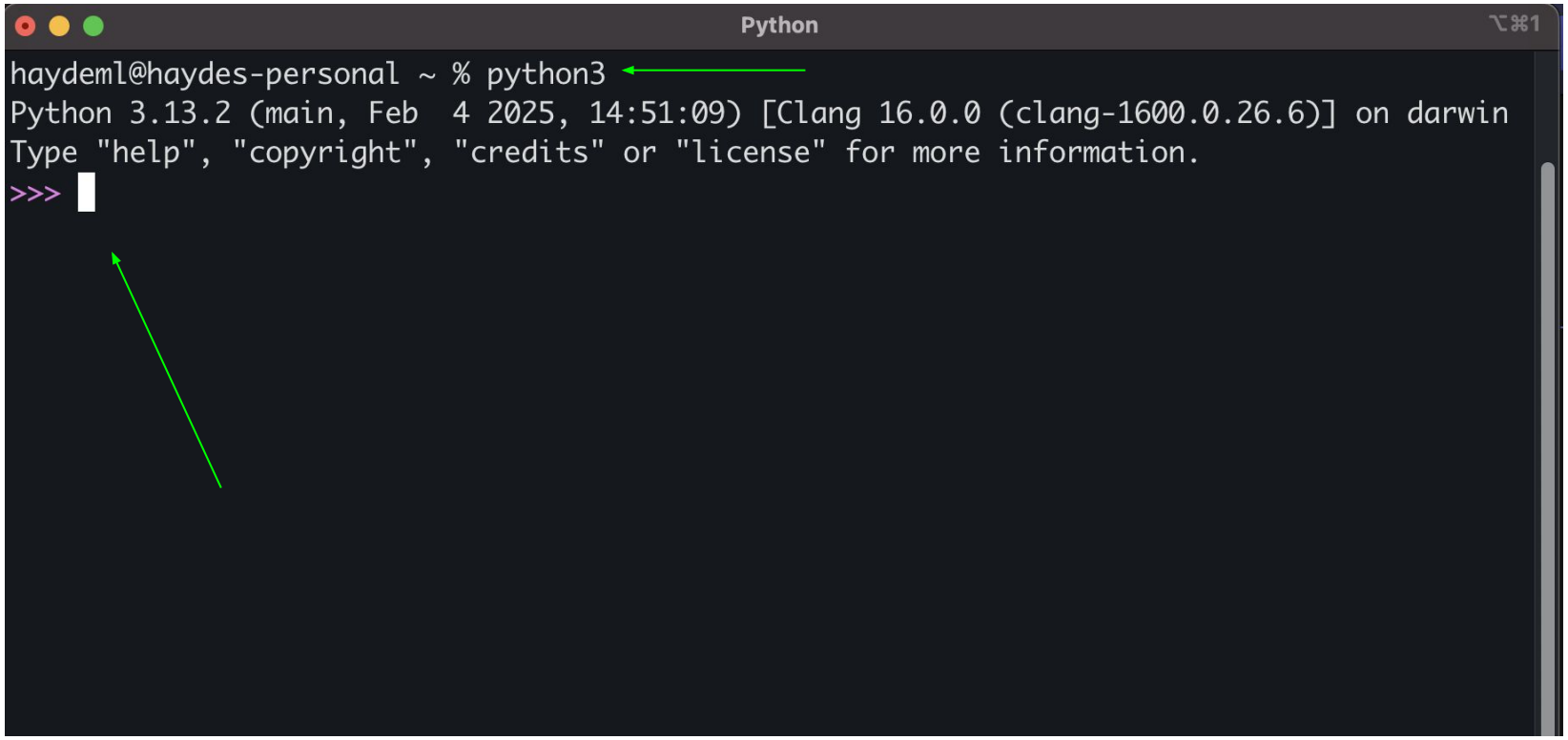
WHO USES IT AND HOW?

NETFLIX GIVES DEVELOPERS AUTONOMY IN CHOOSING THE BEST LANGUAGE BUT USES PYTHON HEAVILY FOR REAL-TIME ANALYTICS, RECOMMENDATIONS, AND PERSONALIZATION.

SYNTAX

INTERACTIVE MODE PROGRAMMING

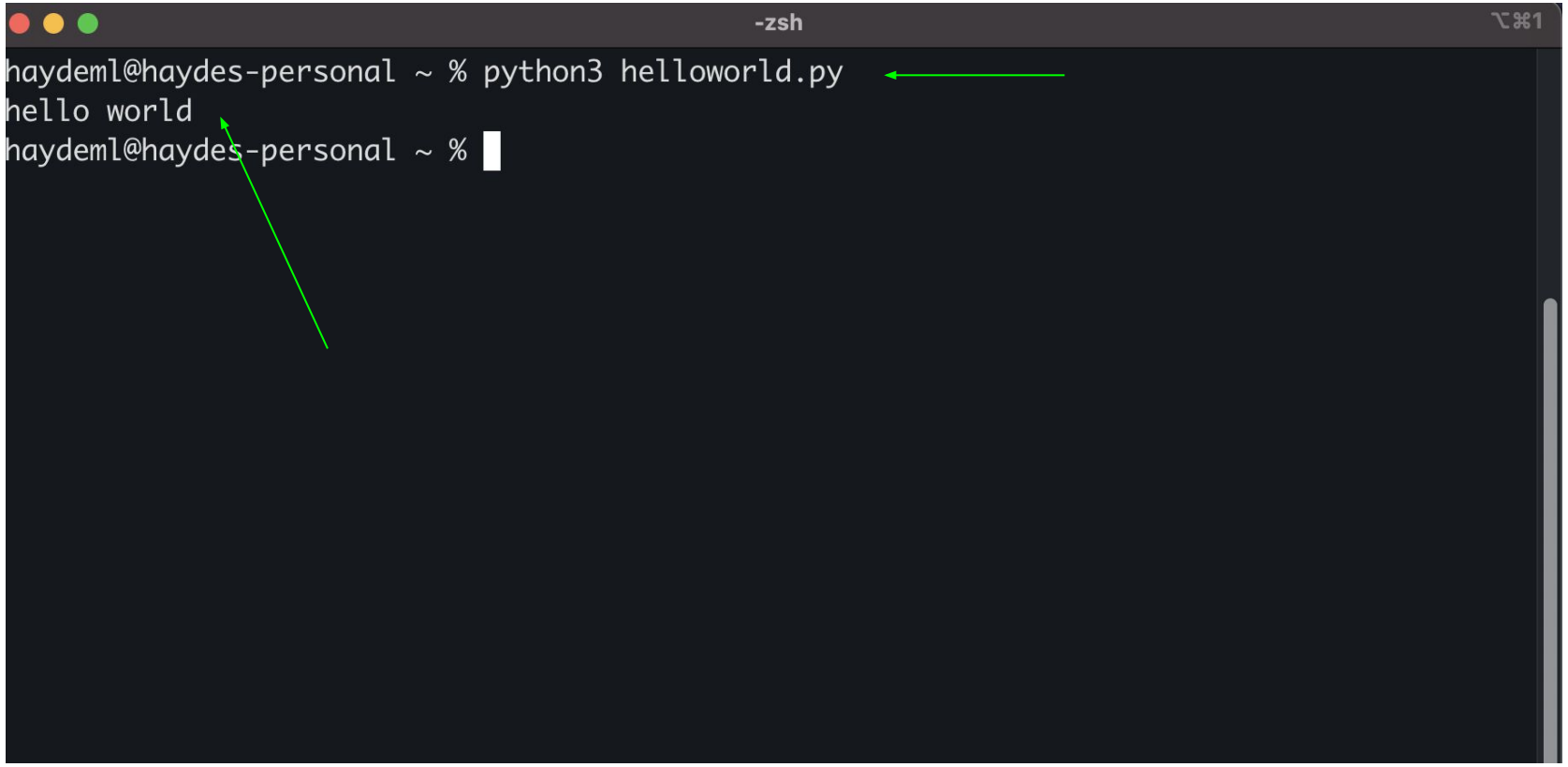
INVOKING THE INTERPRETER WITHOUT PASSING A SCRIPT FILE AS A PARAMETER.



SYNTAX

SCRIPT MODE PROGRAMMING

RUNNING A PYTHON SCRIPT FILE EXECUTES IT UNTIL COMPLETION, AFTER WHICH THE INTERPRETER EXITS.



SYNTAX

PYTHON IDENTIFIERS

NAMES USED FOR VARIABLES, FUNCTIONS, CLASSES, AND MODULES.

MUST START WITH A LETTER (A-Z, `a-z`) OR UNDERSCORE (_), FOLLOWED BY LETTERS, DIGITS (0-9), OR UNDERSCORES.

PYTHON IS CASE-SENSITIVE, SO `DataScience` AND `datascience` ARE DIFFERENT IDENTIFIERS.

```
# Identifiers (Variable Names)
my_variable = 10  # Valid identifier
_my_private_var = 20  # Underscore for private variables
__strong_private_var = 30  # Double underscore for strong private
```

SYNTAX

PYTHON IDENTIFIERS

HERE ARE NAMING CONVENTIONS FOR PYTHON IDENTIFIERS:

- CLASSES: START WITH UPPERCASE LETTERS.
- PRIVATE IDENTIFIERS: START WITH `_single_underscore`.
- STRONGLY PRIVATE: START WITH `__double_underscore`.
- SPECIAL METHODS: START AND END WITH `__double_underscores__` (E.G., `__init__`).

```python
# Class Name (UpperCamelCase as per naming conventions)
class MyClass:
    def __init__(self, value):
        self.value = value  # 'self' is a reference to the instance
```

SYNTAX

RESERVED WORDS

PYTHON HAS A SET OF KEYWORDS THAT CANNOT BE USED AS IDENTIFIERS.

EXAMPLES: IF, ELSE, WHILE, CLASS, DEF, IMPORT.

```python
# Reserved Words (example usage)
def example_function():
    for i in range(3):  # Loop with indentation
        print(f"Iteration {i}")  # Formatted string (f-string)
```

SYNTAX

LINES AND INDENTATION

PYTHON DOES NOT USE BRACES {} TO DEFINE CODE BLOCKS. INSTEAD, INDENTATION IS REQUIRED AND ENFORCED.

```python
# Indentation & Code Blocks
def check_number(num):
    if num > 0:
        print("Positive number")
    elif num == 0:
        print("Zero")
    else:
        print("Negative number")
```

SYNTAX

MULTI-LINE STATEMENTS

PYTHON ALLOWS MULTI-LINE STATEMENTS USING:

BACKSLASH (\) TO EXPLICITLY CONTINUE A LINE.

BRACKETS ([], {}, ()) TO IMPLICITLY CONTINUE.

```
# Multi-line statement using backslash
total = 1 + 2 + 3 + \
        4 + 5 + 6

# Multi-line statement inside brackets (No backslash needed)
numbers = [
    1, 2, 3,
    4, 5, 6
]
```

SYNTAX

QUOTATION IN PYTHON

PYTHON SUPPORTS:

SINGLE QUOTES (') , DOUBLE QUOTES (") , AND TRIPLE QUOTES (''' OR """) FOR MULTI-LINE STRINGS.

```
# Different types of quotations
single_quote = 'This is a single-quoted string'
double_quote = "This is a double-quoted string"
triple_quote = """This is a multi-line string using triple quotes"""
```

SYNTAX

COMMENTS IN PYTHON

- A COMMENT STARTS WITH # AND EXTENDS TO THE END OF THE LINE.
- MULTI-LINE COMMENTS USE TRIPLE QUOTES (''' OR """).

```
# Single-line comment: This is a Python example covering multiple syntax concepts

"""
Multi-line comment:
This script demonstrates identifiers, indentation, multi-line statements,
quotes, comments, reserved words, and suites in Python.
"""
```

SYNTAX

MULTIPLE STATEMENT GROUPS AS SUITES

A SUITE IS A GROUP OF STATEMENTS FORMING A CODE BLOCK. COMPOUND OR COMPLEX STATEMENTS, SUCH AS IF, DEF REQUIRE A HEADER LINE AND A SUITE.

HEADER LINES BEGIN THE STATEMENT (WITH THE KEYWORD) AND TERMINATE WITH A COLON (:) AND ARE FOLLOWED BY ONE OR MORE LINES WHICH MAKE UP THE SUITE.

```
if condition:
    statement1
    statement2
```

```
# Using a suite (code block)
if __name__ == "__main__":
    example_function()  # Calling the function
    obj = MyClass(42)  # Creating an instance
    check_number(obj.value)  # Calling a function
    print(numbers)  # Printing list
```

SOME PYTHON LIBRARIES

- PANDAS – ESSENTIAL FOR DATA MANIPULATION AND ANALYSIS.
- NUMPY – PROVIDES ADVANCED MATHEMATICAL FUNCTIONS AND ARRAY OPERATIONS.
- SCIPY – BUILT ON NUMPY, INCLUDES TOOLS FOR SCIENTIFIC COMPUTING.
- MATPLOTLIB & SEABORN – USED FOR DATA VISUALIZATION AND STATISTICAL GRAPHICS.
- FASTAPI – A MODERN WEB FRAMEWORK OPTIMIZED FOR PERFORMANCE.
- TENSORFLOW & PYTORCH – USED FOR MACHINE LEARNING AND DEEP LEARNING.

```python
# Importing libraries
import numpy as np  # Import NumPy with alias
import pandas as pd  # Import Pandas with alias
```

PIP

`pip` IS PYTHON'S PACKAGE MANAGER USED TO INSTALL AND MANAGE LIBRARIES.

PYTHON 3.4+ INCLUDES PIP BY DEFAULT.

`pip install package_name` INSTALLS PACKAGES FROM PYPI (PYTHON PACKAGE INDEX).

```
haydeml@haydes-personal ~ % pip3 install pandas
```