

Week 3: Complex Data Types

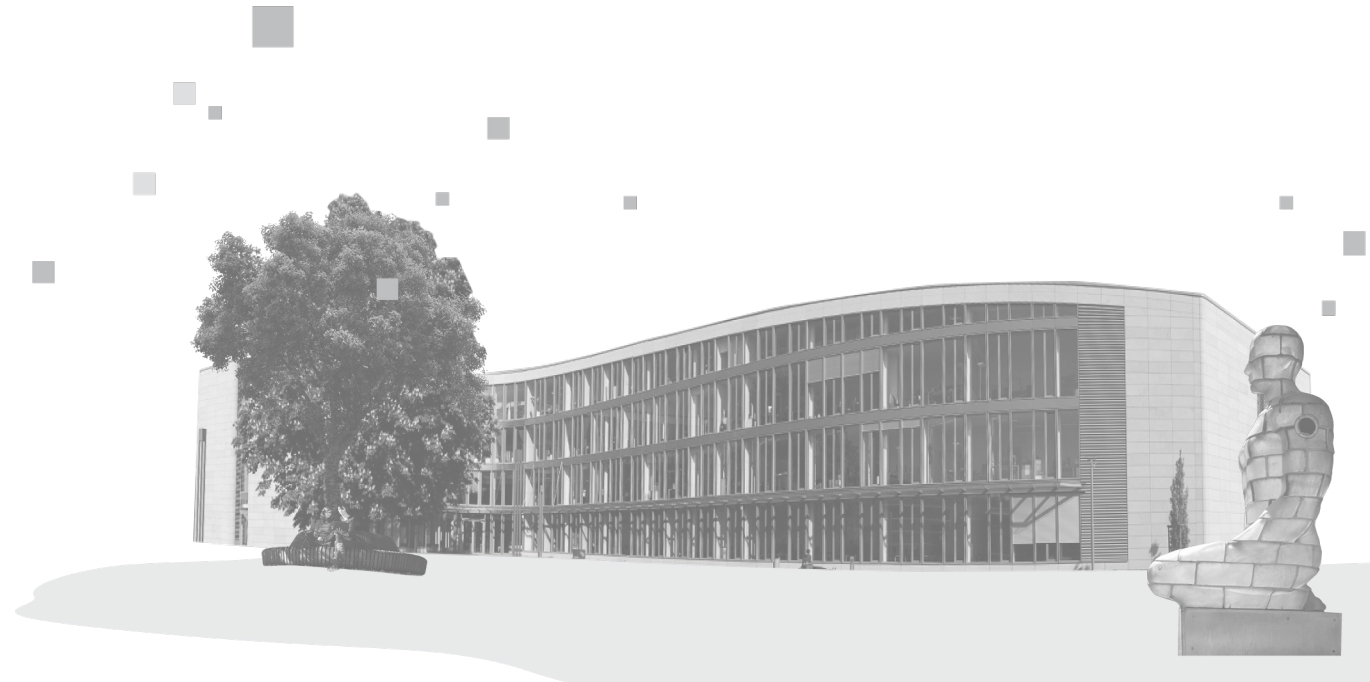
Unit 4: Safe Dictionary Access

Christian Drumm

Stephan Jacobs

**Design IT.
Create Knowledge.**

www.hpi.de



Safe dictionary access

Ensuring safe access to a dictionary

- Accessing a dictionary with a nonexistent key results in a *KeyError*
 - Program is interrupted/aborted
 - Similar to accessing a list with a nonexistent index
- Using the keyword `in` it is possible to prevent this error.
 - `x in dict` returns `True` if the key `x` exists in the dictionary `dict`, `False` if it does not exist.
- Alternatively: Use the dictionary-method `dict.get()`
 - It returns either the assigned value or an error message, which can be defined by the programmer.

```
dict = {1234: "Peter", 2345: "Jane"}  
print(dict[1234])  
print(dict[3456])
```

Peter

```
-----  
KeyError                                Traceback (most recent call last)  
~\AppData\Local\Temp\ipykernel_5544\557728999.py in <module>  
      1 dict = {1234: "Peter", 2345: "Jane"}  
      2 print(dict[1234])  
----> 3 print(dict[3456])  
  
KeyError: 3456
```

Safe dictionary access

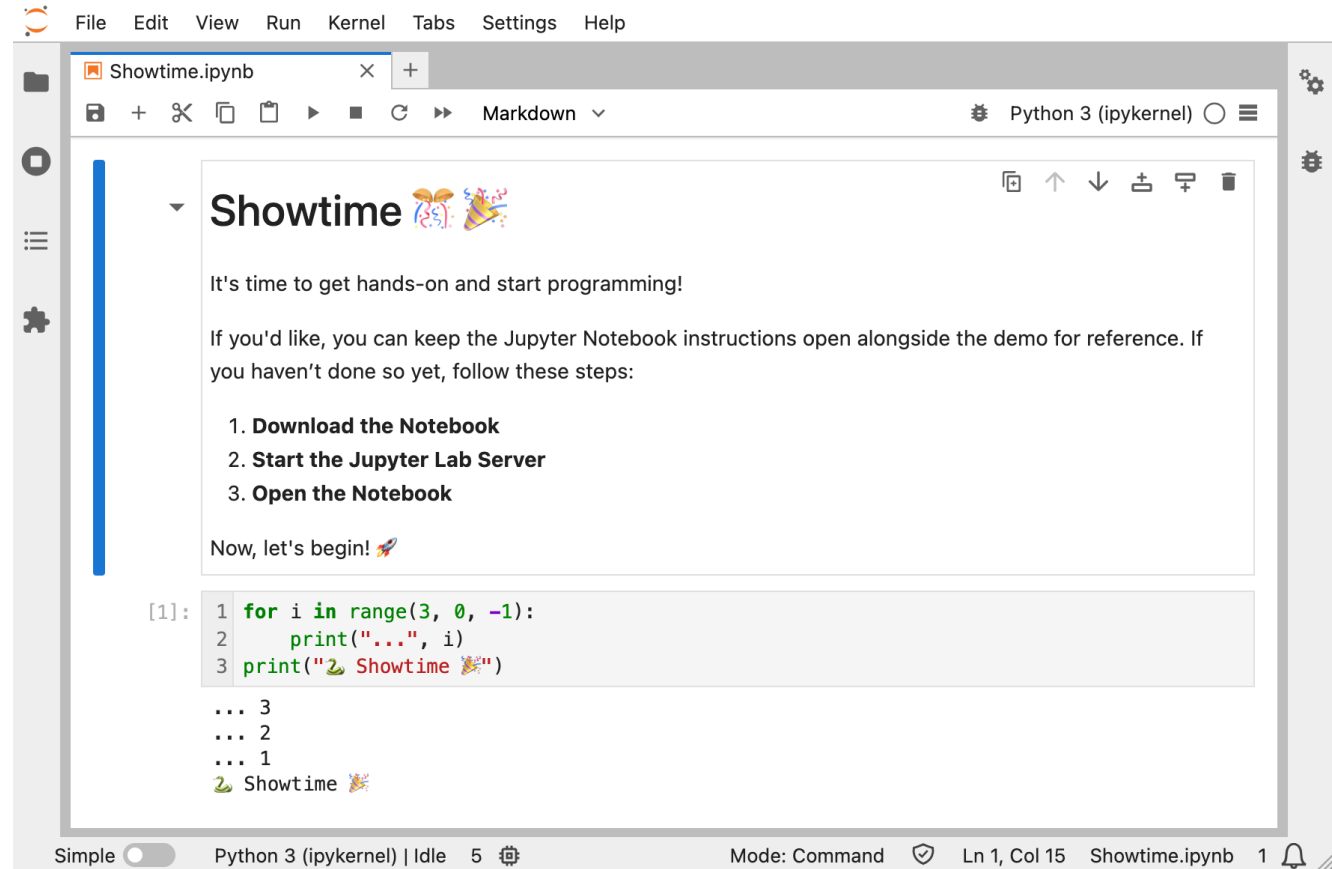
Showtime

Now it's time to get hands on and start programming!

If you like, you can open the [Jupyter Notebook](#) instructions in parallel to the demo.

If you haven't done so yet:

- [Download the Notebook](#)
- [Start the Jupyter Server](#) and open the Notebook



The screenshot shows a Jupyter Notebook window titled 'Showtime.ipynb'. The interface includes a top menu bar (File, Edit, View, Run, Kernel, Tabs, Settings, Help) and a toolbar with icons for file operations and execution. The notebook content is divided into two main sections. The first section, titled 'Showtime' with a party popper emoji, contains text instructions: 'It's time to get hands-on and start programming!', 'If you'd like, you can keep the Jupyter Notebook instructions open alongside the demo for reference. If you haven't done so yet, follow these steps:', a numbered list (1. Download the Notebook, 2. Start the Jupyter Lab Server, 3. Open the Notebook), and 'Now, let's begin!' with a rocket emoji. The second section is a code cell labeled '[1]:' containing a Python loop that prints '... 3', '... 2', '... 1', and '🎉 Showtime 🎉'. The output of the code cell is visible below the code. The bottom status bar shows 'Simple' mode, 'Python 3 (ipykernel) | Idle', '5' lines, 'Mode: Command', 'Ln 1, Col 15', and the filename 'Showtime.ipynb'.

```
File Edit View Run Kernel Tabs Settings Help

Showtime.ipynb
Python 3 (ipykernel)

Showtime 🎉

It's time to get hands-on and start programming!

If you'd like, you can keep the Jupyter Notebook instructions open alongside the demo for reference. If you haven't done so yet, follow these steps:

1. Download the Notebook
2. Start the Jupyter Lab Server
3. Open the Notebook

Now, let's begin! 🚀

[1]: 1 for i in range(3, 0, -1):
      2     print("...", i)
      3 print("🎉 Showtime 🎉")

      ... 3
      ... 2
      ... 1
      🎉 Showtime 🎉

Simple Python 3 (ipykernel) | Idle 5 Mode: Command Ln 1, Col 15 Showtime.ipynb 1
```

Safe dictionary access

Summary / key takeaways

In this unit you learned ...

- ... how to prevent access errors in dictionaries
- ... how to get the value for specific keys securely without interrupting the main program

