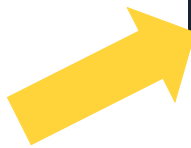# Exceptions

What are Exceptions?

```
user_input = input("Enter a number: ")
number = int(user_input)
print(f"Your number is {number}.")
```

```
In [5]: runfile('/home/jonas/untitled0.py', wdir='/home/jonas')

Enter a number: 3
Your number is 3.

In [6]: runfile('/home/jonas/untitled0.py', wdir='/home/jonas')

Enter a number: string
Traceback (most recent call last):

  File "/home/jonas/untitled0.py", line 11, in <module>
    number = int(user_input)

ValueError: invalid literal for int() with base 10: 'string'
```

Exceptions will end your program!

Why should I care?

```
In [26]: existing_file = Path("/home/jonas/Dokumente/pythonBootcamp2021/code/.gitignore")
    ...: existing_content = existing_file.read_text()
    ...: print(f"{existing_content[:30]} ...")
```

```
In [27]: not_existing_file = Path("/home/jonasDokumente/pythonBootcamp2021/code/
defenitlyNotAnExistingFile")
    ...: not_existing_content = not_existing_file.read_text()
    ...: print(not_existing_content)
```

Vocabular

Python Raises Exceptions

The rest of the world Throws Errors

Raising Exceptions

```
In [5]: print("<<< Start >>>")
   ...: raise ValueError("Some helpful message.")
   ...:
   ...: print("<<< End >>>")
<<< Start >>>
Traceback (most recent call last):

  File "<ipython-input-5-b7c926507724>", line 3, in <module>
    raise ValueError("Some helpful message.")

ValueError: Some helpful message.
```

Built-in Exceptions	
NameError	NotImplementedError
AttributeError	SystemExit
ImportError	TypeError
ModuleNotFoundError	ValueError
IndexError	ZeroDivisionError
KeyError	FileNotFoundError
KeyboardInterrupt	FileExistsError

Base Exceptions:

Exception
ArithmeticError
LookupError
BufferError

https://docs.python.org/3/library/exceptions.html

Catching Exceptions
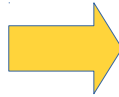
```
In [4]:
   ...:
   ...: try:
   ...:     file = Path("SomeFileThatDoesNotExist")
   ...:     content = file.read_text()
   ...:
   ...: except NameError as e:
   ...:     print(e)
   ...:     content = "Default text"
   ...:
   ...: except FileNotFoundError as e:
   ...:     print(f"Error while reading file! {e}")
   ...:     content = "Default text"
   ...:
   ...: print(content)
name 'Path' is not defined
Default text
```

```
In [5]: from pathlib import Path
   ...:
   ...: try:
   ...:     file = Path("SomeFileThatDoesNotExist")
   ...:     content = file.read_text()
   ...:
   ...: except NameError as e:
   ...:     print(e)
   ...:     content = "Default text"
   ...:
   ...: except FileNotFoundError as e:
   ...:     print(f"Error while reading file! {e}")
   ...:     content = "Default text"
   ...:
   ...: print(content)
Error while reading file! [Errno 2] No such file or directory: 'SomeFileThatDoesNotExist'
Default text
```

Handling Exceptions

```python
      8
      9    # Handling files and paths
     10    from pathlib import Path
     11
     12    #
     13    #   READ FILE WITH TRY-CATCH
     14    #
     15    # Ask user for a file to read. Ask again if file does not exist
     16    while True:
     17        try:
     18            # Ask for file path
     19            user_filename = input("Enter file name: ")
     20
     21            # Try to read file without checking if it exists
     22            file = Path(user_filename)
     23            print(f"Reading {str(file)} ...")
     24            content = file.read_text()
     25
     26            # Leave the while loop
     27            break
     28
     29        except FileNotFoundError:
     30            # Catch Exception: File was not found. Ask user again.
     31            print("File not found! Try again.")
     32
     33    # Continue with the rest of the program
     34    print(content)
     35
     36    #
     37    #   READ FILE WITHOUT TRY-CATCH
     38    #
     39    # Ask user for a file to read.
     40    user_file = input("Enter file name: ")
     41    user_file = Path(user_file)
     42    # Ask the user again, if the file does not exist
     43    while user_file.is_file() == False:
     44        user_file = input("File not found. Enter new file name: ")
     45        user_file = Path(user_file)
     46
     47    # Read text
     48    content = user_file.read_text()
     49    # Continue with the rest of the program
     50    print(content)
```

```
Enter file name: a
Reading a ...
File not found! Try again.

Enter file name: README.md
Reading README.md ...
# pythonBootCamp2021
Examples and excercises for my talk at the python boot camp 2021


Enter file name:a

File not found. Enter new file name: README.md
# pythonBootCamp2021
Examples and excercises for my talk at the python boot camp 2021
```

Exercise 9: New Hello, You!

Rewrite the "Hello, You!" program.
Ask the user for its name and age.
Make sure that the age is an integer.
Print a personalised greeting.