Exceptions

Explaining and giving examples how to handle exceptions in the Python programming language.

- Exceptions: What are they?
- Try statement defined
- Try statement
- Try Statement Syntax
- How to handle a ValueError exception
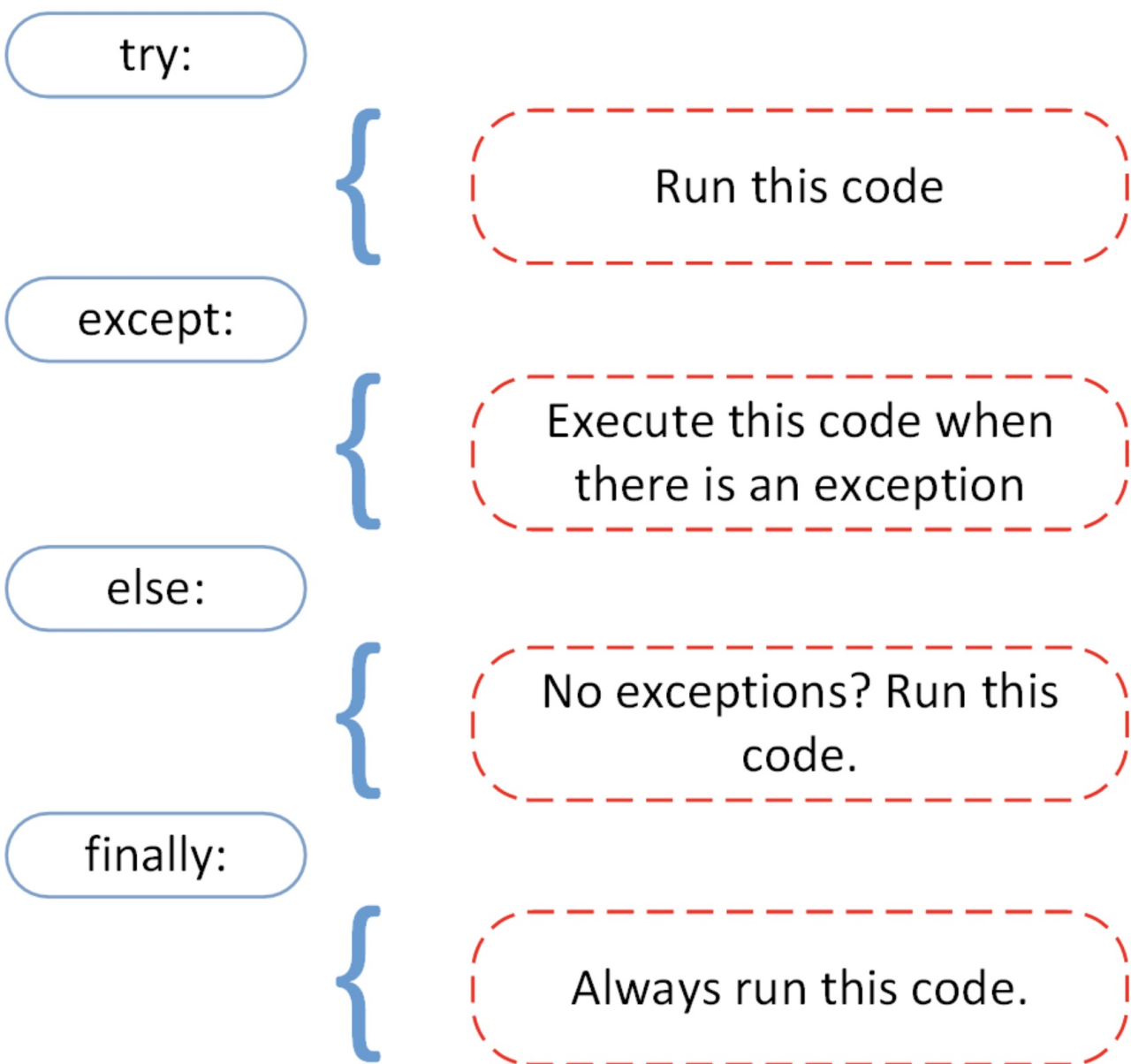- How to handle ALL exceptions
- Resources

Exceptions: What are they?

- When an error happens in our code, it is called an exception.
- Python will normally stop and generate (throw) an error message to the console.
- 'Try' statements are used to handle exceptions.
- This makes it possible to code for most common user input errors.

Try Statement Defined

- In a try statement, you code any statements that may throw an exception in the try clause. Then, you can code an except clause that handles any exception that occurs.
- If you don't code the name of an exception type in the except clause, the except clause handles all types of exceptions that can occur.
- When an exception occurs, Python skips any remainng statements in the try clause and executes the statements in the except clause
- This is known as exception handling

Try Statement

Try statement syntax

This try statement catches an exception.

```
In [ ]:   try:
              statements
          except [ExceptionName]:
              statements
```

How to handle a ValueError exception

In this example we enter the value 55.

```
In [1]:   try:
              number = int(input("Enter a number between 1 and 100: "))
              print("You entered a valid number of " + str(number) + ".")
          except ValueError:
              print("You entered an invalid integer. Please try again.")
          print("Thanks")

          Enter a number between 1 and 100: 55
          You entered a valid number of 55.
          Thanks
```

In this example we enter the value Five. Notice how the print function in the except ValueError is executed.

```
In [2]:   try:
              number = int(input("Enter a number between 1 and 100: "))
              print("You entered a valid number of " + str(number) + ".")
          except ValueError:
              print("You entered an invalid integer. Please try again.")
          print("Thanks")

          Enter a number between 1 and 100: Five
          You entered an invalid integer. Please try again.
          Thanks
```

How to handle ALL exceptions

Example of NO exception being thrown

```
In [4]:   try:
              numMiles = int(input("Enter miles driven: "))
              numGallons = int(input("Enter amount of gas used: "))
              print("You traveled " + str(numMiles) + " and used " + str(numGallons) + " gallons of gas.")
          except :
              print("You entered an invalid integer. Please try again.")
          print("Thanks")

          Enter miles driven: 150
          Enter amount of gas used: 15
          You traveled 150 and used 15 gallons of gas.
          Thanks
```

Example of an exception being thrown!

```
In [6]:   try:
              numMiles = int(input("Enter miles driven: "))
              numGallons = int(input("Enter amount of gas used: "))
              print("You traveled " + str(numMiles) + " and used " + str(numGallons) + " gallons of gas.")
          except :
              print("You entered an invalid integer. Please try again.")
          print("Thanks")

          Enter miles driven: 150
          Enter amount of gas used: ten
          You entered an invalid integer. Please try again.
          Thanks
```

Resources

Sources and deeper learning research:

1. Python Programming-Beginner to Pro.Michael Urban,Joel Murach.Murach Press.2016.ISBN:978-1-890774-97-4.www.murach.com.

1. W3Schools.com - Python Try Except