

این قسمت:

پیاده سازی بلوک های تودرتو
مدیریت خطا، فراخوانی دستی
استثناء ها، به کارگیری بلوک
finally

آموزش رایگان زبان برنامه نویسی پایتون

except `__del__` and `__getstate__` methods, which are called when an object is deleted or pickled, respectively. The `__del__` method is called when the object is garbage collected, and the `__getstate__` method is called when the object is pickled. The `__del__` method is called when the object is garbage collected, and the `__getstate__` method is called when the object is pickled.

[illegible]

□□□□□□ □□□□ □□□□□□ □□□□□□□□

[illegible]

.Python IDLE.1 New File File Python IDLE.1

.Python IDLE.2

```
TryAgain= True
```

```
while TryAgain:
```

```
    try:
```

```
        Value=int(input("Type a whole number. "))
```

```
    except ValueError:
```

```
        print("You must type a whole number!")
```

```
    try:
```

```
        DoOver=input("Try again (y/n)? ")
```

```
    except:
```

```
        print("OK see you next time!")
```

```
        TryAgain=False
```

```
    else:
```

```
        if (str.upper(DoOver) == "N"):
```

```
            TryAgain=False
```

```
except KeyboardInterrupt:
```

```
    print("You pressed Ctrl+C")
```

```
    print("see you next time!")
```

```
    TryAgain=False
```

```
else:
```

```
    print(Value)
```

```
    TryAgain=False
```

```
nested.py - C:/Node/nested.py (3.7.2)
File Edit Format Run Options Window Help
TryAgain = True
while TryAgain:
    try:
        Value = int(input("Type a whole number. "))
    except ValueError:
        print("You must type a whole number!")
        try:
            DoOver = input("Try again (y/n)? ")
        except:
            print("OK, see you next time!")
            TryAgain = False
        else:
            if (str.upper(DoOver) == "N"):
                TryAgain = False
    except KeyboardInterrupt:
        print("You pressed Ctrl+C!")
        print("See you next time!")
        TryAgain = False
    else:
        print(Value)
        TryAgain = False
```

이 코드는 사용자가 정수 값을 입력할 때까지 계속 실행됩니다. (이 코드는 Python 3.7.2 버전에서 실행되었습니다.) 이 코드는 try-except 문을 사용하여 ValueError 예외를 처리합니다. try 블록에서는 사용자가 입력한 값을 정수로 변환하려고 시도합니다. 만약 ValueError 예외가 발생하면, except 블록에서는 사용자에게 메시지를 출력하고 다시 시도할지 여부를 묻습니다. 사용자가 'N'을 입력하면 TryAgain 변수를 False로 설정하여 while 루프를 종료시킵니다. 또한 KeyboardInterrupt 예외를 처리하여 Ctrl+C 키를 누르면 프로그램을 종료할 수 있도록 합니다. try-except 문을 사용하여 KeyboardInterrupt 예외를 처리합니다. try 블록에서는 사용자가 입력한 값을 정수로 변환하려고 시도합니다. 만약 ValueError 예외가 발생하면, except 블록에서는 사용자에게 메시지를 출력하고 다시 시도할지 여부를 묻습니다. 사용자가 'N'을 입력하면 TryAgain 변수를 False로 설정하여 while 루프를 종료시킵니다. 또한 KeyboardInterrupt 예외를 처리하여 Ctrl+C 키를 누르면 프로그램을 종료할 수 있도록 합니다. try-except 문을 사용하여 KeyboardInterrupt 예외를 처리합니다.

3. 이 코드를 실행하려면 Run Module 버튼을 클릭하거나 run 명령을 실행합니다. 이 코드는 Python 3.7.2 버전에서 실행되었습니다.

4. Hello 이 코드는 Python 3.7.2 버전에서 실행되었습니다. 이 코드는 try-except 문을 사용하여 ValueError 예외를 처리합니다. try 블록에서는 사용자가 입력한 값을 정수로 변환하려고 시도합니다. 만약 ValueError 예외가 발생하면, except 블록에서는 사용자에게 메시지를 출력하고 다시 시도할지 여부를 묻습니다. 사용자가 'N'을 입력하면 TryAgain 변수를 False로 설정하여 while 루프를 종료시킵니다. 또한 KeyboardInterrupt 예외를 처리하여 Ctrl+C 키를 누르면 프로그램을 종료할 수 있도록 합니다. try-except 문을 사용하여 KeyboardInterrupt 예외를 처리합니다.

2. Python 3.3.4 Shell

try:

raise ValueError

except ValueError:

print("ValueError Exception!")

Python 3.3.4 Shell
File Edit Shell Debug Options Windows Help
Python 3.3.4 (v3.3.4:7ff62415e426, Feb 10 2014, 18:13:51) [MSC v.1600 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
ValueError Exception!
>>> |
Ln: 6 Col: 4

Python 3.3.4 Shell

Python 3.3.4 (v3.3.4:7ff62415e426, Feb 10 2014, 18:13:51) [MSC v.1600 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
ValueError Exception!
>>> |
Ln: 6 Col: 4

Python 3.3.4 Shell

try:

Ex=ValueError()

Ex.strerror= "Value must be within 1 and 10."

raise Ex

2. Python 3.3.4 Shell

