Curso Python Avançado

Exceções

olist

Exceções

Exceções

Como já vimos durante exercícios, exceções podem ser emitidas durante a execução de um programa.

```
>>> 10 * (1/0)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ZeroDivisionError: division by zero
>>> 4 + spam*3
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'spam' is not defined
>>> '2' + 2
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: Can't convert 'int' object to str implicitl
```

Exceções

Exceções podem ser de diversos tipos, todas as exeções *builtin* herdam do tipo `Exception`.

Tratando Exceções

Tratando Exceções

Exceções são tratadas basicamente utilizando `try` e `except`.

```
>>> while True:
...     try:
...         x = int(input("Please enter a number: "))
...         break
...     except ValueError:
...         print("Oops!  That was no valid number.  Try again...")
...
```

Tratando Exceções

É possível ter múltiplos `except` para cada `try`.

Um `except` pode tratar múltiplos tipos de exceção, ou nenhum tipo (para este caso sempre deve ser definida ao fim da lista, deve ser evitado).

```
>>> while True:
...     try:
...         x = int(input("Please enter a number: "))
...         break
...     except ValueError:
...         print("Oops!  That was no valid number.  Try again...")
...   except (RuntimeError, TypeError, NameError):
...       print("Undesired but expected error happened.")
...   except:
...       print("Some other error happened")
```

Tratando Exceções

A ordem de definição das exeções também é importante quando múltiplas exceções são tratadas e estas possuem relação de herança entre sí.

```
class B(Exception):
    pass

class C(B):
    pass

class D(C):
    pass

for cls in [B, C, D]:
    try:
        raise cls()
    except D:
        print("D")
    except C:
        print("C")
    except B:
        print("B")
```

Exercício

Exercício

Iremos complementar o exercício da aula anterior.

Criar uma interface com menu interativo para realizar operações sobre um correntista.

Erros de entrada e operações devem utilizar exeções e serem tratados.

Priorizar separação da lógica de interface da lógica de operações e correntista.

Sempre deve ser visível:
- Dados do Correntista
- Saldo atual

As opções de operação devem ser:
- Fazer Depósito
- Fazer Saque
- Ver Histórico
- Sair

Extra (opcional):
- Cadastrar Correntista
- Selecionar Correntista

Outros menus devem ficar disponíveis somente após um correntista for selecionado.

```
Correntista: Léo Willian Kölln
Saldo: R$17,50

Operações
=========
1 - Fazer Depósito
2 - Fazer Saque
3 - Ver Histórico
4 - Sair

Selecione uma operação: ___
```