Python - Aula

Tratamento de exceções

Python - Tratamento de exceções

1) TIPOS DE ERRO

Erros em Python ocorrem quando uma determinada declaração não está de acordo com os padrões definidos da linguagem.

Python - Tratamento de exceções

1) TIPOS DE ERRO

Erros em Python ocorrem quando a execução do código descrito não está de acordo com os padrões definidos da linguagem.

Erros podem ser divididos em erros de **Sintaxe** e de **Execução**.

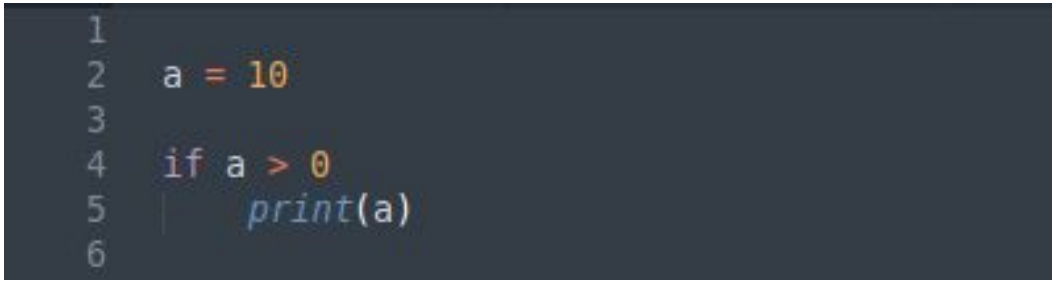
Python - Tratamento de exceções

2) SYNTAX ERRORS

Syntax Errors ocorrem quando a estrutura correta da linguagem (**sintaxe**) não é utilizada. Podem ser chamados também de **Parsing Error**.

Python - Tratamento de exceções

2) SYNTAX ERRORS - Exemplo



```
1
2  a = 10
3
4  if a > 0
5      print(a)
6
```

Python - Tratamento de exceções

2) SYNTAX ERRORS - Exemplo

```
1
2  a = 10
3
4  if a > 0
5      print(a)
6
```

```
  File "/home/alexandre/Documentos/Projetos/f/teste.py", line 4
    if a > 0
           ^
SyntaxError: invalid syntax
>>>
```

Python - Tratamento de exceções

3) EXCEÇÕES

Exceções são erros ou desvios do caminho esperado pelo programa que ocorrem em **tempo de execução**.

Python - Tratamento de exceções

3) EXCEÇÕES

Exceções são erros ou desvios do caminho esperado pelo programa que ocorrem em **tempo de execução**.

Exceções em Python, diferente de outras linguagens, não servem somente para tratamento de erros, podendo ser utilizados para alteração do fluxo do programa.

Python - Tratamento de exceções

3) EXCEÇÕES - Exemplo

```
1
2  texto = "Vovó foi a feira"
3  numero = int(texto)
4
```

Python - Tratamento de exceções

3) EXCEÇÕES - Exemplo

```
1
2  texto = "Vovó foi a feira"
3  numero = int(texto)
4
```

```
Traceback (most recent call last):
  File "/home/alexandre/Documentos/Projetos/f/teste.py", line 3, in <module>
    numero = int(texto)
ValueError: invalid literal for int() with base 10: 'Vovó foi a feira'
>>>
```

Python - Tratamento de exceções

3) EXCEÇÕES - Exemplo

```
1
2   arquivo = open("teste.txt", "r")
3   texto = arquivo.readlines()
4
```

Python - Tratamento de exceções

3) EXCEÇÕES - Exemplo

```
1
2   arquivo = open("teste.txt", "r")
3   texto = arquivo.readlines()
4
```

```
Traceback (most recent call last):
  File "/home/alexandre/Documentos/Projetos/f/teste.py", line 2, in <module>
    arquivo = open("teste.txt", "r")
FileNotFoundError: [Errno 2] No such file or directory: 'teste.txt'
>>>
```

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except

O bloco **TRY** executa o caminho esperado definido para o sistema.

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except

O bloco **TRY** executa o caminho esperado definido para o sistema.

O bloco **EXCEPT** executa o caminho de exceção para determinado comando.

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except

O bloco **TRY** executa o caminho esperado definido para o sistema.

O bloco **EXCEPT** executa o caminho de exceção para determinado comando.

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except

O bloco **TRY** executa o caminho esperado definido para o sistema.

O bloco **EXCEPT** executa o caminho de exceção para determinado comando.

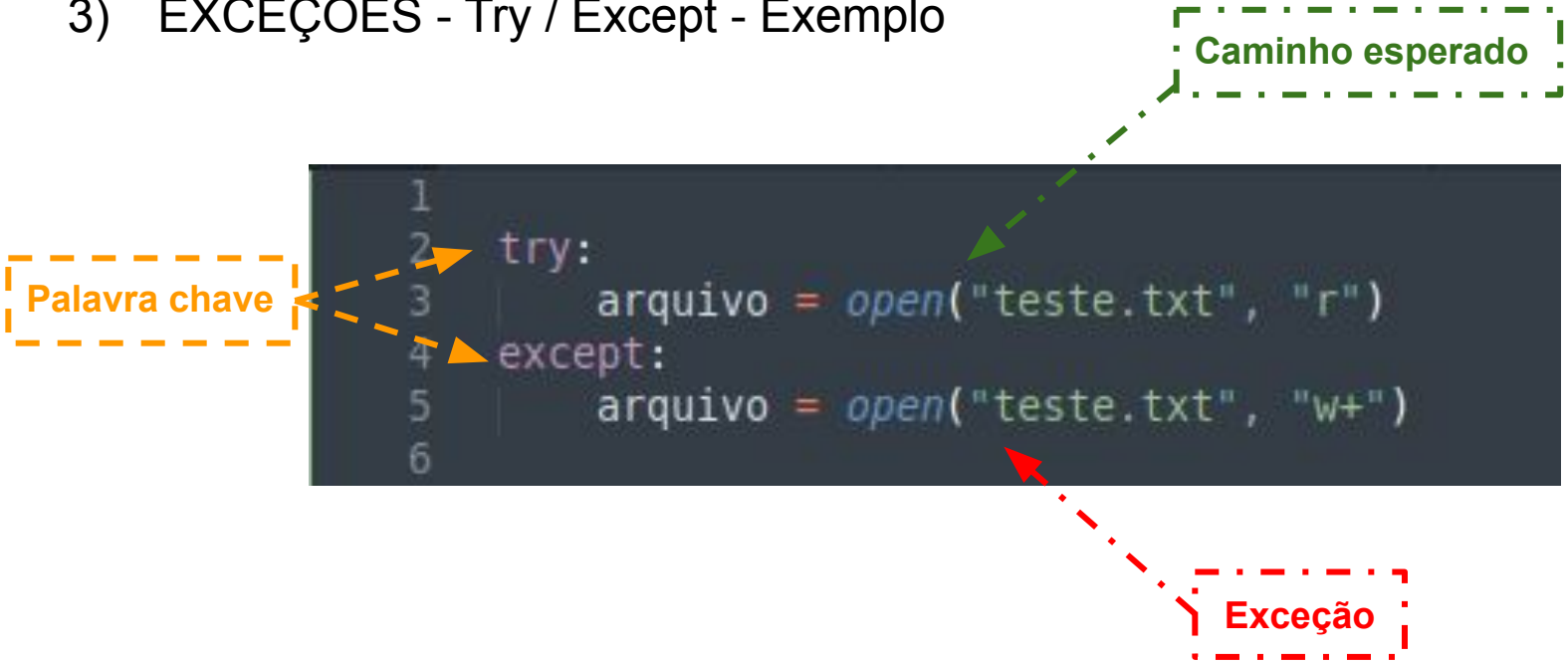
Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except - Exemplo

```
1
2  try:
3      arquivo = open("teste.txt", "r")
4  except:
5      arquivo = open("teste.txt", "w+")
6
```

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except - Exemplo



Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except - Exemplo

```
1
2  a = input()
3
4  try:
5      divisao = 1 / int(a)
6  except ValueError:
7      print("Valor não pode ser convertido para inteiro")
8  except ZeroDivisionError:
9      print("O valor não pode ser zero")
10
```

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except / Finally

O bloco **FINALLY** é executado independente de o código ter gerado ou não uma exceção.

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except / Finally - Exemplo

```
1
2  try:
3      arquivo = open("teste.txt", "r")
4  except:
5      arquivo = open("teste.txt", "w+")
6  finally:
7      arquivo.close()
8
```

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except / Finally - Exemplo

```
1
2  try:
3      arquivo = open("teste.txt", "r")
4  except:
5      arquivo = open("teste.txt", "w+")
6  finally:
7      arquivo.close()
8
```

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except / Else / Finally

O bloco **ELSE** executa apenas quando a instrução do bloco **try** **NÃO** gera exceções.

Python - Tratamento de exceções

3) EXCEÇÕES - Try / Except / Else / Finally - Exemplo

```
1
2  try:
3      arquivo = open("teste.txt", "r")
4  except:
5      arquivo = open("teste.txt", "w+")
6  else:
7      texto = arquivo.readlines()
8  finally:
9      arquivo.close()
10
```

Python - Tratamento de exceções

#EXERCÍCIOS

1) Utilizando tratamento de exceções, crie um programa que, dado um valor inteiro informado pelo usuário, retorne a divisão de 1 por este. Se o valor informado for zero, o programa de informar “Infinito” como resultado. Caso o valor informado não seja um número, o programa deve informar o usuário e continuar solicitando valores até que este seja.