Fatec
Rio Claro

Linguagem de Programação I

Prof. Orlando Saraiva Júnior
orlando.nascimento@fatec.sp.gov.br

"Programming isn't about what you know; it's about what you can figure out."

Chris Pine

"Programação não é sobre o que você sabe; é sobre o que você consegue descobrir."

Chris Pine

Tratamento de Erro

Tratamento de erros

Fatec
Rio Claro

O Python possui diversas exceções integradas que representam uma ampla gama de erros comuns e situações excepcionais. Essas exceções são organizadas em dois grupos:

- Exceções da classe base
- Exceções concretas

O primeiro grupo de exceções compreende classes de exceção que são usadas principalmente como classes base para outras exceções. Por exemplo, neste grupo, você tem a classe **Exception**, que foi especialmente projetada para permitir a criação de exceções personalizadas.

Tratamento de erros

Fatec
Rio Claro

O segundo grupo contém exceções que você normalmente verá em código Python ou obterá ao executar código Python.

```
import builtins
for x in dir(builtins):
    if x.endswith('Error'):
        print(x)
```

Tratamento de erros

Fatec
Rio Claro

Todas as exceções internas que não existem no sistema são derivadas da classe **Exception**. Todas as exceções definidas pelo usuário também devem ser derivadas desta classe.

```
class FatecError(Exception):
    pass

raise FatecError('Erro interno na aplicação')
```

Comunicação InterProcessos

Programa

Fatec
Rio Claro

Um programa é um conjunto de instruções escritas em uma linguagem de programação, armazenadas em um arquivo de forma passiva, como um .py (Python), .java (Java) ou .exe (Windows executável).

Ele representa um plano ou roteiro do que deve ser feito, mas por si só, não realiza nenhuma ação enquanto não for executado.

Pode-se dizer que um programa é uma entidade estática, que descreve logicamente um comportamento esperado, mas não tem vida até que seja colocado em execução pelo sistema operacional.

Processo

Fatec
Rio Claro

Já um **processo** é a instância ativa de um **programa em execução**. Quando um programa é iniciado, o sistema operacional cria um processo que recebe recursos como memória, identificadores (PID), tempo de CPU e canais de entrada/saída.

Esse processo possui um estado próprio, incluindo seu contador de programa (a próxima instrução a executar), pilha de execução e variáveis.

Processo

Fatec
Rio Claro

Assim, vários processos podem ser criados a partir do mesmo programa - cada um com seus próprios dados e execução independente. O conceito de processo é essencial para o multitarefa, permitindo que o computador execute diversos programas simultaneamente.

Comunicação InterProcessos

Fatec
Rio Claro

IPC (*Inter-Process Communication*) é um conjunto de mecanismos que permite que diferentes processos — que normalmente operam de forma isolada — compartilhem dados e coordenem suas ações.

Isso é fundamental em sistemas operacionais modernos, onde múltiplos processos podem precisar colaborar para realizar tarefas complexas.

Entre os métodos mais comuns de IPC estão: ***pipes***, ***sockets***, ***memória compartilhada***, ***semáforos***, ***queues*** e ***signals***.

Fatec
Rio Claro

Na prática, a escolha do método de IPC depende do contexto: por exemplo, processos que rodam em máquinas diferentes geralmente se comunicam via sockets, enquanto processos locais podem usar memória compartilhada ou pipes para maior desempenho.

Em linguagens como Python, Java, C++ e Ruby, há suporte nativo ou via bibliotecas para implementar esses mecanismos, tornando o IPC uma técnica poderosa para criar sistemas paralelos, distribuídos e responsivos.

Comunicação InterProcessos

Fatec
Rio Claro

O uso correto do IPC é essencial para garantir que os dados trocados entre processos sejam consistentes, sincronizados e seguros.

Em Python, a comunicação entre processos (Inter-Process Communication – IPC) pode ser realizada de diversas formas, dependendo do contexto (local, remoto, necessidade de paralelismo, performance, etc).

Socket

Socket

Fatec
Rio Claro

Sockets são interfaces que permitem a comunicação entre processos - locais ou remotos - por meio de uma rede (como a Internet ou uma rede local). Eles funcionam como pontos de extremidade para enviar e receber dados entre dois dispositivos.

Existem dois principais tipos de sockets: os sockets ***TCP*** (orientados à conexão), que garantem entrega ordenada e confiável dos dados, e os sockets ***UDP*** (sem conexão), que são mais rápidos, porém não garantem a entrega nem a ordem das mensagens.

Socket

Fatec
Rio Claro

Em Python, o módulo padrão socket fornece uma interface de baixo nível para a criação e manipulação de sockets, permitindo que o desenvolvedor implemente clientes e servidores com facilidade.

Um servidor Python com socket TCP, por exemplo, escuta em uma porta, aceita conexões de clientes e pode receber e enviar dados em tempo real.

Já o cliente se conecta ao endereço e porta do servidor e envia mensagens por meio de métodos simples como send() e recv().

XML-RPC
Remote Procedure Call using XML

XML-RPC

Fatec
Rio Claro

XML-RPC (Remote Procedure Call using XML) é um protocolo simples que permite a comunicação entre programas, mesmo que estejam em linguagens ou sistemas operacionais diferentes, por meio da chamada de funções remotas.

A comunicação ocorre via *HTTP*, e os dados são codificados em *XML*, tornando o protocolo legível e baseado em padrões amplamente adotados.

O cliente envia uma requisição que contém o nome da função a ser executada remotamente, junto com seus parâmetros, e o servidor responde com o resultado.

XML-RPC

Fatec
Rio Claro

Essa abordagem facilita a criação de sistemas distribuídos, nos quais um programa pode chamar funções em outro computador como se fossem locais.

No Python, o módulo padrão **xmlrpc.server** permite criar servidores que disponibilizam funções para serem acessadas remotamente, enquanto o módulo **xmlrpc.client** (ou *ServerProxy*) permite que clientes se conectem a esses servidores e façam chamadas de método.

XML-RPC

Fatec
Rio Claro

A principal vantagem é a simplicidade: basta definir funções comuns em Python, registrá-las em um servidor XML-RPC e o cliente poderá acessá-las facilmente.

Apesar de hoje existirem alternativas mais modernas como REST e gRPC, o XML-RPC ainda é útil para sistemas legados ou aplicações onde simplicidade e interoperabilidade são mais importantes do que desempenho ou flexibilidade.

Fatec
Rio Claro

Dúvidas

Prof. Orlando Saraiva Júnior
orlando.nascimento@fatec.sp.gov.br