



Linguagem de Programação Python

90H

CONTEÚDO PROGRAMÁTICO

1. Introdução

2. Utilizando o interpretador Python

- 2.1. Chamando o interpretador
 - 2.1.1. Passagem de argumentos
 - 2.1.2. Modo interativo
- 2.2. O interpretador e seu ambiente
 - 2.2.1. Edição de código-fonte

3. Uma introdução informal ao Python

- 3.1. Usando Python como uma calculadora
 - 3.1.1. Números
 - 3.1.2. Strings
 - 3.1.3. Listas
- 3.2. Primeiros passos para a programação

4. Mais ferramentas de controle de fluxo

- 4.1. Comandos if
- 4.2. Comandos for
- 4.3. A função range()
- 4.4. Comandos break e continue, e cláusula else, nos laços de repetição
- 4.5. Comandos pass
- 4.6. Instruções match
- 4.7. Definindo funções
- 4.8. Mais sobre definição de funções
 - 4.8.1. Argumentos com valor padrão
 - 4.8.2. Argumentos nomeados
 - 4.8.3. Parâmetros especiais
 - 4.8.3.1. Argumentos posicional-ou-nomeados
 - 4.8.3.2. Parâmetros somente-posicionais
 - 4.8.3.3. Argumentos somente-nomeados
 - 4.8.3.4. Exemplos de funções
 - 4.8.3.5. Recapitulando
 - 4.8.4. Listas de argumentos arbitrárias
 - 4.8.5. Desempacotando listas de argumentos
 - 4.8.6. Expressões lambda
 - 4.8.7. Strings de documentação
 - 4.8.8. Anotações de função
- 4.9. Intermezzo: estilo de codificação

5. Estruturas de dados

- 5.1. Mais sobre listas
 - 5.1.1. Usando listas como pilhas
 - 5.1.2. Usando listas como filas
 - 5.1.3. Compreensões de lista
 - 5.1.4. Compreensões de lista aninhadas
- 5.2. A instrução del
- 5.3. Tuplas e Sequências
- 5.4. Conjuntos
- 5.5. Dicionários
- 5.6. Técnicas de iteração
- 5.7. Mais sobre condições
- 5.8. Comparando sequências e outros tipos

6. Módulos

- 6.1. Mais sobre módulos

- 6.1.1. Executando módulos como scripts
- 6.1.2. O caminho de busca dos módulos
- 6.1.3. Arquivos Python “compilados”
- 6.2. Módulos padrões
- 6.3. A função dir()
- 6.4. Pacotes
 - 6.4.1. Importando * de um pacote
 - 6.4.2. Referências em um mesmo pacote
 - 6.4.3. Pacotes em múltiplos diretórios

7. Entrada e Saída

- 7.1. Refinando a formatação de saída
 - 7.1.1. Strings literais formatadas
 - 7.1.2. O método format()
 - 7.1.3. Formatação manual de string
 - 7.1.4. Formatação de strings à moda antiga
- 7.2. Leitura e escrita de arquivos
 - 7.2.1. Métodos de objetos arquivo
 - 7.2.2. Gravando dados estruturados com json

8. Erros e exceções

- 8.1. Erros de sintaxe
- 8.2. Exceções
- 8.3. Tratamento de exceções
- 8.4. Levantando exceções
- 8.5. Encadeamento de exceções
- 8.6. Exceções definidas pelo usuário
- 8.7. Definindo ações de limpeza
- 8.8. Ações de limpeza predefinidas

9. Classes

- 9.1. Uma palavra sobre nomes e objetos
- 9.2. Escopos e espaços de nomes do Python
 - 9.2.1. Exemplo de escopos e espaço de nomes
- 9.3. Uma primeira olhada nas classes
 - 9.3.1. Sintaxe da definição de classe
 - 9.3.2. Objetos de Class
 - 9.3.3. Objetos instância
 - 9.3.4. Objetos método

9.3.5. Variáveis de classe e instância

9.4. Observações aleatórias

9.5. Herança

9.5.1. Herança múltipla

9.6. Variáveis privadas

9.7. Curiosidades e conclusões

9.8. Iteradores

9.9. Geradores

9.10. Expressões geradoras

10. Um breve passeio pela biblioteca padrão

10.1. Interface com o sistema operacional

10.2. Caracteres curinga

10.3. Argumentos de linha de comando

10.4. Redirecionamento de erros e encerramento do programa

10.5. Reconhecimento de padrões em strings

10.6. Matemática

10.7. Acesso à internet

10.8. Data e hora

10.9. Compressão de dados

10.10. Medição de desempenho

10.11. Controle de qualidade

10.12. Baterias incluídas

11. CRUD e Orientação a objetos