

Linguagem de programação – Python

Relatório

1. Histórico

- **Criação** – é uma linguagem de alto nível, foi criada por Guido Van Rossum em dezembro de 1989. Sua primeira versão oficial foi lançada em fevereiro de 1991.
- **Motivação** – foi criada com a intenção de desenvolver uma linguagem de fácil leitura e escrita, que permitisse uma maior produtividade na programação. Foi projetada para ser uma sucessora da linguagem ABC.

2. Principais IDEs

- **PyCharm** – oferece funcionalidades avançadas, como refatoração de código e suporte a frameworks web como o Django.
- **VSCode** – é uma IDE leve e altamente configurável, que suporta Python através de extensões. Oferece uma grande variedade de plugins que aumentam suas capacidades.
- **Jupyter Notebook** – aplicação web que permite a criação e compartilhamento de documentos que contêm código, visualizações e texto narrativo.
- **Spyder** – voltada principalmente para cientistas de dados e engenheiros, oferecendo uma interface que combina edição de código, debugging e análise interativa de dados.

3. Aplicação

- **O que são** – coleções de módulos que contêm funções e classes que podem ser reutilizadas em diferentes programas. Elas ajudam a economizar tempo e esforço, fornecendo soluções pré-fabricadas para problemas comuns.
- **Como são utilizadas** – para usar uma biblioteca em Python, você precisa primeiro instalá-la usando um gerenciador de pacotes como o

pip. Depois de instalada, você pode importar a biblioteca ou seus módulos específicos no seu código usando a instrução 'import'.

4. Principais Frameworks

- **Django** – framework web de alto nível que permite o desenvolvimento rápido de aplicações web seguras.
- **Flask** – micro-framework web que é leve e flexível, ideal para pequenos projetos e para desenvolvedores que preferem mais controle sobre os componentes da sua aplicação.
- **TensorFlow** – framework de aprendizado profundo que fornece uma interface mais intuitiva e é frequentemente usada para pesquisa e desenvolvimento.
- **Pandas** – uma biblioteca poderosa para análise de dados que fornece estruturas de dados de alto desempenho e ferramentas de análise para Python.
- **NumPy** – biblioteca fundamental para a computação científica com Python, que oferece suporte para arrays multidimensionais e uma vasta coleção de funções matemáticas.

Questões

1. Por que o Python é melhor que o Java?

Python é preferido por sua sintaxe simples e legível, permitindo desenvolvimento rápido e fácil aprendizado.

2. Quantos tipos de dados existem no Python?

Python possui 15 tipos de dados integrados principais: int, float, complex, str, list, tuple, range, dict, set, frozenset, bool, bytes, bytearray, memoryview e none.

3. Qual a diferença entre uma 'tupla' e uma 'lista'?

A principal diferença é que listas são mutáveis e são definidas com colchete, enquanto tuplas são imutáveis e são definidas com parênteses.

4. O que é 'decapagem' e 'desinteressante'?

- Decapagem: processo de serialização de objetos Python em bytes para armazenamento ou transmissão ou para reconstrução dos objetos a partir dos bytes.
- Desinteressante: extração de valores de coleções para variáveis individuais.

5. O que é 'lambda'?

É o que permite a criação de funções anônimas de forma concisa. São frequentemente usadas para operações simples e temporárias.

6. Como a memória é gerenciada no Python?

É gerenciada principalmente através da contagem de referências, onde cada objeto mantém um contador que aumenta quando é referenciado e diminui quando as referências são removidas.

7. O que é 'passar'?

Instrução que não faz nada e é usada como espaço reservado onde a sintaxe exige uma instrução, mas nenhuma ação é necessária.

8. Você pode copiar um objeto em Python?

Sim, porém não todos eles. A maneira mais geral e conhecida de fazer isso é usar o comando `copy.copy()`.

9. Como deletar um arquivo dentro do Python?

Utilizando o comando `os.remove(name_of_the_file)`.

10. O que é um 'dicionário'?

São os inbuilt. São compostos de chaves e os valores correspondentes da chave.

11. Python é uma linguagem interpretada?

Sim, pois é um processo de três vias, você escreve o código-fonte, o python o converte em uma linguagem intermediária e depois é novamente transformado em códigos de máquinas.

12. Como o Python é orientado a objeto?

Linguagens em classes e objetos dessas classes. Python é exatamente isso, mas ela ainda possui os recursos que são creditados os OOPs.

13. O que é 'fatiar'?

É quando você seleciona vários itens de lugares como listas, sequências de caracteres e assim por diante.