Linguagem de Programação

Python

Relatório

Histórico-Criação-Quem? Quando? Motivação?

Quais são os Principais IDEs?

Sua aplicação (Onde é utilizada?)

O que são bibliotecas? Como são utilizadas?

Quais os Principais frameworks(exemplos).

Python foi criado por Guido van Rossum na década 1980, com o objetivo de ser fácil de ler e usar e foi motivada pela necessidade de uma linguagem mais simples e legível para os desenvolvedores. Incluem Pycharm: que oferece recursos avançados, como ferramentas de controle de versão. Visual Studio Code: Vem oferecendo uma interface amigável e personalizável, além de uma comunidade ativa de desenvolvedores contribuindo para sua extensão. Eclipse: Tem recursos como autocompletamento, depuração, verificação de erros e suporte para bibliotecas e Spyder. Em diversas áreas sua aplicação é utilizada como desenvolvimento web, analise de dados e intelig~encia artificial entre outras. As bibliotecas são coleções de código pré-compilado usadas para realizar tarefas específicas, como Numpy, Pandas e Matplotlib. Incluem Django, Flask, pyramide e Bottle, usados para os desenvolvimentos de aplicativos web.

Questões

1.2: Porque o Python é melhor que o Java?

Por várias razões, incluindo sua sintaxe simples e legível, suporte integrada por várias bibliotecas e frameworks populares, melhor desempenho para tarefas e menores e de curtas duração.

1.3: Quantos tipos de Dados existem no Python?

Python possui diversos tipos de dados, incluindo números inteiros (‘int’), números de ponto flutuante (‘float’), strings (‘str’), listas (‘list’), tuplas (‘tuple’), conjuntos (‘set’), dicionários (‘dict’), entre outros.

1. 4: Qual é a diferença entre uma 'tupla' e uma 'lista'?

A principal diferença é que uma lista é mutável (ou seja, pode ser modificada após a criação), enquanto uma tupla é imutável (não pode ser modificada após a criação). Listas são definidas por colchetes ‘[ ]’, e tuplas por parênteses ‘( )’.

1.5: O que é 'decapagem' e 'desinteressante'?

 **Decapagem (stripping)**: Este termo poderia se referir à operação de remover caracteres especiais de uma string, como espaços em branco no início e no fim da string, utilizando métodos como .strip() em Python.

 **Desinteressante (uninteresting)**: Em contexto de programação, isso poderia se referir a um código que não é relevante para a discussão atual, ou a uma operação que não causa impacto significativo ou interessante no código ou na execução do programa.

1.6: O que é 'lambda'?

Em Python, lambda é uma palavra-chave que é usada para definir funções anônimas, ou seja, funções pequenas e temporárias que não são definidas com um nome formal usando def. São úteis para situações onde funções simples são necessárias.

1.7: Como a memória é gerenciada no Python?

 Python utiliza um sistema de gerenciamento de memória automático, conhecido como coleta de lixo (garbage collection). Isso significa que o Python automaticamente gerencia o espaço de memória alocado para objetos que não são mais utilizados, liberando-os para serem reutilizados.

1.8: O que é 'passar'?

 Em Python, ‘pass’ é uma palavra-chave que serve como um espaço reservado para onde um comando é esperado, mas nenhuma ação é necessária. É frequentemente usado quando a sintaxe requer alguma instrução, mas o programador não quer executar nenhum código real.

1.9: Você pode copiar um objeto em Python?

Sim, é possível copiar objetos em Python. Isso pode ser feito utilizando a função copy() para objetos mutáveis como listas e dicionários, ou utilizando o módulo copy para copiar objetos mais complexos.

1.10: Como deletar um arquivo dentro do Python?

Para deletar um arquivo em Python, você pode usar a função os.remove() do módulo os. Por exemplo:

import os os.remove('nome\_do\_arquivo.txt')

1.11: 0 que éum 'dicionário??

Um dicionário em Python é uma estrutura de dados que mapeia chaves a valores. É definido por chaves {} e contém pares chave-valor separados por vírgula. É útil para armazenar e acessar dados de forma rápida através de uma chave, ao invés de um índice numérico.

1.12: Python é uma linguagem interpretada?

Sim, Python é geralmente considerada uma linguagem interpretada. Isso significa que o código fonte é executado linha por linha, traduzindo e executando cada linha conforme o programa é executado.

1.14: Como o Python é orientado a objeto?

Python suporta programação orientada a objetos através de classes e objetos. Tudo em Python é um objeto, incluindo números, strings , listas, e até funções. Classes definem estruturas para objetos e permitem o uso de herança, polimorfismo e encapsulamento.

1.15: 0 que é 'fatiar"?

'Fatiar' em Python refere-se ao processo de acessar partes de uma sequência (como uma lista ou uma string ) utilizando o operador de slicing (:). Isso permite acessar múltiplos elementos ou partes de uma sequência de maneira eficiente.