O comando while é usado para percorrer um objeto iterável. Faz sentido usar while quando sabemos o número exato de vezes que nosso bloco de código deve ser executado. A afirmação é FALSA A afirmação é VERDADEIRA

**O ChatGPT disse:**

ChatGPT

A afirmação é **FALSA**.

O comando while é utilizado para executar um bloco de código enquanto uma condição for verdadeira, e não é a escolha ideal quando sabemos o número exato de iterações. Para isso, geralmente usamos o comando for, que é mais adequado para percorrer objetos iteráveis quando o número de iterações é conhecido. O while é mais apropriado para situações em que o número de repetições pode variar ou depende de condições dinâmicas.

4o mini

**Você disse:**

Range é uma função built-in do Python, ela é usada para produzir uma sequência de números inteiros. A afirmação é FALSA A afirmação é VERDADEIRA

**O ChatGPT disse:**

ChatGPT

A afirmação é **VERDADEIRA**.

A função range é uma função embutida (built-in) em Python que gera uma sequência de números inteiros. É comumente usada em loops, como o for, para iterar um número específico de vezes. Você pode especificar o início, o fim e o passo da sequência.

4o mini

**Você disse:**

Qual o comando utilizado interromper o ciclo de iteração dos comando for e while? continue break stop end Nenhuma das alternativas.

**O ChatGPT disse:**

ChatGPT

O comando utilizado para interromper o ciclo de iteração dos comandos for e while é **break**.

O break encerra o loop imediatamente, enquanto o continue pula para a próxima iteração do loop. As opções "stop", "end" e "Nenhuma das alternativas" estão incorretas.

4o mini

**Você disse:**

Qual o comando utilizado para pular um ciclo de iteração dos comando for e while? skip gonext ignore continue Nenhuma das alternativas.

**O ChatGPT disse:**

ChatGPT

O comando utilizado para pular uma iteração nos comandos for e while é **continue**.

O continue faz com que o loop ignore o restante do código na iteração atual e passe diretamente para a próxima iteração. As outras opções (skip, gonext, ignore e "Nenhuma das alternativas") estão incorretas.

4o mini

**Você disse:**

Quais são as estruturas condicionais disponíveis em Python? Nenhuma das alternativas. if / for / else if / elif / else if / while / else if / for

**O ChatGPT disse:**

ChatGPT

As estruturas condicionais disponíveis em Python são **if / elif / else**.

O comando if é usado para verificar uma condição, elif (que significa "else if") permite verificar múltiplas condições, e else é usado para definir um bloco de código que será executado se nenhuma das condições anteriores for verdadeira. As outras opções apresentadas estão incorretas.

4o mini

**Você disse:**

Qual a função principal da indentação em um programa Python? Nenhuma das alternativas. Indentar código é uma forma de manter o código fonte mais legível e manutenível. Tornar a execução do programa mais rápida, com o código indentado a leitura feita pelo interpretador fica mais performática. Através da indentação o interpretador consegue determinar onde um bloco de comando inicia e onde ele termina. A indentação não exerce nenhum papel principal, ela é opcional. Porém o seu uso é altamente recomendado, por tornar o código mais legível.

**O ChatGPT disse:**

ChatGPT

A função principal da indentação em um programa Python é **através da indentação o interpretador consegue determinar onde um bloco de comando inicia e onde ele termina**.

Em Python, a indentação é fundamental para definir a estrutura do código, já que a linguagem utiliza a indentação para identificar blocos de código em vez de chaves ou palavras-chave como em outras linguagens. Portanto, a indentação não é opcional; ela é essencial para a execução correta do programa.

4o mini

**Você disse:**

Quais são as estruturas de repetição disponíveis em Python? Nenhuma das alternativas. for e else for e while do while e for for e if

**O ChatGPT disse:**

ChatGPT

As estruturas de repetição disponíveis em Python são **for e while**.

O for é usado para iterar sobre uma sequência (como uma lista, tupla ou string), enquanto o while executa um bloco de código enquanto uma condição for verdadeira. As outras opções apresentadas estão incorretas.

texto = " Python ".lstrip(). Qual o valor de texto? Nenhuma das alternativas. " Python" " Python " "Python" "Python "

**O ChatGPT disse:**

ChatGPT

O valor de texto será "Python ". O método lstrip() remove apenas os espaços em branco à esquerda da string, mas não remove os espaços à direita. Portanto, a resposta correta é "Python ".

4o mini

**Você disse:**

Utilizando fatiamento de sequências, como podemos acessar o primeiro caractere da variável curso = "Python"? Nenhuma das alternativas. curso[1] curso[0] curso.first() curso[-1]

**O ChatGPT disse:**

ChatGPT

Para acessar o primeiro caractere da variável curso = "Python", você deve usar curso[0]. Portanto, a resposta correta é:

**curso[0]**.

As outras alternativas estão incorretas:

* curso[1] acessa o segundo caractere.
* curso.first() não é um método válido para strings em Python.
* curso[-1] acessa o último caractere.

4o mini

**Você disse:**

Qual método da classe string converte todas os caracteres para maiúsculo? to\_upper() lower() upper() Nenhuma das alternativas. to\_lower()

**O ChatGPT disse:**

ChatGPT

O método da classe string que converte todos os caracteres para maiúsculo é o upper(). Portanto, a resposta correta é:

**upper()**.

As outras alternativas estão incorretas:

* to\_upper() e to\_lower() não são métodos válidos em Python.
* lower() converte os caracteres para minúsculo.

4o mini

**Você disse:**

No programa: curso = "Python" print(curso[::-1]) Qual será a mensagem exibida para o usuário? nohtyP Python Nenhuma das alternativas. python nohtyp

**O ChatGPT disse:**

ChatGPT

No programa curso = "Python" print(curso[::-1]), a mensagem exibida será nohtyP. O fatiamento [::-1] inverte a string. Portanto, a resposta correta é:

**nohtyP**.

4o mini

**Você disse:**

No programa: PI = 3.14159 print(f"Valor de PI: {PI:.2f}") Qual será a mensagem exibida para o usuário? Valor de PI: 3 Valor de PI: 3.0 Valor de PI: 3.14159 Valor de PI: 3.14159 Valor de PI: 3.14

**O ChatGPT disse:**

ChatGPT

No programa PI = 3.14159 print(f"Valor de PI: {PI:.2f}"), a mensagem exibida será:

**Valor de PI: 3.14**.

O formato :.2f indica que o valor deve ser exibido com duas casas decimais.

4o mini

**Você disse:**

Strings de múltiplas linhas são definidas informando 3 aspas simples ou duplas durante a atribuição. Elas podem ocupar várias linhas do código, e todos os espaços em branco são incluídos na string final. A afirmação é VERDADEIRA A afirmação é FALSA

**O ChatGPT disse:**

ChatGPT

A afirmação é **VERDADEIRA**.

Strings de múltiplas linhas em Python podem ser definidas usando três aspas simples (''') ou três aspas duplas ("""). Elas podem ocupar várias linhas no código e todos os espaços em branco, incluindo quebras de linha, são incluídos na string final.

4o mini

**Você disse:**

No programa: curso = "Python" print(f"Bem vindo ao curso de {curso.upper()}!") Qual será a mensagem exibida para o usuário? Nenhuma das alternativas. "Bem vindo ao curso de {curso.upper()}!" "Bem vindo ao curso de python!" "Bem vindo ao curso de Python!" "Bem vindo ao curso de PYTHON!"

**O ChatGPT disse:**

ChatGPT

No programa curso = "Python" print(f"Bem vindo ao curso de {curso.upper()}!"), a mensagem exibida será:

**"Bem vindo ao curso de PYTHON!"**.

O método upper() converte todos os caracteres da string curso para maiúsculo.

4o mini

Parte superior do formulário



Parte inferior do formulário