

Funções internas (built-in functions) que não precisam de importação para serem utilizadas:

Funções de Manipulação de Dados

- len() - Retorna o comprimento de um objeto.
- type() - Retorna o tipo do objeto.
- str() - Converte um objeto em string.
- int() - Converte um objeto em número inteiro.
- float() - Converte um objeto em número de ponto flutuante.
- list() - Converte um objeto em lista.
- dict() - Converte um objeto em dicionário.
- set() - Converte um objeto em conjunto.
- tuple() - Converte um objeto em tupla.
- enumerate() - Retorna um objeto enumerado (índice + valor).

Funções Matemáticas

- abs() - Retorna o valor absoluto de um número.
- round() - Arredonda um número.
- max() - Retorna o maior valor de uma lista ou iterável.
- min() - Retorna o menor valor de uma lista ou iterável.
- sum() - Retorna a soma de um iterável.
- pow() - Calcula a potência (base, expoente).
- divmod() - Retorna o quociente e o resto da divisão.

Funções de Controle de Fluxo

- all() - Retorna True se todos os itens do iterável forem verdadeiros.
- any() - Retorna True se pelo menos um item do iterável for verdadeiro.
- filter() - Filtra elementos de um iterável com base em uma função.
- map() - Aplica uma função a cada item de um iterável.
- zip() - Combina dois ou mais iteráveis em tuplas.

Funções de Entrada e Saída

- print() - Imprime no console.
- input() - Recebe entrada do usuário.
- open() - Abre um arquivo.

Funções de Manipulação de Strings

- len() - Retorna o comprimento da string.
- ord() - Retorna o código Unicode de um caractere.
- chr() - Retorna o caractere Unicode de um código.
- format() - Formata strings.

Funções de Manipulação de Coleções

- sorted() - Retorna uma nova lista ordenada.
- reversed() - Retorna um iterador com os elementos invertidos.
- range() - Gera uma sequência de números.
- slice() - Cria um objeto de slice (subconjunto).
- del - Remove elementos de uma coleção.

Funções para Debug e Execução

- help() - Mostra ajuda sobre um objeto.
- dir() - Mostra os atributos e métodos de um objeto.
- id() - Retorna o identificador único de um objeto.
- eval() - Avalia uma expressão como código Python.
- exec() - Executa um código Python.
- globals() - Retorna o dicionário de variáveis globais.
- locals() - Retorna o dicionário de variáveis locais.

Funções Diversas

- `isinstance()` - Verifica se um objeto é de um tipo específico.
- `hash()` - Retorna o hash de um objeto.
- `callable()` - Verifica se o objeto é chamável.
- `bin()` - Converte um número inteiro para binário.
- `hex()` - Converte um número inteiro para hexadecimal.
- `oct()` - Converte um número inteiro para octal.

1. Built-in

Functions

- `repr()` - Retorna uma representação string de um objeto.
- `bytes()` - Cria um objeto de bytes.
- `memoryview()` - Cria uma visão de memória do objeto.
- `ascii()` - Retorna uma versão string de um objeto com caracteres não-ASCII escapados.
- `complex()` - Cria um número complexo.

2. Explorando com o `dir` e `help`:

- Use `dir(__builtins__)` no Python para listar todas as funções built-in.
- Use `help(funcao)` para obter informações detalhadas de qualquer função.

3. Funções em Módulos e Pacotes Padrão: Além das funções built-in, existem muitas outras em módulos como:

- **math** (operações matemáticas avançadas): `math.sqrt`, `math.sin`, `math.log`, etc.
- **random** (números aleatórios): `random.randint`, `random.choice`, `random.shuffle`.
- **datetime** (datas e horários): `datetime.now`, `datetime.timedelta`.
- **os** (sistema operacional): `os.path.join`, `os.listdir`, `os.remove`.
- **sys** (interação com o interpretador): `sys.argv`, `sys.exit`.
- **itertools** (ferramentas para iteráveis): `itertools.combinations`, `itertools.permutations`.
- **functools** (ferramentas de funções): `functools.reduce`, `functools.lru_cache`.

Funções mais comuns em Python para manipulação de listas, dicionários, tuplas, conjuntos, e strings:

Manipulação de Listas

1. **append(item)** - Adiciona um item ao final da lista.
 2. **extend(iterável)** - Adiciona múltiplos elementos de um iterável à lista.
 3. **insert(index, item)** - Insere um item em uma posição específica.
 4. **remove(item)** - Remove a primeira ocorrência do item especificado.
 5. **pop([index])** - Remove e retorna o item no índice especificado (ou o último, se o índice for omitido).
 6. **clear()** - Remove todos os elementos da lista.
 7. **index(item[, start, end])** - Retorna o índice do primeiro item encontrado na lista.
 8. **count(item)** - Conta quantas vezes um item aparece na lista.
 9. **sort(key=None, reverse=False)** - Ordena os elementos da lista (in-place).
 10. **reverse()** - Inverte a ordem da lista (in-place).
 11. **copy()** - Retorna uma cópia superficial da lista.
 12. **len(lista)** - Retorna o número de elementos na lista.
 13. **max(lista) / min(lista)** - Retorna o maior ou menor elemento da lista.
 14. **sum(lista)** - Retorna a soma dos elementos (se forem numéricos).
 15. **enumerate(lista)** - Retorna um objeto enumerado com índice e valor.
-

Manipulação de Dicionários

1. **keys()** - Retorna todas as chaves do dicionário.
2. **values()** - Retorna todos os valores do dicionário.
3. **items()** - Retorna pares (chave, valor) como tuplas.
4. **get(chave[, default])** - Retorna o valor de uma chave, ou um valor padrão se a chave não existir.

5. **pop(chave[, default])** - Remove e retorna o valor associado à chave especificada.
 6. **popitem()** - Remove e retorna o último par (chave, valor) inserido.
 7. **clear()** - Remove todos os itens do dicionário.
 8. **update(outro_dict)** - Atualiza o dicionário com outro dicionário ou pares chave-valor.
 9. **setdefault(chave[, default])** - Retorna o valor da chave, ou adiciona a chave com um valor padrão, se não existir.
 10. **len(dicionário)** - Retorna o número de itens no dicionário.
 11. **del dicionário[chave]** - Remove a chave especificada.
 12. **in** - Verifica se uma chave está no dicionário.
-

Manipulação de Tuplas

1. **len(tupla)** - Retorna o número de elementos.
 2. **index(item)** - Retorna o índice do item especificado.
 3. **count(item)** - Conta a ocorrência do item.
 4. **in** - Verifica se um item está na tupla.
-

Manipulação de Conjuntos (Set)

1. **add(item)** - Adiciona um item ao conjunto.
2. **update(iterável)** - Adiciona múltiplos itens ao conjunto.
3. **remove(item)** - Remove o item do conjunto (gera erro se não existir).
4. **discard(item)** - Remove o item do conjunto (não gera erro se não existir).
5. **pop()** - Remove e retorna um item aleatório.
6. **clear()** - Remove todos os itens do conjunto.
7. **union(outro_set) / |** - Retorna a união de dois conjuntos.
8. **intersection(outro_set) / &** - Retorna a interseção de dois conjuntos.

9. **difference(outro_set) / -** - Retorna os elementos presentes apenas no conjunto atual.
 10. **symmetric_difference(outro_set) / ^** - Retorna elementos únicos em cada conjunto.
 11. **issubset(outro_set)** - Verifica se o conjunto atual é um subconjunto.
 12. **issuperset(outro_set)** - Verifica se o conjunto atual é um superconjunto.
-

Manipulação de Strings

1. **len(string)** - Retorna o comprimento da string.
 2. **lower() / upper()** - Converte para minúsculas ou maiúsculas.
 3. **strip()** - Remove espaços ou caracteres das extremidades.
 4. **split(separador)** - Divide a string em uma lista.
 5. **join(iterável)** - Junta elementos de um iterável em uma única string.
 6. **replace(antigo, novo)** - Substitui partes da string.
 7. **find(substring)** - Retorna o índice da primeira ocorrência da substring.
 8. **startswith(prefixo) / endswith(sufixo)** - Verifica se a string começa ou termina com um prefixo ou sufixo.
 9. **isnumeric() / isdigit() / isalpha() / isalnum()** - Verifica se a string é numérica, contém apenas dígitos, letras ou uma combinação.
 10. **format() / f-strings** - Formata a string.
 11. **count(substring)** - Conta o número de ocorrências de uma substring.
 12. **capitalize()** - Coloca a primeira letra em maiúscula.
 13. **title()** - Coloca a primeira letra de cada palavra em maiúscula.
 14. **center(width)** - Centraliza a string.
-