1. Fatiamento de String

Sintaxe / Função	O que faz	Exemplo
`texto[inicio:fim]`	Pega do índice `inicio` até `fim - 1`	`"Felipe"[0:3] → 'Fel'`
`texto[:fim]`	Pega do início até `fim - 1`	`"Felipe"[:4] → 'Feli'`
`texto[inicio:]`	Pega de `inicio` até o fim	`"Felipe"[2:] → 'lipe'`
`texto[::passo]`	Pega pulando de `passo` em `passo`	`"Felipe"[::2] → 'Flp'`
`texto[::-1]`	Inverte a string	`"Felipe"[::-1] → 'epileF'`
`texto[-n:]`	Pega os últimos `n` caracteres	`"Felipe"[-2:] → 'pe'`
`texto[:-n]`	Pega tudo, menos os últimos `n`	`"Felipe"[:-2] → 'Feli'`

2. Análise de String

Função	O que faz	Exemplo
`len(texto)`	Conta caracteres	`len("Felipe") → 6`
`texto.count("e")`	Conta quantas vezes aparece	`"Felipe".count("e") → 2`
`texto.find("li")`	Índice da 1ª ocorrência (ou -1)	`"Felipe".find("li") → 2`
`texto.index("li")`	Igual ao `find`, mas dá erro se não achar	`"Felipe".index("li") → 2`
`"sub" in texto`	Verifica se existe	`"li" in "Felipe" → True`
`texto.startswith("Fe")`	Começa com...?	`"Felipe".startswith("Fe") → True`
`texto.endswith("pe")`	Termina com...?	`"Felipe".endswith("pe") → True`
`texto.isalpha()`	Só letras?	`"Felipe".isalpha() → True`
`texto.isdigit()`	Só números?	`"123".isdigit() → True`
`texto.isalnum()`	Letras e números?	`"abc123".isalnum() → True`
`texto.isspace()`	Só espaços?	`" ".isspace() → True`

3. Transformações de String

Função	O que faz	Exemplo
`texto.replace("a", "o")`	Troca texto	`"pato".replace("a", "o") → 'poto'`
`texto.upper()`	Tudo maiúsculo	`"Felipe".upper() → 'FELIPE'`
`texto.lower()`	Tudo minúsculo	`"Felipe".lower() → 'felipe'`
`texto.capitalize()`	1ª letra maiúscula	`"feLiPe".capitalize() → 'Felipe'`
`texto.title()`	1ª letra de cada palavra maiúscula	`"felipe henrique".title() → 'Felipe Henrique'`
`texto.strip()`	Remove espaços nas pontas	`" teste ".strip() → 'teste'`
`texto.lstrip()`	Remove espaços à esquerda	`" teste".lstrip() → 'teste'`
`texto.rstrip()`	Remove espaços à direita	`"teste ".rstrip() → 'teste'`
`texto.swapcase()`	Inverte maiúsculas/minúsculas	`"FeLiPe".swapcase() → 'fElIpE'`
`texto.center(10, "-")`	Centraliza com preenchimento	`"abc".center(10, "-") → '---abc----'`
`texto.zfill(5)`	Adiciona zeros à esquerda	`"42".zfill(5) → '00042'`

4. Junção e Separação

Função	O que faz	Exemplo
`sep.join(lista)`	Junta elementos usando `sep`	`"-".join(["a","b","c"]) → 'a-b-c'`
`texto.split()`	Divide por espaço	`"a b c".split() → ['a','b','c']`
`texto.split(",")`	Divide pelo separador	`"a,b,c".split(",") → ['a','b','c']`
`texto.partition("x")`	Divide em 3 partes: antes, separador, depois	`"abcxdef".partition("x") → ('abc','x','def')`
`texto.rsplit(" ", 1)`	Divide a partir da direita, limite de cortes	`"um dois tres".rsplit(" ", 1) → ['um dois', 'tres']`