


→ Tomorrow Aman will take a session on zoom
viz not assignment solving. He will discuss your
general doubts

→ Can we have an extended 30 min session ^{non-compulsory} after
class just for doubts? 57% yes no

Agenda → Strings → slicing in strings / lists

→ operators for strings

→ unbuil functions on string

→ problems

→ formality of strings

→ mutability of strings

⇒ # string indexing

①

→ In python strings are indexed just like lists
i.e the first character is given 0th index.

"codechef"
↓ ↓ ↓ ↓ ↓ ↓ ↓
0 1 2 3 4 5 6 7

→ first character → 0
last character → $\text{len}(\text{string}) - 1$

#dictionary order is also called as lexicographical order

→ formatting strings in python ↔ string interpolation

↳ You can add a variable inside a string on runtime without using concatenation (C-style way)

fill in the blanks

always want a string here

" welcome, _____, to the year. "

you don't want any type of variable to fill here

In more fill ups we will pass variables

you always want an integer here

The diagram illustrates string interpolation using a template string: " welcome, _____, to the year. ". Annotations include: "fill in the blanks" at the top; "always want a string here" with an arrow pointing to the opening quote; "you don't want any type of variable to fill here" with an arrow pointing to the blank space; "In more fill ups we will pass variables" with an arrow pointing to the blank space; and "you always want an integer here" with an arrow pointing to the closing quote. A question mark is placed above the blank space, and a red 'X' is drawn over the blank space and the closing quote.

→ to apply this rule use format specifier:

dod → int

dos → string

fof → float

foijs → boolean

⋮
⋮
⋮
⋮
⋮

⋮

2
"codechef"
dech

Slicing \rightarrow string / list

$l = [1, 2, 3, 4, 5, 6, 7, 8, 9]$
1 2 3 +1

l [start index : end index + 1 : jump]
after optional \rightarrow default + 1