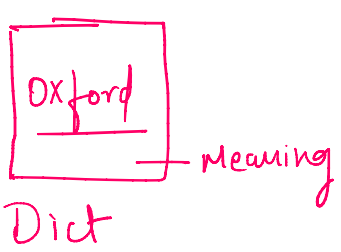
List 1

Wednesday, July 26, 2023 8:58 PM

Data Structures



Alpha

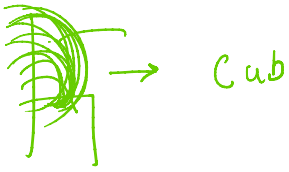
*

Random

Data loss ×

Arranged ×

• → Structure the Data



- Organize, Store ✓
- Process ✓
- Retrieve ✓

name = Virat Kholi

age =

run_1 = 72

run_2 = 34

run_3 = 75

⋮

370

runs_virat = [73, 34, 21, 18 ..... 40]

List - [ ]

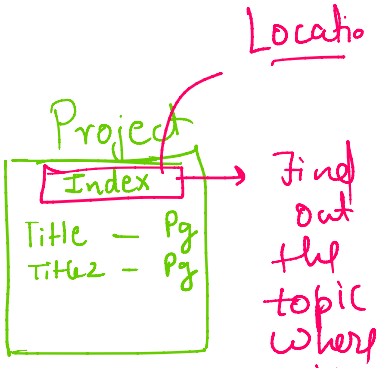
→ comma seperated

→ order → you provide

→ no limit

How can I access

Indexing → [ ]



runs_virat = [67, 32, 41, 92, 73, 21]
0 1 2 3 4 5

Indexing start from 0

runs_virat [1]

Ans - 32

runs_virat [2]
41

How many Matched

How many Matched

Total elements in list

len(runs_virat)

↳ 6

→ How I get virat last match score

runs_virat = [67, 32, 41, 92, 73, 21]
0 1 2 3 4 5

→ runs_virat[len(runs_virat)-1]

6-1

5

Negative Indexing

runs_virat = [67, 32, 41, 92, 73, 21]
-6 -5 -4 -3 -2 -1

Negative → starts -1

Negative → starts

runs_virat [-1]

↳ 21

runs_virat [-5]

↳ 32

runs_virat = [67, 32, 41, 92, 73, 21]
-6 -5 -4 -3 -2 -1

#small

runs_virat [- len(runs_virat)]

↓

-6

67

→ Print the total of runs on odd Position

1 2 3 4 5 6

runs_virat = [67, 32, 41, 92, 73, 21]
0 1 2 3 4 5

runs_virat[0] + runs_virat[2] + runs_vi[4]

→ List → Method

function

runs_virat = [67, 32, 41, 92, 73, 21]

add More runs

→ runs_virat.append(26)

[67, 32, 41, 92, 73, 21, 26]

end of the list

def square(x)
return x^2

} Task

def append } It is only made for list

runs_virat = [67, 32, 41, 92, 73, 21]

0 1

Insert at particular loc.

[67, New, 32, 41, 92, 73, 21]

[67, New, 32, 41, 92, 73, 21]

runs_virat.insert(1, 100) Runs

[67, 100, 32, 41, 92, 73, 21]

runs_virat.insert(0, 25)

25, 67, 10. - — ~ - .

→ [32, 99, 21]

runs_virat.extend([32, 99, 21]) L

Multiple Values

but only at the end

( ) [25, 67, 100, 99, 21, 32, 99, 21]

→ Iterating List

for i in range(5): 0, 1, 2, 3, 4

Print(i)

iterable object 0, 1, 2, 3, 4

Iterator

0
1
2
3
4

0, 1, 2, 3, 4

one by one element

List is also iterable

[26, 34, 29]

for i in [34, 75, 21, 99, 45]

Print (i)

34
75
21
99
45