List:

A white background with red text

AI-generated content may be incorrect.

What is the List:

Creation:

L1=[1,2,3,4,5,6]

print(L1)

# [1, 2, 3, 4, 5, 6]

L2=[1.8,1.8,3.5,4.6,5.7]

print(L2)

#[1.8, 1.8, 3.5, 4.6, 5.7]

L3=["John","Angle"]

print(L3)

#['John', 'Angle']

#List(iterable)

L4=list((3,5,7,9))

print(L4)

#[3, 5, 7, 9]

L5=list("abcde")

print(L5)

#['a', 'b', 'c', 'd', 'e']

L6=[]

print(L6)

#[]

Representation

L1=[6,5,4,2,3,2]

for i in L1:

    print(i,end=' ')

#   6 5 4 2 3 2

Heterogeneous

L1=[7,3.2,"john",True,5+6j]

for i in L1:

    print(i,end=' ')

# 7 3.2 john True (5+6j)

A screenshot of a computer

AI-generated content may be incorrect.

L1=[7,3.2,"john",True,5+6j]

for i in L1:

    print(i,end=' ')

# 7 3.2 john True (5+6j)

L1[3]=5

print(L1)

# [7, 3.2, 'john', 5, (5+6j)]

L1[2]="Raghu"

print(L1)

# [7, 3.2, 'Raghu', 5, (5+6j)]

L1[2][3]='a'

print(L1)

#  TypeError: 'str' object does not support item assignment

Mutable

L1=[2,4,6,8,10,12]

L1[2]=16 #Modify

L1.append(25) #Append

print(L1)

# [2, 4, 16, 8, 10, 12, 25]

L1=[2,4,6,8,10,12]

L1[2]=16 #Modify

L1.append(25) #Append

print(L1)

# [2, 4, 16, 8, 10, 12, 25]

* Ordered collection of heterogenous elements
* It’s a mutable
* Can have duplicates

INDEXING AND SLICING

Read:

# list is mutable

#Read/Write

#Read

 # Indexing

 # slicing

#Read Indexing

L1=[3,6,9,12,15,18,21]

print(L1[4])

# 15

#Slicing

print(L1[::]) #Start,Stop,Step

# [3, 6, 9, 12, 15, 18, 21]

print(L1[:]) #Begin,#End

# [3, 6, 9, 12, 15, 18, 21]

print(L1[2:]) #2 to End

#[9, 12, 15, 18, 21]

print(L1[:6]) #0 to 5

[3, 6, 9, 12, 15, 18]

print(L1[2:6]) #2 to 5

# [9, 12, 15]

print(L1[-5:-2]) #-ve indexing

# [9, 12, 15]

#forward print

#9,12,15

print(L1[::])

#[3, 6, 9, 12, 15, 18, 21]

print(L1[::1])

# [3, 6, 9, 12, 15, 18, 21]

print(L1[::2])

#[3, 9, 15, 21]

print(L1[::3])

# [3, 12, 21]

print(L1[::-1])

# [21, 18, 15, 12, 9, 6, 3]

#Reverse,backward,indices also backwards

print(L1[4::-1])

# [15, 12, 9, 6, 3]

print(L1[4:0:-1])

# [15, 12, 9, 6]

print(L1[-3:-7:-1])

# [15, 12, 9, 6]

Indexing and Slicing (Write)

L1=[1,2,3,4,5]

L1[1]=10

print(L1)

# [1, 10, 3, 4, 5]

L1[3]=[10,11]

print(L1)

#Nested List

# [1, 10, 3, [10, 11], 5]

# L1=[1,2,3,4,5]

#L1[start:stop] #  any no of elements can insert

# L1[start:stop:step]

#step no of elements

L1=[1,2,3,4,5]

L1[0:0]=[10]

print(L1)

# [10, 1, 2, 3, 4, 5]

L1[3:3]=[10]

print(L1)

# [10, 1, 2, 10, 3, 4, 5]

L1=[1,2,3,4,5]

L1[5:5]=[10]

print(L1)

#[1, 2, 3, 4, 5, 10]

L1=[1,2,3,4,5]

L1[9:9]=[10]

print(L1)

#[1, 2, 3, 4, 5, 10]

L1=[1,2,3,4,5]

L1[3:3]=[10,11,12]

print(L1)

# [1, 2, 3, 10, 11, 12, 4, 5]

L1=[1,2,3,4,5]

L1[1:4]=[10]

#as many will values we can ive

#1 10 15 (2,3,4)--> Removed

print(L1)

# [1, 10, 5]

L1=[1,2,3,4,5]

L1[1:4]=[10,11]

print(L1)

#[1, 10, 11, 5]

L1=[1,2,3,4,5]

L1[1:4]=[10,11,12,13]

print(L1)

#[1, 10, 11, 12, 13, 5]

### step

# need to give exact number

L1=[1,2,3,4,5]

L1[::2]=[10,11,12]

#1 3 5 replace

print(L1)

# [10, 2, 11, 4, 12]

L1=[1,2,3,4,5]

L1[::-1]=[10,11,12,13,14]

print(L1)

# [14, 13, 12, 11, 10]

L1=[1,2,3,4,5]

L1[3:0:-1]=[12,13,14]

#

print(L1)

#[1, 14, 13, 12, 5]

A screenshot of a computer screen

AI-generated content may be incorrect.

List Concatination and repetation:

Operations on list