How to Change/Replace an Element of a list? In [1]: x=[10,20,30,40,50] x[1]=200 Х [10, 200, 30, 40, 50] In [2]: #Syntax list_name[index]=new_value x=[10, 20, 30, 40, 50, 20, 30, 30, 20]value=20 count=0 for i in x: if i==value: print(i) count=count+1 print(count) break print(x) x[count]=200 20 [10, 20, 30, 40, 50, 20, 30, 30, 20] [10, 200, 30, 40, 50, 20, 30, 30, 20] Introduction to Tuple In []: --> Tuple is exactly same as list but the only difference between list and tuple is lists are mutable(Changeable) and tuples are immutable(Not changeable). --> Tuple elements are fixed you can **not** perform any change **in** it(Immutability). In other words we can also say that tuple is a read only mode of list. --> Indexing is very important is case of tuple(Ordered collection of element) --> It stores disimilar data like list. --> Duplicates are allowed in case of Tuple --> Parenthesis are used to represent a Tuple. Creation of Tuple Object In [20]: #if you know the tuple element x=(10,20,30,40)type(x) tuple Out[20]: In [21]: #if you want to implement empty tuple x=tuple() type(x) tuple Out[21]: In [22]: **x=()** type(x) tuple Out[22]: In [24]: #Creation of tuple with one element X=(10,)type(x) tuple In [27]: x=10,20,"str",40 #Tuple Packing print(x) #Whatever elements are there all elements are packed together to form a tuple. (10, 20, 'str', 40) In []: Note --> in case of tuple we don't have any method for insertion, deletion, updation because tuple is immutable that means we cannot perform any change in the tuple object. We can only access the content of the tuple. Slicing and Indexing in Tuple How we can access the element of a tuple: 1. Indexing 2.Slicing In [31]: #Indexing --> tuple_name[index_value] #Slicing --> tuple_name[begin_index : end_index :step] x=(10,20,30,40,50)print(x[1]) #20 #print(x[5]) #errorprint(x[::]) #whole tuple #print(x[80]) #error print(x[2:9]) #30 40 50 print(x[::-1]) #50 40 30 20 10 print(x[::-4]) #50 10 print(x[10::9]) #empty tuple print(x[0:0:0]) #error (10, 20, 30, 40, 50) (30, 40, 50) (50, 40, 30, 20, 10) (50, 10)() () In [36]: x=("Python", "Hello", "98", 90, 209) print(x[1][1])#e print(x[0][4]) #0 print(x[3][4]) #error print(x[5][1]) #error **TypeError** Traceback (most recent call last) Input In [36], in <cell line: 4>() 2 print(x[1][1])#e 3 print(x[0][4]) #0 ----> **4** print(x[3][4]) #error 5 print(x[5][1]) **TypeError**: 'int' object is not subscriptable In [43]: x=[10,20,[20,30,(70,80)],[50,60]] x[2][2][1] Out[43]: Tuple vs Immutability In [44]: x=(10,20,30,40,50)x[1]=200 Χ Note --> Tuple are immutable, Immutable means we cannot perform any change in it If we will try to perform any change then either you will get an error or a new object will be created **TypeError** Traceback (most recent call last) Input In [44], in <cell line: 2>() $1 \times (10, 20, 30, 40, 50)$ ----> **2** x[1]=200 3 X TypeError: 'tuple' object does not support item assignment **Concatenation Operator** In [45]: #Join two Tuple x=(10,20,30)y=(50,60,70)z= x+y(10, 20, 30, 50, 60, 70) **Repitition Operator** In [50]: #Repeat the Tuples x=(10,20,30)y=x*3 У (10, 20, 30, 10, 20, 30, 10, 20, 30) Membership Operator In [47]: #Check weather a given element is member of the tuple or not x=(10, 20, 30, 40)40 not in x True Out[47]: Max and Min Function Max Function --> It will give you the maximum element of the sequence. Min Function --> It will give you the minimum element of the sequence. Nite --> Max and Min function will work with list as well as tuple In [51]: x=[10,20,30,40] max(x)Out[51]: In [52]: x=[10,20,30,40] min(x)Out[52]: In [53]: x=(10,20,30,40)min(x)10 Out[53]: In [55]: x=(10,20,30,40)max(x)Out[55]: In [57]: x=(10,20,30,40)sorted(x) x[-1:-3:-1](40, 30)Out[57]: In [58]: $x=\{10,20,30\}$ max(x)Out[58]: **Python Problems** In []: Write a Python program which iterates the integers **from** 1 to 50. For multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print "FizzBuzz". In [7]: **for** i **in** range(1,51): **if** i**%3**==0: print("Fizz") **elif** i**%5**==0: print("Buzz") **elif** i%3==0 and i%5==0: print("FizzBuzz") Fizz Buzz Fizz Fizz Buzz Fizz Fizz Fizz Buzz Fi77 Fi77 Buzz Fizz Fizz Fizz Buzz Fizz Fizz Buzz Fizz Fizz Fizz Buzz Lists Vs Tuple In []: List Tuple 1.List is represented with the Tuple is represented with parenthesis help of square brackets. 2.Lists are mutable. Tuples are immutable 3.List is slower than Tuple Tuple is Faster 4.List consumes more memeory Tuple consumes less memeory 5. If you want to perform any updation Tuple is useful to access the element like deletion, insertion etc then you should use list Introduction to Dictionary In []: Dictionary --> if you want to represent a group of element in the form of keys and values then we should use dictionary. name: "Pratyush" batch:DS290922A college : Amity Properties of Dictionaries: 1.keys cannot be duplicate if you are using duplicate keys then only last one will be considered 2.Disimilar objects are allowed **for** keys **as** well **as** values. 3.Indexing is not possible(unordered collection of data) 4.Dictionaries are mutable(chnageable) 5.indexing and slicing concept is not applicable in case of dict and set. 6. Mutable objects cannot be used **as** a key. 7. Curly braces are used to represent a dictionary and each and every key value pair is sepreated by colon. Creation of dictionary object In []: Empty dictionary: d={} d=dict() If you know the element: d={"name":"Pratyush", "class":90} How to add entry in dictionary In [60]: d={} d["name"]="Pratyush" d["batch"]="DS290922" {'name': 'Pratyush', 'batch': 'DS290922'} How to Access Elements from a Dictionary We can only access the elements of a dictionary with the help of keys. Syntax: dictname[key] In [3]: d={1:2,3:4,5:6} d[3] Out[3]: How to Update the Value of a dictionary In []: We can update the value in the dictionary with the help of keys. Syntax: dict_name[key]=updated_value In [70]: d={"name":"Pratyush", "City":"Delhi", "Mobile_no":9721890876, "Education":"M.Tech"} print(d) d["Batch"]="DS290922A" print(d) d["name"]="Noor" d["Mobile_no"]=9000000 print(d) {'name': 'Pratyush', 'City': 'Delhi', 'Mobile_no': 9721890876, 'Education': 'M.Tech'} {'name': 'Pratyush', 'City': 'Delhi', 'Mobile_no': 9721890876, 'Education': 'M.Tech', 'Batch': 'DS290922A'} {'name': 'Noor', 'City': 'Delhi', 'Mobile_no': 9000000, 'Education': 'M.Tech', 'Batch': 'DS290922A'} Note --> If you want to access any value of the dicitonary you can access with the help of key. if the key is not present you will get the key error How to delete a key from the dictionary We can delete the key value pair from the dictionary with the help of del keyword if we want to delete a key value pair then Syntax: del dict_name[key] **if** we want to delete whole dictionary then Syntax: del dict_name if we want to delete the all content of a dictionary then Syntax: dict_name.clear() del Keyword d={"name":"Pratyush","City":"Delhi","Mobile_no":9721890876,"Education":"M.Tech"} print(d) del d["City"] print(d) d["State"]="Uttar Pradesh" print(d) {'name': 'Pratyush', 'City': 'Delhi', 'Mobile_no': 9721890876, 'Education': 'M.Tech'} {'name': 'Pratyush', 'Mobile_no': 9721890876, 'Education': 'M.Tech'} {'name': 'Pratyush', 'Mobile_no': 9721890876, 'Education': 'M.Tech', 'State': 'Uttar Pradesh'} clear function In []: clear function --> If we want to delete all key value pair from the dictionary then we should use clear function. In [75]: d={"name":"Pratyush","City":"Delhi","Mobile_no":9721890876,"Education":"M.Tech"} print(d) d.clear() print(d) {'name': 'Pratyush', 'City': 'Delhi', 'Mobile_no': 9721890876, 'Education': 'M.Tech'} {} Report Card Problem In [82]: n=int(input("Enter the Number of Student : ")) #5 1=[] for i in range(n): Name=input("Enter Student Name") Rollno=int(input("Enter Student Roll No :")) Class=int(input("Enter Student Class :")) Section=int(input("Enter Student Section :")) print("Student Name is ", Name) print("Roll No is ", Rollno) print("Class is ",Class) print("Section is ", Section) list = ["Maths", "Science", "Hindi", "Python", "Java"] for j in list: ele=float(input(f"Enter Your Marks in {j}")) 1.append(ele) print(str(j)+" Marks are "+str(ele)) print("Total Marks :500 ") print("Obtained Marks of the Student are : ", sum(1)) y=sum(1)//len(1)print("Average of the Student : ",y) print("Student Grade and Position : ") **if** y>90 **and** y<100: print("Congratulations Your Grade is A") **if** y>95 **and** y<100: print("You are a Tooper and You will get Gold Medal Very Soon") print("Sorry You got A grade but you are not Eligible for Gold Medal") **elif** y>80 **and** y<90: print("You Got B Grade") **elif** y>70 **and** y<80: print("You Got C Grade") **elif** y>60 **and** y<70: print("You Got D Grade") **elif** y>50 **and** y<60: print("You Got E Grade") print("Sorry You are Failed") print("Thanking You Hope You Got Good Marks!!!!") print() print(""" Autherization Signature""") Enter the Number of Student : 1 Enter Student Namenamish Enter Student Roll No :98 Enter Student Class :12 Enter Student Section :9 *********************** Student Name is namish Roll No is 98 Class is 12 Section is 9 ************************* Enter Your Marks in Maths98.23 Maths Marks are 98.23 Enter Your Marks in Science23.34 Science Marks are 23.34 Enter Your Marks in Hindi23.57 Hindi Marks are 23.57 Enter Your Marks in Python12.34 Python Marks are 12.34 Enter Your Marks in Java345.45 Java Marks are 345.45 Total Marks :500 Obtained Marks of the Student are : 502.93 Average of the Student: 100.0 Student Grade and Position : Sorry You are Failed Thanking You Hope You Got Good Marks!!!! Autherization Signature