

data structure

```
a=[1,2,3,4,5,6,7,8]
print(a)

[1, 2, 3, 4, 5, 6, 7, 8]

b=[1,2,3,"raki"]
print(b)

[1, 2, 3, 'raki']
```

accessing item

```
print(a[0])
print(a[-1])

1
8
```

modifying items

```
a[0]=10
print(a)

[10, 2, 3, 4, 5, 6, 7, 8]
```

adding items append

```
a.append(6)
a

[10, 2, 3, 4, 5, 6, 7, 8, 6, 6]

#insert item
a.insert(1,15)
a

[10, 15, 15, 15, 2, 3, 4, 5, 6, 7, 8, 6, 6]
```

removing items

```
#remove item
a.remove(15)
a

[10, 15, 15, 2, 3, 4, 5, 6, 7, 8, 6, 6]
```

```
#pop remove item
a.pop(1)
a
[10, 15, 2, 3, 4, 5, 6, 7, 8, 6, 6]
```

other operations

```
#len
s=[10,20,30,40,50]
len(s)
5

#sort
car=["maruti","honda","tata"]
car.sort()
car
['honda', 'maruti', 'tata']

d=(1,2,3,4,5,6,7,)
(1, 2, 3, 4, 5, 6, 7, 8)
```

iterating through alist

```
for i in a:
    print(i)

10
15
2
3
4
5
6
7
8
6
6
```

tupels

```
r=(10,20,30,40,50)
print(r[3])

40
```

dictionary

```

dog={
    "name": "rocky",
    "age": 12,
    "breed": "pug"
}
dog

{'name': 'rocky', 'age': 12, 'breed': 'pug'}

#accessing
print(dog["name"])

rocky

#modifying
dog["age"]=10
dog

{'name': 'rocky', 'age': 10, 'breed': 'pug'}

#remove
del dog["breed"]
dog

{'name': 'rocky', 'age': 10}

#iterating
for key,value in dog.items():
    print(key,value)

name rocky
age 10

```

set

```

naman={1,2,3,4,5,6}
naman

{1, 2, 3, 4, 5, 6}

#adding
naman.add(0)
naman

{0, 1, 2, 3, 4, 5, 6, 7, 9}

#removing
naman.remove(5)
naman

{0, 1, 2, 3, 4, 6, 7, 9}

```

set operation

```
#union
t={1,2,3,4}
m={5,6,7,8}
t|m

{1, 2, 3, 4, 5, 6, 7, 8}

#intersection
t&m

set()

#difference
t-m

{1, 2, 3, 4}
```

merge 2 list

```
list1=[1,2,3,4,5]
list2=[6,7,8,9,10]
merged_list=list1+list2
merged_list

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

dictionary operation

```
student={"name":"johan","age":21,"marks":85}
print("name:",student["name"])
student["marks"]=90
print("updated marks:",student["marks"])

name: johan
updated marks: 90
```

finding the max and min in list

```
num=[10,20,30,40,50]
max_num=max(num)
min_num=min(num)
print("max:",max_num)
print("min:",min_num)

max: 50
min: 10
```

count frequency of element in list

```

numbers=[1,2,2,4,4,4,4]
frequency={}
for num in numbers:
    frequency[num]=frequency.get(num,0)+1
print("frequency of element:",frequency)

frequency of element: {1: 1, 2: 2, 4: 4}

```

palindrom

```

number=int(input("enter a number:"))
reversenum=0
temp=number
while temp>0:
    digit=temp%10
    reversenum=reversenum*10+digit
    temp//=10
if number==reversenum:
    print("palindrome")
else:
    print("not palindrome")

enter a number:676
palindrome

def isPalindrome(self, x):
    """
    :type x: int
    :rtype: bool
    """
    if x < 0 or (x % 10 == 0 and x != 0):
        return False

    reversed_half = 0
    while x > reversed_half:
        reversed_half = reversed_half * 10 + x % 10
        x //= 10

    return x == reversed_half or x == reversed_half // 10

import random
answers = [
    "It is certain.",
    "It is decidedly so.",
    "Without a doubt.",
    "Yes definitely.",
    "You may rely on it.",
    "As I see it, yes.",

```

```
        "Most likely.",
        "Outlook good.",
        "Yes.",
        "Signs point to yes."
    ]
    print("ask the magic 8 ball a question")
    input()
    print(random.choice(answers))
ask the magic 8 ball a question
34
It is certain.
```