Login User:

users=["john","bob","alex","alice","charlie","john","alex","john","alex"]

d4={}

for i in users:

    if i in d4:

        d4[i]+=1

        print(i,d4[i])

    else:

        d4[i]=1

        print(i,d4[i])

print(d4)

$ python login.py

john 1

bob 1

alex 1

alice 1

charlie 1

john 2

alex 2

john 3

alex 3

{'john': 3, 'bob': 1, 'alex': 3, 'alice': 1, 'charlie': 1}

2.

Invert a dictionary:

original={"a":1,"b":2,"c":1,"d":2,"e":3,"f":2}

inverted={}

for x,y in original.items():

    if y not in inverted:

        inverted[y]={x}

    else:

        inverted[y].add(x)

print(inverted)

$ python invert.py

{1: {'c', 'a'}, 2: {'d', 'b', 'f'}, 3: {'e'}}

3.

Isomorphic strings:

Str1=add

Str2=egg

a->e

d->g

d->g

str1=”abccba”

str2=”gmssmg”

a->g

b->m

c->s

c->s

b->m

a->g

str1=apple

str2=table

a->t

p->a

p->b 🡪 Not ismorphic

str1=sky

str2=bee

s->b

k->e

y->e 🡪 Wrong

A screenshot of a computer

AI-generated content may be incorrect.

str1="abccbab"

str2="gmssmgp"

j=0

flag=True

if len(str1) != len(str2):

    flag=False

else:

    map1,map2={},{}

    for c1,c2 in zip(str1,str2):

        if c1 in map1:

            if map1[c1]!=c2:

                flag=False

        else:

            map1[c1]=c2

        if c2 in map2:

            if map[c2]!=c1:

                flag=False

            else:

                map2[c2]=c1

if flag!=False:

    print("ismorphic")

else:

    print("Not morphic")

Case 1:

str1="abccba"

str2="gmssmg"

$ python isomorphic.py

ismorphic

Case 2:

str1="abccbab"

str2="gmssmgp"

$ python isomorphic.py

Not morphic

NESTED LOOP to dictionary:

A screenshot of a computer

AI-generated content may be incorrect.

data=[["james",25,"NY"],["Kiran",30,"DEL"],["Smith",24,"PAR"],["Raj",27,"DEL"]]

header=['name','age','city']

result=[]

length=len(header)

for i in range(length):

    newdict={}

    for row in data:

        if row[i] not in newdict:

            newdict[row[i]]=[row]

        else:

            newdict[row[i]].append(row)

    result.append(newdict)

print("Dict")

for i in range(length):

    print("\n"+header[i])

    for key,value in result[i].items():

        print(f'{key:<10}:{value}')

Output:

Dict

name

james :[['james', 25, 'NY']]

Kiran :[['Kiran', 30, 'DEL']]

Smith :[['Smith', 24, 'PAR']]

Raj :[['Raj', 27, 'DEL']]

age

25 :[['james', 25, 'NY']]

30 :[['Kiran', 30, 'DEL']]

24 :[['Smith', 24, 'PAR']]

27 :[['Raj', 27, 'DEL']]

city

NY :[['james', 25, 'NY']]

DEL :[['Kiran', 30, 'DEL'], ['Raj', 27, 'DEL']]

PAR :[['Smith', 24, 'PAR']]

Dynamic KEY:

import uuid

items=[

    ["laptop",1200],

    ["mouse",20],

    ["keyboard",30],

    ["tablet",200]

]

item\_data={}

for i in items:

    id=uuid.uuid5(uuid.NAMESPACE\_OID,i[0])

    key=id.hex[:6]

    item\_data[key]=i

for k,v in item\_data.items():

    print(f'{k}{v}')

$ python dynamic\_key.py

bd0220['laptop', 1200]

db0304['mouse', 20]

c2f82e['keyboard', 30]

56831a['tablet', 200]