c1=999888777

c2=111222333

p1=9876

p2=1234

c=int(input("Enter card number: "))

p=int(input("Enter pin number: "))

if c==c1 and p==p1:

print("Transaction successfull ")

elif c==c2 and p==p2:

print("transaction successfull ")

else:

print("incorrect card or pin nimber")

**op:**

**Enter card number: 999888777**

**Enter pin number: 9876**

**Transaction successfull**

**========================================================================== RESTART: K:/python training/card.py**

**Enter card number: 999888777**

**Enter pin number: 1234**

**incorrect card or pin nimber**

**program:**

c1=999888777

c2=111222333

p1=9876

p2=1234

c=int(input("Enter card number: "))

p=int(input("Enter pin number: "))

if c==c1 and p==p1:

print("Transaction successfull ")

elif c==c2 and p==p2:

print("transaction successfull ")

else:

print("incorrect card or pin number")

e=input("Forgot pin?(y/n)")

if e=="y":

print("Update your pin")

c=int(input("Enter card number: "))

new=int(input("Enter new pin: "))

if c==c1:

p1=new

print("Pin updated successfully ")

elif c==c2:

p2=new

print("Pin updated successfully ")

else:

print("Card does not exist ")

else:

print("Retry later")

**output:**

**Enter card number: 999888777**

**Enter pin number: 1234**

**incorrect card or pin number**

**Forgot password?(y/n)y**

**Update your password**

**Enter card number: 999888777**

**Enter new password: 1234**

**Pin updated successfully**

c1=999888777

c2=111222333

p1=9876

p2=1234

c=int(input("Enter card number: "))

p=int(input("Enter pin number: "))

if c==c1 and p==p1:

print("Transaction successfull ")

elif c==c2 and p==p2:

print("transaction successfull ")

else:

print("incorrect card or pin number")

e=input("Forgot pin?(y/n)")

if e=="y":

print("Update your pin")

c=int(input("Enter card number: "))

new=int(input("Enter new pin: "))

confirm=int(input("Confirm new pin: "))

while new!=confirm:

print("new pin and confirm pin does'nt match:( re-enter")

new=int(input("Enter new pin: "))

confirm=int(input("Confirm new pin: "))

else:

if c==c1:

p1=new

print("Pin updated successfully ")

elif c==c2:

p2=new

print("Pin updated successfully ")

else:

print("Card does not exist ")

else:

print("Retry later")

**output:**

**Enter card number: 999888777**

**Enter pin number: 1234**

**incorrect card or pin number**

**Forgot pin?(y/n)y**

**Update your pin**

**Enter card number: 999888777**

**Enter new pin: 4562**

**Confirm new pin: 1234**

**new pin and confirm pin does'nt match:( re-enter**

**Enter new pin: 1234**

**Confirm new pin: 1234**

**Pin updated successfully**

msg="Python Programming"

print(msg)

Python Programming

print(msg[1:18])

ython Programming

print(msg[0:8])

Python P

print(msg[0:18])

Python Programming

print(msg[7:18])

Programming

print(msg[5:])

n Programming

print(msg[:6])

Python

string methods:

**lang="python"**

**print(lang.isalpha())**

**True**

**num="12345"**

**print(num.isdigit())**

**True**

**print(lang.isdigit())**

**False**

**print(num.isalpha())**

**False**

**pwd="abc123"**

**print(pwd.isalnum())**

**True**

**Find and replace methods::**

**s="welcome to pyhton programming"**

**print(s.find("python"))**

**-1**

**print(s.find("pyhton"))**

**11**

**print(s.find("pyhton",0,10))**

**-1**

**print(s.replace("p","P"))**

**welcome to Pyhton Programming**

**split method:**

**s="Welcome to python programming"**

**print(s.split())**

**['Welcome', 'to', 'python', 'programming']**

**print(s)**

**Welcome to python programming**

**Join method:**

**print("-".join(s))**

**W-e-l-c-o-m-e- -t-o- - -p-y-t-h-o-n- -p-r-o-g-r-a-m-m-i-n-g**

# swap first and last letter

str=input()

if len(str)>=2:

    ns =str[-1]+ str[1:-1]+str[0]

    print("Modified string :",ns)

else:

    print("string is too short.")

**op:**

**hello world**

**Modified string : dello worlh**

**hi**

**Modified string : ih**

**k**

**string is too short.**

# remove odd index values

s=input()

ns= s[::2]

print("Modified string :",ns)

**op:**

**hello**

**Modified string : hlo**

**# display length until the user enter quit(QUIT)**

**s=input('')**

**while s.upper()!= "QUIT" :**

**print(len(s))**

**s=input()**

**else:**

**print("Exiting the programming....")**

**op:**

**hello**

**5**

**tKE ME CHURCH**

**13**

**hello world**

**11**

**quit**

**Exiting the programming....**

s=''

count=1

while count<=5 :

    s=input()

    if len(s)<6 :

        print("Enter again: ")

    else:

        print(f"{count}:{s}")

        count+=1

**op:**zayedd

1:zayedd

zayed

Enter again:

karthik

2:karthik

hhhhhhhh

3:hhhhhhhh

rdgydyhdy

4:rdgydyhdy

grsdrygsgf

5:grsdrygsgf

**LISTS:**

Sequence of comma separated values/elements/items

Between square brackets.

Concatenation(+)

lst=[10,20,30]

lst1=['a','b','c']

lst2=lst+lst1

print(lst2)

**op:::**

**[10, 20, 30, 'a', 'b', 'c']**

#repetition(\*)

lst=[10,20,30]

lst1=lst\*2

print(lst1)

**op::**

**[10, 20, 30, 10, 20, 30]**

#nested lists

lst=[11,22,33,[44,55,66],77,88]

print(lst)

print(lst[3])

op::

[11, 22, 33, [44, 55, 66], 77, 88]

[44, 55, 66]

**op::**

**[11, 22, 33, [44, 55, 66], 77, 88]**

**[44, 55, 66]**

#append() method

lst=[11,22,33,[44,55,66],77,88]

lst.append(99)

print(lst)

op:

[11, 22, 33, [44, 55, 66], 77, 88, 99]

#count()

k=[1,2,3,4,1,21,5,1,1,33,]

print(k.count(1))

print(k.count(20))

op::

4

0

# insert()

k=[1,2,3,4,1,21,5,1,1,33]

k.insert(20,'kart')

print(k)

op::

[1, 2, 3, 4, 1, 21, 5, 1, 1, 33, 'kart']

# insert()

k=[1,2,3,4,1,21,5,1,1,33]

k.insert(2,'kart')

print(k)

op::

[1, 2, 'kart', 3, 4, 1, 21, 5, 1, 1, 33]

#pop()

k=[1, 2, 'kart', 3, 4, 1, 21, 5, 1, 1, 33]

k.pop()

print(k)

k.pop(3)

print(k)

Op::

[1, 2, 'kart', 3, 4, 1, 21, 5, 1, 1]

[1, 2, 'kart', 4, 1, 21, 5, 1, 1]

##sort()

v=[1,2,3,4,1,21,5,1,1,33]

print(v)

v.sort()

print(v)

op::

[1, 2, 3, 4, 1, 21, 5, 1, 1, 33]

[1, 1, 1, 1, 2, 3, 4, 5, 21, 33]

##extend()

k=[11,22,33]

v=[44,55,66]

k.extend(v)

print(k)

v.extend(k)

print(v)

op::

[11, 22, 33, 44, 55, 66]

[44, 55, 66, 11, 22, 33, 44, 55, 66]

#remove()

k=[11,22,33]

k.remove(33)

print(k)

k.remove(44)

print(k)

op::

[11, 22]

Traceback (most recent call last):

File "c:\Users\karu8\python\lists.py", line 62, in <module>

k.remove(44)

ValueError: list.remove(x): x not in list

#reverse()

k=[11,22,33]

k.reverse()

print(k)

OP::

[33, 22, 11]

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

n=int(input("Enter no of places to shift :"))

l=l[-n:]+l[:-n]

print(l)

l=l[n:]+l[:n]

print(l)

**op::** **[3, 4, 5, 1, 2]**

**[1, 2, 3, 4, 5]**

**Tuple:**

# tpl=(11,22,33,44,55)

# # for i in tpl:

# #     print(i,end=" ")

# # tpl=(11,22,33,44,55)

# # index=0

# # while index<len(tpl):

# #     print(tpl[index],end=' ')

# #     index+=1

# tpl=(11,22,33,44,55)

# for i in range(len(tpl)):

#     print(tpl[i],end=' ')

**Op:**

**11 22 33 44 55**

#immutable

k=11,22,33,44,55

print(k)

print(k[0])

k[0]=90

op::

(11, 22, 33, 44, 55)

11

Traceback (most recent call last):

File "c:\Users\karu8\python\tuples.py", line 20, in <module>

k[0]=90

s=([1,2,3,4],[11,22],'python')

s[0][0]=10

del(s[0][2])

print(s[0][0])

print(s)

**op:** **10**

**([10, 2, 4], [11, 22], 'python')**

**Adding elements into tuple:**

tpl=()

n=int(input("Enter no.of elements in tuple :"))

for i in range(n):

    ele=int(input())

    tpl=tpl+(ele,)

print(tpl)

**OP::**

**Enter no.of elements in tuple :5**

**10**

**20**

**30**

**40**

**50**

**(10, 20, 30, 40, 50)**