1.To print Hello world

if \_\_name\_\_ == '\_\_main\_\_':

print("Hello, World!")

2.Arithmetic Operators

f \_\_name\_\_ == '\_\_main\_\_':

a = int(input())

b = int(input())

c=a+b

d=a-b

e=a\*b

print(c)

print(d)

print(e)

3.Division

f \_\_name\_\_ == '\_\_main\_\_':

a = int(input())

b = int(input())

print(a//b)

print(a/b)

4.Loops

if \_\_name\_\_ == '\_\_main\_\_':

n = int(input())

for i in range(n):

print(i\*\*2)

5.Pint functions

if \_\_name\_\_ == '\_\_main\_\_':

n = int(input())

for i in range(1,n+1):

print(i,end='')

6.Nested Lists

n = int(input())

students = []

for \_ in range(n):

    name = input()

    score = float(input())

    students.append([name, score])

unique\_grades = sorted({score for name, score in students})

second\_lowest = unique\_grades[1]

second\_lowest\_students = [name for name, score in students if score == second\_lowest]

for name in sorted(second\_lowest\_students):

    print(name)

7.Finding the percentage

n = int(input())

student\_marks = {}

for \_ in range(n):

    line = input().split()

    name = line[0]

    scores = list(map(float, line[1:]))

    student\_marks[name] = scores

query\_name = input()

average = sum(student\_marks[query\_name]) / len(student\_marks[query\_name])

print(f"{average:.2f}")

8. Lists

lst = []

N = int(input())

for \_ in range(N):

    cmd = input().strip().split()

    if cmd[0] == "insert":

        lst.insert(int(cmd[1]), int(cmd[2]))

    elif cmd[0] == "print":

        print(lst)

    elif cmd[0] == "remove":

        lst.remove(int(cmd[1]))

    elif cmd[0] == "append":

        lst.append(int(cmd[1]))

    elif cmd[0] == "sort":

        lst.sort()

    elif cmd[0] == "pop":

        lst.pop()

    elif cmd[0] == "reverse":

        lst.reverse()

9.Tuples

n = int(input())

integer\_list = map(int, input().split())

t = tuple(integer\_list)

print(hash(t))

10. Swap case

def swap\_case(s):

    return s.swapcase()

if \_\_name\_\_ == '\_\_main\_\_':

    s = input()

    result = swap\_case(s)

    print(result)