

```
In [1]: """Q.1. Create two int type variables, apply addition, subtraction, division and multiplications
and store the results in variables. Then print the data in the following format by calling the
variables:
First variable is __ & second variable is __.
Addition: __ + __ = __
Subtraction: __ - __ = __
Multiplication: __ * __ = __
Division: __ / __ =
"""
```

Cell In[1], line 1  
Q.1. Create two int type variables, apply addition, subtraction, division and multiplications  
^

SyntaxError: invalid syntax

```
In [13]: a=int(input("enter the first no. "))
b=int(input("enter the second no. "))
print("first variable & second variable is:" ,a,b)
c=a+b
print("Addition :",c)
d=a-b
print("Subtract:",d)
e=a*b
print("multiplication",e)
if b!=0:
    f=a/b
    print("devsion",f)
```

enter the first no.5  
enter the second no4  
first variable & second variable is: 5 4  
Addition : 9  
Subtract: 1  
multiplication 20  
devsion 1.25

```
In [ ]: """Q.2. What is the difference between the following operators:
(i) '/' & '//'
(ii) '*' & '^'"""
```

```
In [15]: ANSWER: '/' operator with integers, it performs floating-point division, and the result is a floating-point number.
Example: 5 / 2 would result in 2.5.
AND '/' operator with integers, it performs division and rounds down to the nearest integer.
Example: 5 // 2 would result in 2.
```

Out[15]: "ANSWER: '/' operator with integers, it performs floating-point division, and the result is a floating-point number.\nExample: 5 / 2 would result in 2.5.\nAND '/' operator with integers, it performs division and rounds down to the nearest integer.\nExample: 5 // 2 would result in 2."

```
In [ ]: # Q.3. List the logical operators.
answer:AND Operator (and):
Returns True if both operands are true.
Example: True and False evaluates to False.

OR Operator (or):
Returns True if at least one of the operands is true.
Example: True or False evaluates to True.

NOT Operator (not):
Returns the opposite boolean value of the operand.
Example: not True evaluates to False
```

```
In [ ]: Q.4. Explain right shift operator and left shift operator with examples.
answer:Right Shift Operator (>>):
The right shift operator shifts the bits of a binary number to the right by a specified number of positions.

ex: The binary representation of 16 is 10000. When we right shift by 2 positions, we get 0001, which is equivalent to the decimal number 4.

Left Shift Operator (<<):
The left shift operator shifts the bits of a binary number to the left by a specified number of positions.

Ex: The binary representation of 8 is 1000. When we left shift by 3 positions, we get 1000000, which is equivalent to the decimal number 64
```

```
In [ ]: Q.5. Create a list containing int type data of length 15. Then write a code to check if 10 is
present in the list or no
```

```
In [39]: import random

# Generate a list of 15 random numbers
random_numbers = [random.randint(1, 100) for _ in range(15)]

# Print the list
print(random_numbers)
for i in random_numbers:
    if i==10:
        print("yes")

[49, 10, 16, 31, 90, 64, 10, 72, 5, 86, 37, 26, 8, 73, 84]
yes
yes
```

In [ ]:

In [ ]: