

```

#variables
name = "Aishu"
rollno = 3
is_student = True
print(name)
print(rollno)
print(is_student)

Aishu
3
True

#datatypes
num = 10
pi=3.14
message="Hello World"
is_value=True
a = 1j
b = ["a", "b", "c"]
c = ("apple", "banana", "cherry")
d = range(5)
e = {"name" : "Aishu", "rollno" : 3}
f = {"abi", "aishu", "anita"}
g = frozenset({"apple", "banana", "cherry"})
h = b"Hello"
i = bytearray(5)
j = memoryview(bytes(5))
k = None
print(type(num))
print(type(pi))
print(type(message))
print(type(is_value))
print(type(a))
print(type(b))
print(type(c))
print(type(d))
print(type(e))
print(type(f))
print(type(g))
print(type(h))
print(type(i))
print(type(j))
print(type(k))

<class 'int'>
<class 'float'>
<class 'str'>
<class 'bool'>
<class 'complex'>
<class 'list'>

```

```
<class 'tuple'>
<class 'range'>
<class 'dict'>
<class 'set'>
<class 'frozenset'>
<class 'bytes'>
<class 'bytearray'>
<class 'memoryview'>
<class 'NoneType'>
```

```
#for loop
```

```
names = ["Abi", "Aishu", "Anita"]
for x in names:
    print(x)
```

```
Abi
Aishu
Anita
```

```
#while loop
```

```
i = 1
while i < 6:
    print(i)
    i += 1
```

```
1
2
3
4
5
```

```
#functions
```

```
def tri_recursion(k):
    if(k > 0):
        result = k + tri_recursion(k - 1)
        print(result)
    else:
        result = 0
    return result
print("\nRecursion Example Results")
tri_recursion(5)
```

```
Recursion Example Results
```

```
1
3
6
10
15
15
```

```
#arithmetic operators
```

```
num1 = int(input("Enter the First Value: "))
num2 = int(input("Enter the Second Value: "))
add = num1 + num2
sub = num1 - num2
multi = num1 * num2
div = num1 / num2
mod = num1 % num2
expo = num1 ** num2
print(f"The Sum of {num1} and {num2} = {add}")
print(f"The Subtraction of {num2} from {num1} = {sub}")
print(f"The Multiplication of {num1} and {num2} = {multi}")
print(f"The Division of {num1} and {num2} = {div}")
print(f"The Modulus of {num1} and {num2} = {mod}")
print(f"The Exponent Value of {num1} and {num2} = {expo}")
```

```
Enter the First Value: 10
```

```
Enter the Second Value: 5
```

```
The Sum of 10 and 5 = 15
```

```
The Subtraction of 5 from 10 = 5
```

```
The Multiplication of 10 and 5 = 50
```

```
The Division of 10 and 5 = 2.0
```

```
The Modulus of 10 and 5 = 0
```

```
The Exponent Value of 10 and 5 = 100000
```

```
#list
```

```
my_list = [1, 2, 3, 4, 5]
```

```
my_list[1] = 20
```

```
print("List after updating the second element:", my_list)
```

```
my_list.append(6)
```

```
print("List after appending a new element:", my_list)
```

```
my_list.insert(2, 30)
```

```
print("List after inserting a new element at index 2:", my_list)
```

```
my_list.remove(4)
```

```
print("List after removing the element with value 4:", my_list)
```

```
del my_list[3]
```

```
print("List after deleting the element at index 3:", my_list)
```

```
popped_element = my_list.pop()
```

```
print("List after popping the last element:", my_list)
```

```
print("Popped element:", popped_element)
```

```
print("List after clearing all elements:", my_list)
```

```
List after updating the second element: [1, 20, 3, 4, 5]
```

```
List after appending a new element: [1, 20, 3, 4, 5, 6]
```

```
List after inserting a new element at index 2: [1, 20, 30, 3, 4, 5, 6]
```

```
List after removing the element with value 4: [1, 20, 30, 3, 5, 6]
```

```
List after deleting the element at index 3: [1, 20, 30, 5, 6]
```

```
List after popping the last element: [1, 20, 30, 5]
```

Popped element: 6
List after clearing all elements: [1, 20, 30, 5]

#tuple

```
fruits = ("apple", "banana", "cherry")
print(fruits[0])
print(fruits[1])
print(len(fruits))
mixed_tuple = (2, "Hello", "Python", 2)
print(mixed_tuple)
tuple1 = ("abc", 34, True, 40, "male")
print(tuple1)
```

```
apple
banana
3
(2, 'Hello', 'Python', 2)
('abc', 34, True, 40, 'male')
```

#dictionary

```
MLB_team = {
    'Colorado': 'Rockies',
    'Boston': 'Red Sox',
    'Minnesota': 'Twins',
    'Milwaukee': 'Brewers',
    'Seattle': 'Mariners'
}
print(f"MLB Teams: {MLB_team}")
print(f"Team in Boston: {MLB_team.get('Boston', 'Not found')}")
MLB_team['New York'] = 'Yankees'
print(f"Updated MLB Teams: {MLB_team}")
del MLB_team['Milwaukee']
print(f"Updated MLB Teams (after removal): {MLB_team}")
```

```
MLB Teams: {'Colorado': 'Rockies', 'Boston': 'Red Sox', 'Minnesota': 'Twins', 'Milwaukee': 'Brewers', 'Seattle': 'Mariners'}
Team in Boston: Red Sox
Updated MLB Teams: {'Colorado': 'Rockies', 'Boston': 'Red Sox', 'Minnesota': 'Twins', 'Milwaukee': 'Brewers', 'Seattle': 'Mariners', 'New York': 'Yankees'}
Updated MLB Teams (after removal): {'Colorado': 'Rockies', 'Boston': 'Red Sox', 'Minnesota': 'Twins', 'Seattle': 'Mariners', 'New York': 'Yankees'}
```

#String manipulation

```
word = "Computer Science"
print(f"Original string: {word}")
print(f"First letter: {word[0]}")
print(f"Length of the string: {len(word)}")
print(f"Substring: {word[9:16]}")
```

```
str1 = "Computer"
str2 = "Engineering"
combined = str1 + " " + str2
print(f"Combined string: {combined}")
```

Original string: Computer Science
First letter: C
Length of the string: 16
Substring: Science
Combined string: Computer Engineering

```
#if statement
```

```
x = 10
if x > 5:
    print("x is greater than 5")
```

x is greater than 5

```
#if else statement
```

```
x = 3
if x == 4:
    print("Yes")
else:
    print("No")
```

No

```
#nested if else statement
```

```
letter = "A"
if letter == "B":
    print("Letter is B")
else:
    if letter == "C":
        print("Letter is C")
    else:
        print("Letter is A")
```

Letter is A

```
#if elif else statement
```

```
letter = "A"
if letter == "B":
    print("Letter is B")
elif letter == "C":
    print("Letter is C")
elif letter == "A":
    print("Letter is A")
else:
    print("Letter isn't A, B, or C")
```

Letter is A

```
#ternary  
result = "Yes" if x == 4 else "No"  
print(result)
```

No