

```
# Data Types
integerNum = 10
floatNum = 3.14
stringMsg = "Hello, Python!"
booleanVal = True
listItems = [1, 2, 3, 4, 5]
tupleItems = (10.0, 20.0)
dictItems = {"name": "Alice", "age": 30}
setItems = {1, 2, 3, 4, 5}
```

```
# Printing values and their types
print(f"Value: {integerNum} -> Type: {type(integerNum)}")
print(f"Value: {floatNum} -> Type: {type(floatNum)}")
print(f"Value: {stringMsg} -> Type: {type(stringMsg)}")
print(f"Value: {booleanVal} -> Type: {type(booleanVal)}")
print(f"Value: {listItems} -> Type: {type(listItems)}")
print(f"Value: {tupleItems} -> Type: {type(tupleItems)}")
print(f"Value: {dictItems} -> Type: {type(dictItems)}")
print(f"Value: {setItems} -> Type: {type(setItems)}")
```

```
PS C:\Users\msdak> python -u "c:\Users\msdak\OneDrive\Desktop\
Value: 10 -> Type: <class 'int'>
Value: 3.14 -> Type: <class 'float'>
Value: Hello, Python! -> Type: <class 'str'>
Value: True -> Type: <class 'bool'>
Value: [1, 2, 3, 4, 5] -> Type: <class 'list'>
Value: (10.0, 20.0) -> Type: <class 'tuple'>
Value: {'name': 'Alice', 'age': 30} -> Type: <class 'dict'>
Value: {1, 2, 3, 4, 5} -> Type: <class 'set'>
```

```
from tabulate import tabulate
a, b = map(int, input().split())
```

```
results = [
    ["Operation", "Result"],
    [f"{a} + {b}", a + b],
    [f"{a} - {b}", a - b],
    [f"{a} * {b}", a * b],
    [f"{a} / {b}", a / b],
    [f"{a} // {b}", a // b],
]
```

```
print(tabulate(results,
    headers="firstrow",
    tablefmt="psql"))
```

```
PS C:\Users\msdak> python -u
5 2
+-----+-----+
| Operation | Result |
+-----+-----+
| 5 + 2     | 7      |
| 5 - 2     | 3      |
| 5 * 2     | 10     |
| 5 / 2     | 2.5    |
| 5 // 2    | 2      |
+-----+-----+
```

```
age = int(input("AGE: "))
# Conditional statements
if age >= 18:
    print("You can vote")
else:
    print("You cannot vote.")

# multiple conditions
grade = int(input("GRADE: "))
if grade >= 90:
    print("You got an A!")
elif grade >= 80:
    print("You got a B!")
elif grade >= 70:
    print("You got a C!")
elif grade >= 60:
    print("You got a D!")
else:
    print("You got an F.")
```

```
PS C:\Users\msdak>
AGE: 10
You cannot vote.
AGE: 18
You can vote
AGE: 22
You can vote
```

```
PS C:\Users\msdak>
GRADE: 65
You got a D!
GRADE: 80
You got a B!
GRADE: 98
You got an A!
GRADE: 10
You got an F.
```

```
# multiplication table for n
n=5
for i in range(1,11):
    print(f"{n} X {i} = {n*i}")

# 1 to 10 without 5
i=1
while i<11:
    if i!=5:
        print(i,end=' ')
    i+=1

#skip odd and break at 8
for i in range(11):
    if i%2==1:
        continue
    if i==8:
        break
    print(i,end=" ")
```

```
PS C:\Users\msdak> py
PROG 1
5 X 1 = 5
5 X 2 = 10
5 X 3 = 15
5 X 4 = 20
5 X 5 = 25
5 X 6 = 30
5 X 7 = 35
5 X 8 = 40
5 X 9 = 45
5 X 10 = 50
PROG 2
1 2 3 4 6 7 8 9 10
PROG 3
0 2 4 6
PS C:\Users\msdak>
```

```
#function
def greet():
    print("Inside a function")

#parameter and return type
def compound_interest(principal, rate, time):
    Amount = principal * (pow((1 + rate / 100), time))
    CI = Amount - principal
    return CI

#LAMBDA
import math
area = lambda r : math.pi*r*r
print(f"AREA: {area(int(input('RADIUS: ')))}")

#RECURSION
def factorial(n):
    if n<1:
        return 1
    return n*factorial(n-1);
print(f"Factorial: {factorial(int(input('N: ')))}")
```

```
PROG 1
Inside a function

PROG 2
P: 3000, R: 3, T: 5
CI: 477.82222290000027
P: 2000, R: 1.5, T: 5
CI: 154.56800776874888
P: 1500, R: 2, T: 5
CI: 156.1212048
```

```
PROG 3
RADIUS: 3
AREA: 28.274333882308138
RADIUS: 1
AREA: 3.141592653589793
RADIUS: 2
AREA: 12.566370614359172

PROG 4
N: 5
Factorial: 120
N: 6
Factorial: 720
N: 3
Factorial: 6
```