

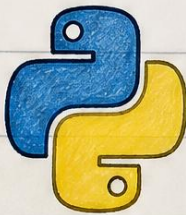
S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

Assignment Statement and Expressions

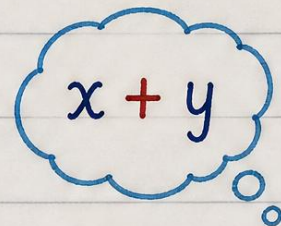
- Assignment statement का उपयोग किसी value को variable में store करने के लिए किया जाता है।
'=' assignment operator है।
- Expression variables, values और operators का ऐसा combination है जो एक result देता है।



Python प्रोग्राम



मान की स्टोरेज



Expression

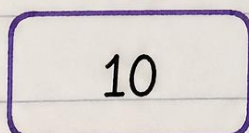
Code

```
x = 10  
y = 5  
result = x + y  
print(result)
```

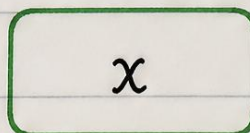
Output

15

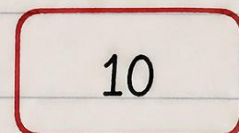
Value



Variable



Result



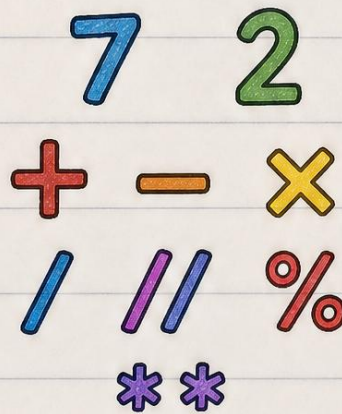
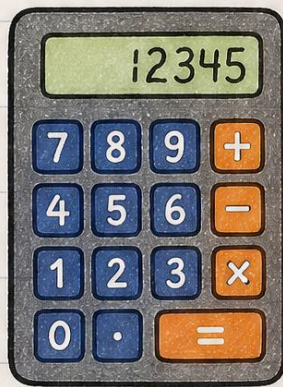
S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

Arithmetic Operators

- Arithmetic operators mathematical calculations के लिए उपयोग होते हैं। '+' addition, '-' subtraction, '*' multiplication, '/' division, '//' floor division, '%' remainder और '**' exponentiation के लिए होते हैं।



$$7 + 2 = 9$$

$$7 - 2 = 5$$

$$7 * 2 = 14$$

$$7 / 2 = 3.5$$

$$7 // 2 = 3$$

$$7 \% 2 = 1$$

$$2 ** 3 = 8$$

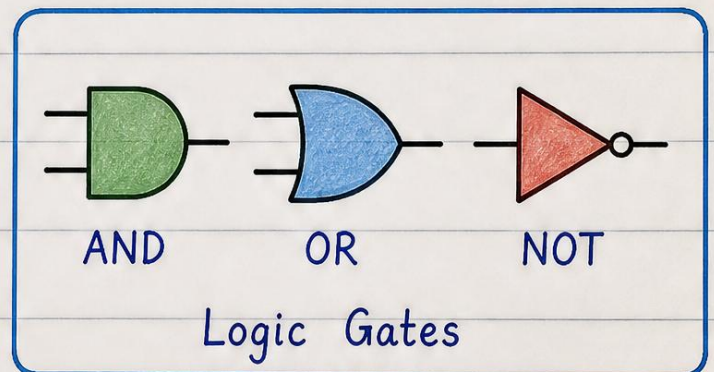
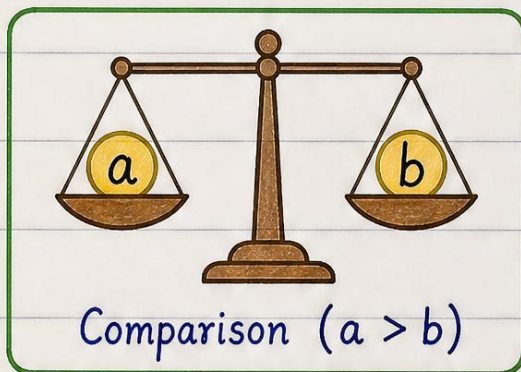
S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

Relational and Logical Operators

- Relational operators दो values की तुलना करते हैं और True या False देते हैं। Operators: $==$, $!=$, $>$, $<$, $>=$, $<=$.
- Logical operators conditions को जोड़ते हैं :
and, or, not.



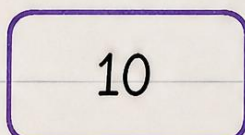
Code

```
a = 10  
b = 5  
print(a > b)  
print(a == b)  
print(a > 5 and b < 10)
```

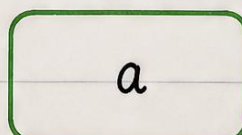
Output

True
False
True

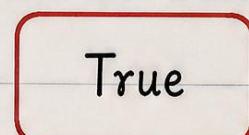
Value



Variable



Result



S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

Bitwise Operators

- Bitwise operators numbers के binary bits पर काम करते हैं। '&' AND, '|' OR, '^' XOR, '~' NOT, '<<' left shift और '>>' right shift हैं।

1010
0110
1101

- $5 \& 3 = 1$
- $5 | 3 = 7$
- $5 \wedge 3 = 6$
- $5 \ll 1 = 10$
- $5 \gg 1 = 2$

1010

101

Binary

→

001

Result

011

Binary

→

111

Result

&

AND

|

OR

^

XOR

~

NOT

<<

Left Shift

>>

Right Shift

S.P Group of Institute

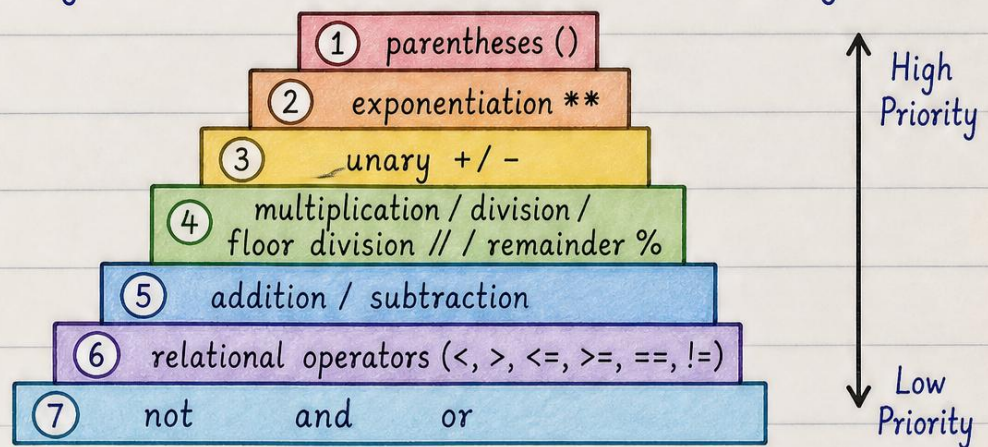
Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

Operator Precedence

- जब expression में कई operators हों, तो Python एक निश्चित क्रम में calculation करता है। सामान्य क्रम: parentheses (), exponentiation **, unary +/ -, multiplication/division/floor division/remainder, addition/subtraction, relational operators, not, and, or.
- समान priority पर evaluation सामान्यतः left-to-right होता है।

()



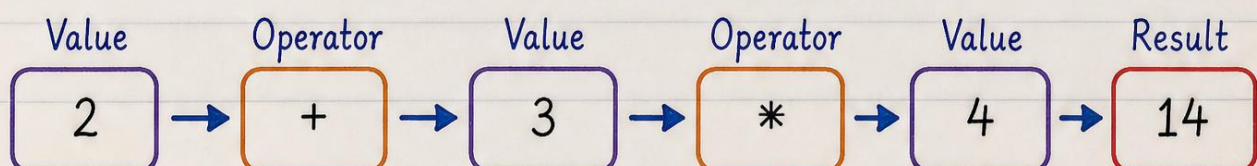
Code

```
result = 2 + 3 * 4
print(result)
result2 = (2 + 3) * 4
print(result2)
```

Output

14

20



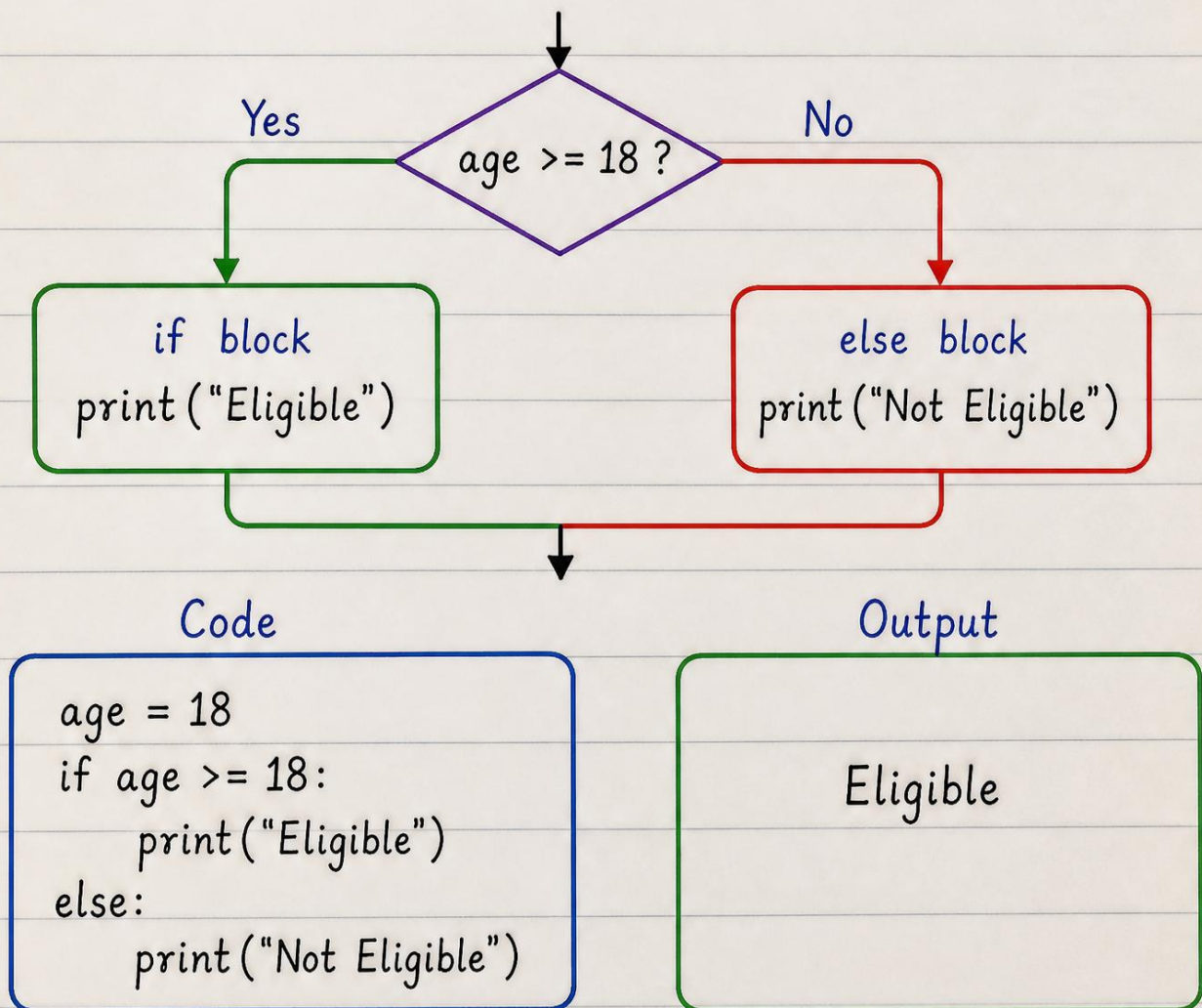
S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

Conditional Statements: if and if-else

- Conditional statement condition के आधार पर decision लेने के लिए उपयोग होता है। if में condition True होने पर block execute होता है। if-else में True के लिए if block और False के लिए else block चलता है।



S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

if-elif-else and Simple Program

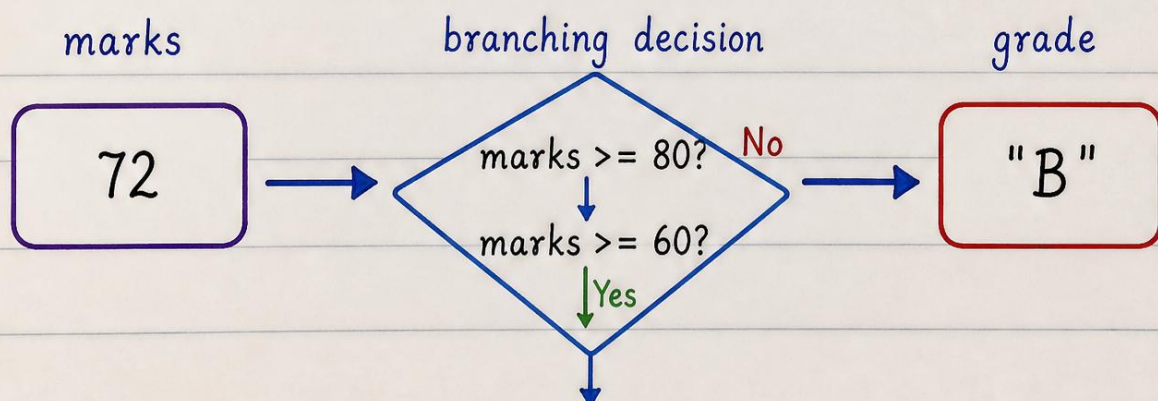
- कई conditions में if-elif-else का उपयोग किया जाता है। Python पहले if condition check करता है, फिर क्रम से elif conditions check करता है; कोई condition True न हो तो else चलता है।

Code

```
marks = 72
if marks >= 80:
    grade = "A"
elif marks >= 60:
    grade = "B"
else:
    grade = "C"
print(grade)
```

Output

B



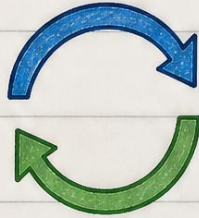
S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

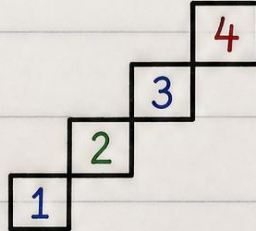
Unit 4: Operators, Expressions and Python Statements

Iterative Computation, Control Flow and range()

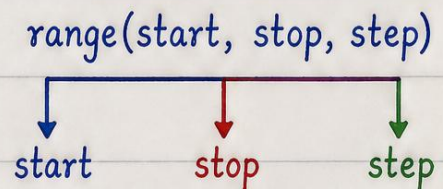
- किसी task को बार-बार दोहराकर result प्राप्त करना iterative computation कहलाता है।
- Program में statements के चलने के क्रम को control flow कहते हैं।
- range() क्रमिक numbers का sequence बनाता है।
- range(start, stop, step) में stop शामिल नहीं होता।



बार-बार दोहराना



क्रमिक numbers

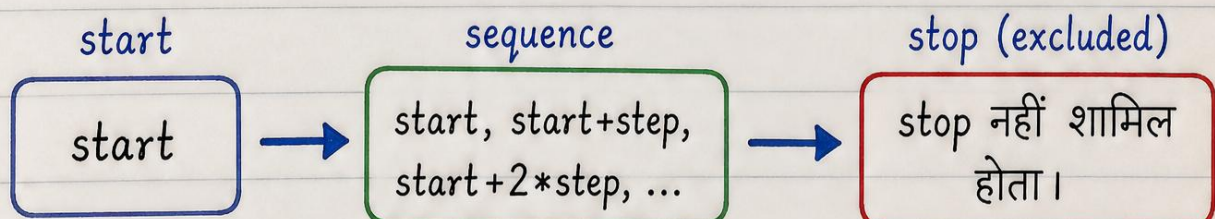


Code

```
for i in range(1, 5):  
    print(i)
```

Output

```
1  
2  
3  
4
```



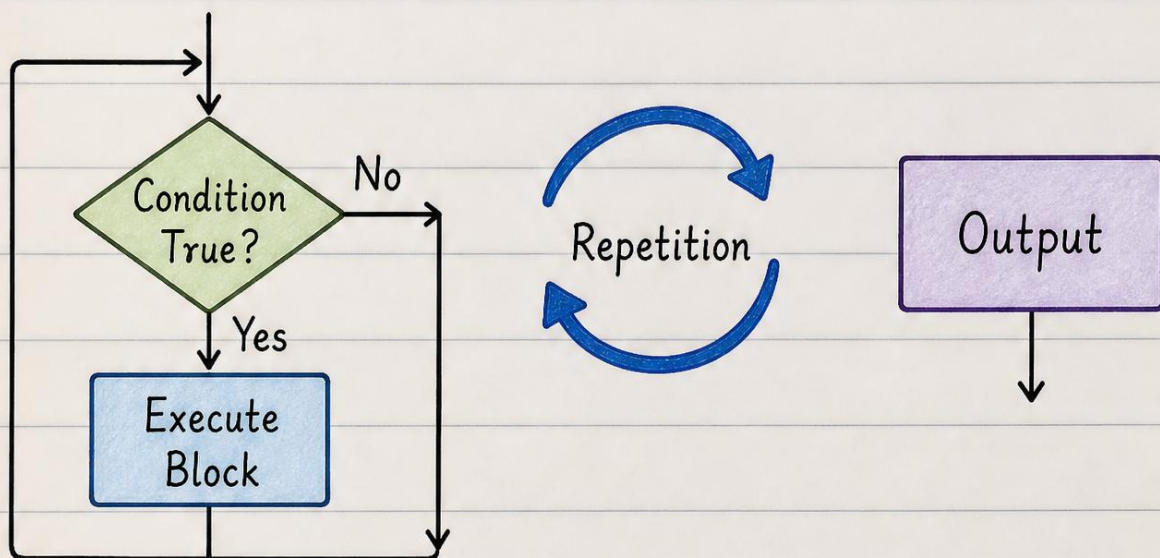
S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

while Statement

- while statement तब तक block को repeat करता है जब तक condition True रहती है। Repetition रोकने के लिए condition को eventually False होना चाहिए।



Code

```
count = 1
while count <= 3:
    print(count)
    count = count + 1
```

Output

```
1
2
3
```

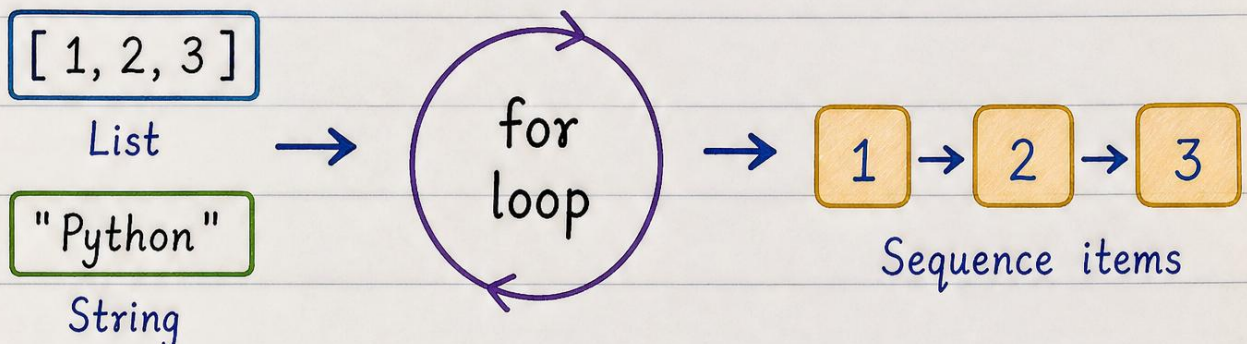

S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

for loop

- for loop किसी sequence के प्रत्येक item पर एक-एक करके कार्य करता है। यह lists, strings और range() के साथ बहुत उपयोगी है।



Code

```
for number in range(1, 4):  
    print("Number =", number)
```

Output

```
Number = 1  
Number = 2  
Number = 3
```


S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

break and continue

- break statement loop को तुरंत समाप्त करता है।
continue statement current iteration को skip करके अगली iteration पर ले जाता है।



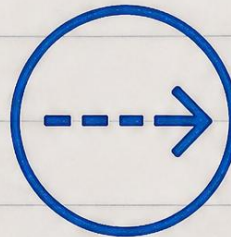
break

Code

```
for i in range(1, 6):  
    if i == 3:  
        break  
    print(i)
```

Output

1
2



continue

Code

```
for i in range(1, 6):  
    if i == 3:  
        continue  
    print(i)
```

Output

1
2
4
5

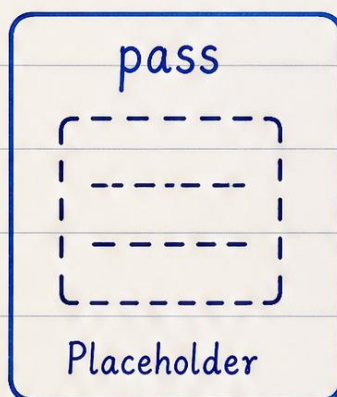
S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 4: Operators, Expressions and Python Statements

pass, else and assert

`pass` एक empty statement है; इसका उपयोग placeholder के रूप में किया जाता है। Loop का `else` तब चलता है जब loop break के बिना पूरा हो जाए। `assert` किसी condition को test करता है; condition False होने पर `AssertionError` मिलता है।

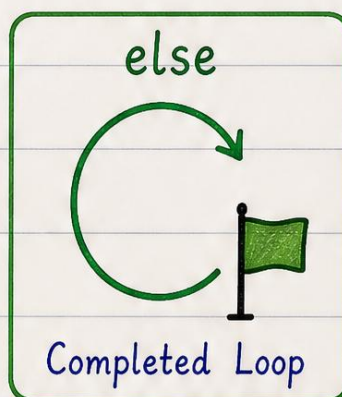


Code

```
for i in range(3):  
    pass  
print("Done")
```

Code

```
x = 10  
assert x > 0  
print("Valid")
```



Output

Done

Output

Valid