

ITD62-123 COMPUTER PROGRAMMING

IMI62-122 FUNDAMENTAL OF COMPUTER PROGRAMMING

Theerat Saichoo & Yunyong Punsawad
School of Informatics, Walailak University



Chapter 2

Condition Statements

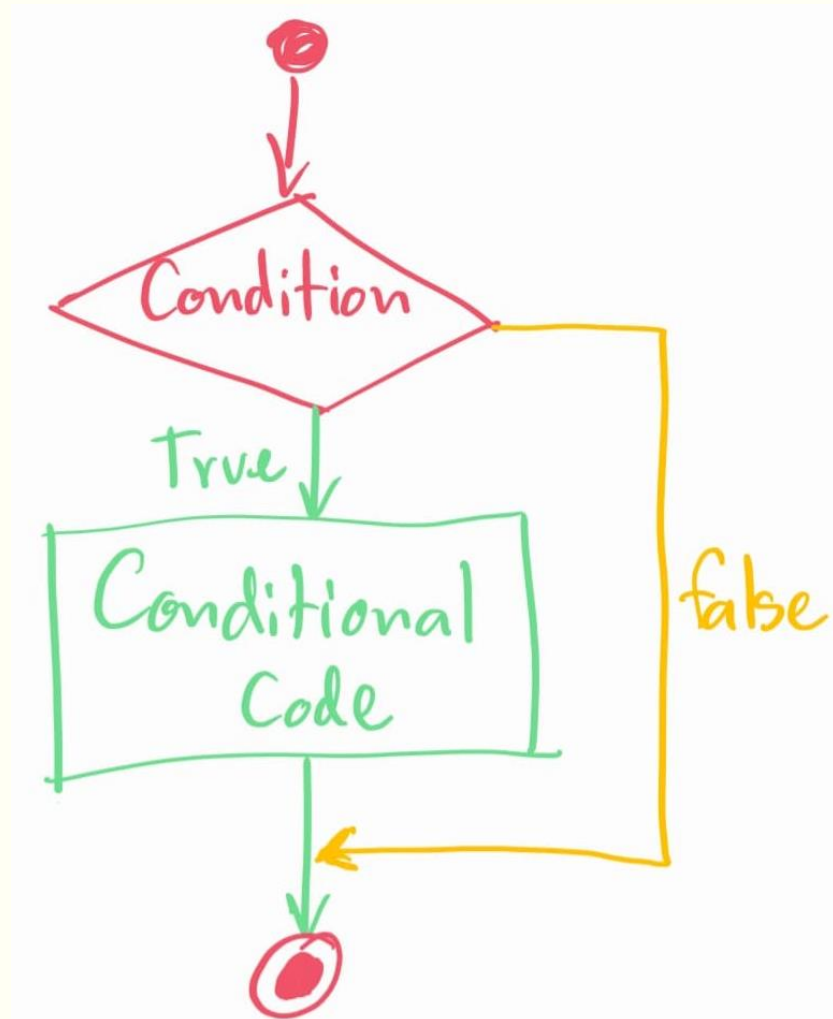


Topics

- Decision making
- Python Conditions
- If statement
- If... else statement
- If... elif statement
- Nested if statements

Decision making

- Decision-making structures require that the programmer specifies one or more conditions to be evaluated or tested by the program



Python Conditions

- Python supports the usual logical conditions from mathematics:
 - Equals: `a == b`
 - Not Equals: `a != b`
 - Less than: `a < b`
 - Less than or equal to: `a <= b`
 - Greater than: `a > b`
 - Greater than or equal to: `a >= b`
- These conditions can be used in several ways, most commonly in "if statements" and loops.
- An "if statement" is written by using the if keyword.

if statements

- An **if** statement consists of a Boolean expression followed by one or more statements.
- **Syntax**

```
if boolean_expression :  
    /* statement(s) will execute if the boolean expression is true */
```

if statements – example code

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



If... else statements

- An if statement can be followed by an optional else statement, which executes when the Boolean expression is false.
- **Syntax**

```
if boolean_expression :  
    /* statement(s) will execute if the boolean expression is true */  
else:  
    /* statement(s) will execute if the boolean expression is false */
```


If...else statements – example code

```
a = 200  
b = 33  
  
if b > a:  
    print("b is greater than a")  
else:  
    print("a is greater than b")
```



If...elif statements

- The **elif** keyword is python's way of saying "if the previous conditions were not true, then try this condition".
- **Syntax**

```
if boolean_expression :  
    /* statement(s) will execute if the boolean expression 1 is true */  
elif boolean_expression :  
    /* statement(s) will execute if the boolean expression 2 is true */  
elif boolean_expression :  
    /* statement(s) will execute if the boolean expression 3 is true */  
else:  
    /* executes when the none of the above condition is true */
```

If...elif statements – example code

```
a = 10
b = 10
if a > b :
    print("a is greater than b")
elif a==b:
    print("a and b are equal")
elif a < b:
    print("b is greater than a")
else:
    print("nothing a and b")
```



Nested if statements

- It is always legal in Python to nest if-else statements, which means you can use one if or else if statement inside another if or else if statement(s).
- **Syntax**

```
if boolean_expression :  
    /* statement(s) will execute if the boolean expression is true */  
    if boolean_expression :  
        /* statement(s) will execute if the boolean expression is true */  
    else:  
        /* statement(s) will execute if the boolean expression is false */  
else:  
    /* statement(s) will execute if the boolean expression is false */
```

Nested if statements – example code

```
x = 41

if x > 10:
    print("Above 10,")
    if x > 30:
        print("and also above 30!")
    else:
        print("but not above 30.")
else:
    print("equal or less than 10,")
```



THE END

Class activity 3

1. จงเขียนโปรแกรมรับเลข 2 จำนวนจากผู้ใช้ แล้วนำมาบวกกัน ถ้าผลบวกนั้นมากกว่า 50 แสดง คำว่า " **Greater than 50** " อีกหนึ่งบรรทัด แสดงผลโดยใช้รูปแบบตามตัวอย่าง

ครั้งที่ 1 ทดสอบด้วยการป้อน ตัวเลข 5 และ 5
ตามลำดับ จะได้ผลลัพธ์ดังนี้

Enter Number : 5

Enter Number : 5

$5 + 5 = 10$

ครั้งที่ 2 ทดสอบด้วยการป้อน ตัวเลข 25 และ 45
ตามลำดับ จะได้ผลลัพธ์ดังนี้

Enter Number : 25

Enter Number : 45

$25 + 45 = 70$

Greater than 50

Class activity 3

2. จงเขียนโปรแกรมรับตัวเลขจำนวนเงินฝากธนาคารเป็นจำนวนเต็ม 1 จำนวน แล้วนำไปคิดดอกเบี้ยและแสดงผลจำนวนดอกเบี้ย ตามเงื่อนไขต่อไปนี้

- เงินฝาก 100,001 บาทขึ้นไป อัตราดอกเบี้ยร้อยละ 5
- เงินฝาก 50,001 - 100,000 บาท อัตราดอกเบี้ยร้อยละ 4
- เงินฝาก 10,001 - 50,000 บาท อัตราดอกเบี้ยร้อยละ 3
- เงินฝาก 1 - 10,000 บาท อัตราดอกเบี้ยร้อยละ 2

Class activity 3

ครั้งที่ 1 ทดสอบด้วยการป้อน เงินฝาก จะได้ผลลัพธ์ดังนี้ ครั้งที่ 3 ทดสอบด้วยการป้อน เงินฝาก จะได้ผลลัพธ์ดังนี้

Enter deposit : 1000

Interest = 20

Enter deposit : 70000

Interest = 2800

ครั้งที่ 2 ทดสอบด้วยการป้อน เงินฝาก จะได้ผลลัพธ์ดังนี้ ครั้งที่ 4 ทดสอบด้วยการป้อน เงินฝาก จะได้ผลลัพธ์ดังนี้

Enter deposit : 20000

Interest = 600

Enter deposit : 150000

Interest = 7500