Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

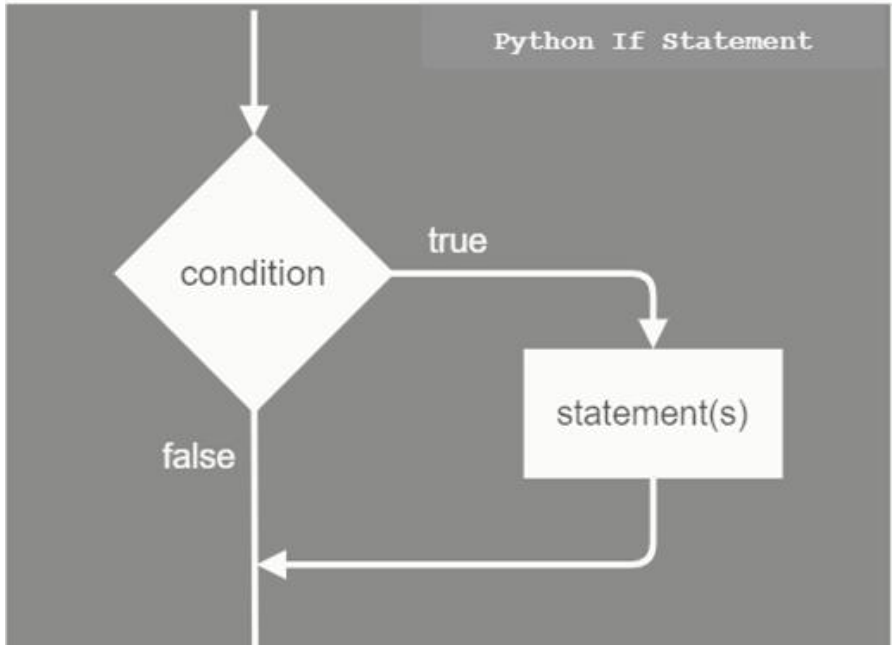
ACCA5036
Scripting and Programming Concepts

Lecture 3:
Control Structures-
Conditionals

Trawsnewid Addysg; Trawsnewid Bywydau
Transforming Education; Transforming Lives

www.uwtsd.ac.uk

Lesson Objectives



- Able to program with decision making using different conditional structure : *If… ; if…else ;if…elif…else*
- Execute different code fragments based on whether a condition is true or false

slido

Please download and install the Slido app on all computers you use

What is the submission date and time for your assessment?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

Join at slido.com #2903058

ⓘ Start presenting to display the joining instructions on this slide.

slido

Please download and install the Slido app on all computers you use

Which function is used to display output in Python?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What is the data type of 3.14 in Python?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

Which function is used to take user input in Python?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

Which of the following is a string in Python?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

How do you convert a string to an integer in Python?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

Which data type is used to store text in Python?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What is the output of print(type(3.14))?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

Which error occurs when you try to use a variable that hasn't been defined?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What will be the output of the following code snippet?

```
print('Hello' + 5)
```

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What will be the output of the following code snippet?

```
print(2 ** '3')
```

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What is the error in this code snippet?

```
print('Hello World)
```

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What will be the output of the following code snippet?

```
x = 10
print(x/0)
```

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What is the output of print(2 ** 3)?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What is the output of print('Hello' + 'World')?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What is the output of print(10 // 3)?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What is the output of print(len('Python'))?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

What is the output of print(bool(0))?

ⓘ Start presenting to display the poll results on this slide.

Activity

Create a program that **generates a username** based on a user's **first name, last name, and birth year** using **string slicing**.

Steps to Follow:

1.Ask the user to input their **first name**, **last name**, and **birth year**.

2.Generate a **username** using the following format:

1. First **three letters** of the first name.
2. First **three letters** of the last name.
3. Last **two digits** of the birth year.

3.Display the generated username.

```
Enter your first name: John
Enter your last name: Doe
Enter your birth year (YYYY): 1995
Your generated username is: JohDoe95
```

Activity - Solution

Create a program that **generates a username** based on a user's **first name, last name, and birth year** using **string slicing**.

Steps to Follow:

1.Ask the user to input their **first name**, **last name**, and **birth year**.

2.Generate a **username** using the following format:

1. First **three letters** of the first name.
2. First **three letters** of the last name.
3. Last **two digits** of the birth year.

3.Display the generated username.

```python
# Step 1: Get user input
first_name = input("Enter your first name: ")
last_name = input("Enter your last name: ")
birth_year = input("Enter your birth year (YYYY): ")

# Step 2: Apply slicing
username = first_name[:3] + last_name[:3] + birth_year[-2:]

# Step 3: Display the generated username
print("Your generated username is:", username)
```

Activity – Research and modify the code

Modify the project to ensure:

- The first name and last name are converted to **lowercase**. (.lower())
- The username is **always at least 6 characters long**.
- Allow the user to **customize** the username format (e.g., choose the length of slices).

```
Enter your first name: John
Enter your last name: Doe
Enter your birth year (YYYY): 1995

✅ Your generated username is: johdoe95
```

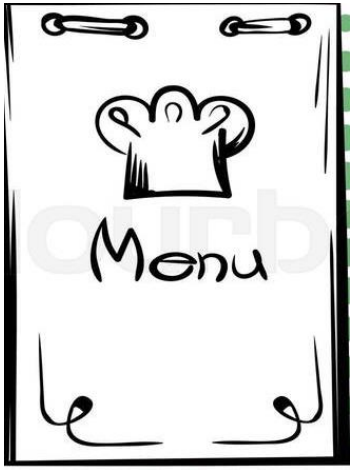
```
Enter your first name: A
Enter your last name: Z
Enter your birth year (YYYY): 2010

✅ Your generated username is: axzxx10
```

Activity – Restaurant menu

Ask the user for their name and location
Ask them to select a dish and quantity from the menu (3 options).
Display the name and “your order is getting ready”

```
Enter your name: Alice
Welcome Alice to Foodie's Delight
Menu:
1. Burger - £5
2. Pizza - £8
3. Pasta - £7
Enter the dish name (1/2/3): 2
Enter quantity: 2
Alice your order is getting ready
Thank you for ordering
```



Conditions

How we make everyday decision?

Evaluating Condition

Let us Python- **if** you like it

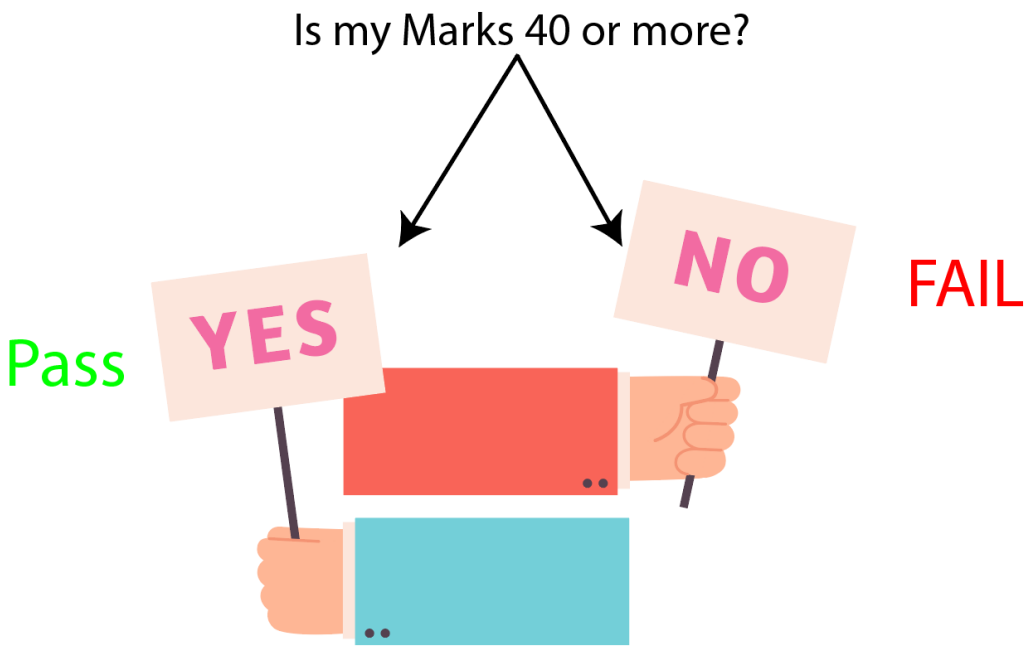
```
# Input student name
name = input("Enter your name: ")

# Input student marks
marks = float(input("Enter your marks: "))

# Print the marks
print(f"Student Name: {name}")
print(f"Marks Obtained: {marks}")
```

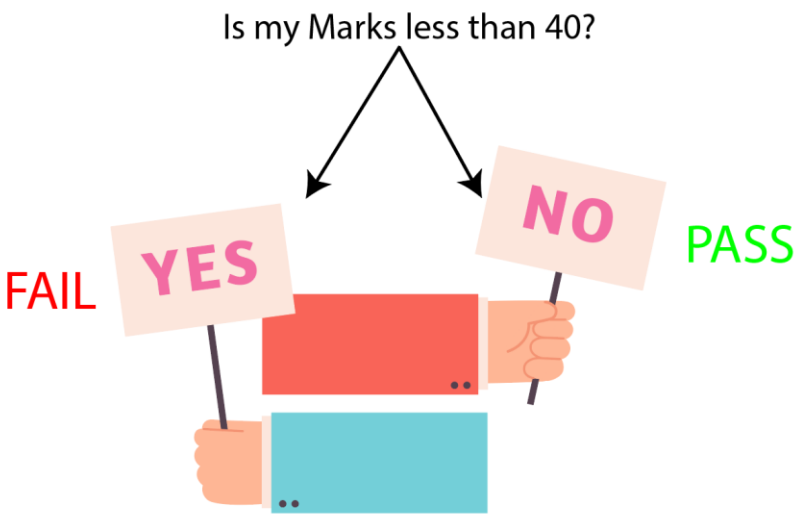
Conditions example

Did I pass my module?



Conditions example

- Did I pass my module?



The condition depends on the question
The question should be answered by **yes/true** OR **no/false**

Syntax- if (life_is_hard): coffee()

```
# Input student name
name = input("Enter your name: ")

# Input student marks
marks = float(input("Enter your marks: "))

# Print the marks
print(f"Student Name: {name}")
print(f"Marks Obtained: {marks}")

# Check if the student passed
if marks >= 40:
    print("Congratulations! You passed.")

# Check if the student failed
if marks < 40:
    print("Sorry, you failed. Better luck next time!")
```

If block

If block

Syntax- if (life_is_hard): coffee()

Thonny - <untitled> @ 17:43

File Edit View Run Tools Help

edge7.py | <untitled> * | <untitled> * | <untitled> * | <untitled> * | <untitled> * | <untitled> *

```python
# Input student name
name = input("Enter your name: ")

# Input student marks
marks = float(input("Enter your marks: "))

# Print the marks
print(f"Student Name: {name}")
print(f"Marks Obtained: {marks}")

# Check if the student passed
if marks >= 40: print("Congratulations! You passed.")
print("Congratulations! You can progress to next term.")
# Check if the student failed
if marks < 40:
    print("Sorry, you failed. Better luck next time!")
    print("you must resit the assessment")
```

If block

If block

Syntax- if (sleepy): coffee() else: still_coffee()

```
1
```

Shell

```
>>>
```

Syntax- if (sleepy): coffee() else: still_coffee()

Write a Python program that takes a student's name and marks as input. The program should then evaluate the student's performance based on the marks and display appropriate messages according to the following criteria:

-If the marks are **40% or above**, the student has **passed**.

-If the marks are **between 20% and 39% (inclusive)**, the student is **allowed to resit the assessment**.

-If the marks are **below 20%**, the student must **redo the course**.

Syntax- if (sleepy): coffee() else: still_coffee()

```python
# Input student name
name = input("Enter your name: ")

# Input student marks
marks = float(input("Enter your marks: "))

# Print the marks
print(f"Student Name: {name}")
print(f"Marks Obtained: {marks}")

# Check conditions using only if-else
if marks >= 40:
    print("Congratulations! You passed.")
else:
    if marks >= 20:  # Marks between 20 and 39
        print("You can resit the assessment.")
    else:  # Marks below 20
        print("You need to redo the course.")
```

if…elif…else statements

```python
# Input student name
name = input("Enter your name: ")
# Input student marks
marks = float(input("Enter your marks: "))
# Print the marks
print(f"Student Name: {name}")
print(f"Marks Obtained: {marks}")

# Check pass/fail conditions
if marks >= 40:
    print("Congratulations! You passed.")
elif marks >= 20:  # Marks between 20 and 39
    print("You can resit the assessment.")
elif marks < 20:  # Marks less than 20
    print("You need to redo the course.")
```

Comparison Operators

Operator	Description	Example
==	If the values of two operands **are equal**, then the condition becomes true.	(a == b) is **False**. (5 == 5) is **True**.
!=	If values of two operands are **not equal**, then condition becomes true.	(a!= b) is **True**.
>	If the value of *left operand* is **greater than** the value of right operand, then condition becomes true.	(a > b) is **False**.
<	If the value of left operand is **less than** the value of right operand, then condition becomes true.	(8 < 9) is **True**.
>=	If the value of left operand is **greater than or equal to** the value of right operand, then condition becomes true.	(5 >= 5) is not true.
<=	If the value of left operand is **less than or equal to** the value of right operand, then condition becomes true.	(2 <= 8) is true.

Logical Operators

Operator	Description	Example
and Logical **AND**	If **both** the operands are **true** then *condition becomes true*.	(a **and** b) is False.
or Logical **OR**	If **any** of the two operands are **non-zero (true)** then *condition becomes true*.	(a **or** b) is True.
not Logical **NOT**	Used to **reverse** the logical state of its operand.	Not(a **and** b) is True [*reversed answer to row 1*].

Membership Operators

Operator	Description	Example
in	Evaluates to *true if it finds the value/variable in the specified sequence* and false otherwise.	x **in** y, here results in a 1 (true) if x is a member of sequence y.
not in	Evaluates to *true if it does not find a variable* in the specified sequence and false otherwise.	x **not in** y, here results in a 1 (true) if x is not a member of sequence y.

Simple if

```
if condition:
    statement(s)
```

If the condition evaluate to TRUE, then the statement(s) will be executed.

Simple **if** example

A program demonstrate simple "**if**" usage. Check if the input number is positive

```
a= int(input("Enter the value of a : "))

if (a>0):
        print("The value ",a," is positive")

print("This is an example of simple if")
```

Output:
Enter the value of a : 5
The value 5 is positive
This is an example of simple if

Enter the value of a : -2
This is an example of simple if

Simple if .. else

```
if (condition):
      statement(s)
else:
       statement(s)
```

- **If** the condition evaluate to TRUE, then the statement(s) under if part will be executed.
- *Otherwise* (**else**), the statement(s) under else part will be executed.

if..else example

A program demonstrate if..else:

```
a= int(input("Enter the value of a : "))
if (a>0):
      print("The value ",a," is positive")
else
      print("The value of ",a," is negative")

print("This is an example of if..else")
```

Output:

Enter the value of a : 5
The value 5 is positive
This is an example of if..else

Enter the value of a : -2
The value -2 is negative
This is an example of if..else

if … elif … else

```
if condition1:
    statement(s)1
elif condition2:
    statement(s)2
else:
    statement(s)1
```

if..elif..else

A program demonstrate if..elif..else:

```
a= int(input("Enter the value of a : "))

if (a>0):
    print("The value ",a," is positive")
elif (a<0):
    print("The value of ",a," is negative")
else :
    print("The value of  "a, " is neutral")

print("This is an example of if..elif..else")
```

Output:

Enter the value of a : 5
The value 5 is positive
This is an example of if..elif..else

Enter the value of a : -2
The value -2 is negative
This is an example of if..elif..else

Enter the value of a : 0
The value of 0 is neutral
This is an example of if..elif..else

Nested if..else statement

Conditional statements can be nested (one inside another).

Note: **Indentations** (spaces at the beginning) are used to indicated the pairings of the **if else** statements (i.e., which ones work together).

For example, a program may check if a person is eligible to vote based on age and citizenship status.

```
age = int(input("Enter your age: "))
citizen = input("Are you a citizen? (yes/no): ")

if age >= 18:
    if citizen.lower() == 'yes':
        print("Eligible to vote")
    else:
        print("Not eligible to vote because you are not a citizen")
else:
    if citizen.lower() == 'yes':
        print("Not eligible to vote because you are under 18")
    else:
        print("Not eligible to vote because you are under 18 and not a citizen")
```

Best Practices and Common Mistakes

- **Best Practices**
  - Simple conditions
  - avoiding deeply nested conditionals
  - clarity over cleverness.

- **Common Mistakes**
  - Confusing == (equality test) with = (assignment)
  - neglecting readability.

Activity- conditional statements

please scan or visit the link for the questions and submission point

https://forms.office.com/e/rAxL1R27mE

Activity-Age Classification Program

Write a program that asks for a user's age and categorizes them into different age groups.

<u>Instructions:</u>

Input the user's age.

If the age is less than 13, display "You are a child."

If the age is between 13 and 19 (inclusive), display "You are a teenager."

If the age is between 20 and 60 (inclusive), display "You are an adult."

If the age is above 60, display "You are a senior."

Activity-Age Classification Program

Write a program that asks for a user's age and categorizes them into different age groups.

Instructions:

Input the user's age.

If the age is less than 13, display "You are a child."

If the age is between 13 and 19 (inclusive), display "You are a teenager."

If the age is between 20 and 60 (inclusive), display "You are an adult."

If the age is above 60, display "You are a senior."

Modify the program to check if the age is less than '0' to print 'this is an invalid age'

Activity-Even or Odd Number Checker

Write a program to check if a given number is even or odd.

Instructions:

Input a number from the user. If the number is divisible by 2 (i.e., number **%** 2 == 0), print "Even."

If the number is not divisible by 2 (i.e., number % 2 **!=** 0), print "Odd."

Activity-Voting Eligibility Program

Write a program that checks if a user is eligible to vote.

Instructions:

Input the user's age.

If the age is 18 or older, print "You are eligible to vote."

If the age is less than 18, print "You are not eligible to vote yet."

Activity-Number Comparison Program

Write a program that takes two numbers as input and compares them.

Instructions:

Input two numbers from the user.

If the first number is greater than the second, print "The first number is greater."

If the first number is smaller than the second, print "The second number is greater."

If the two numbers are equal, print "The numbers are equal."

Activity-Simple Calculator Program

Create a simple calculator that performs basic operations.

Instructions:

Input two numbers from the user.

Ask the user to choose an operation: addition, subtraction, multiplication, or division.

If the operation is addition, print the sum.

If the operation is subtraction, print the difference.

If the operation is multiplication, print the product.

If the operation is division, print the quotient and remainder.

Activity-Grade Evaluation Program

Write a program to assign a grade based on the marks input by the user.

Instructions:

Input the marks from the user.

If the marks are above or equal to 90, assign grade "A."

If the marks are between 70 and 89, assign grade "B."

If the marks are between 50 and 69, assign grade "C."

If the marks are below 50, assign grade "F."

Activity-Display sequence of number

Write a program to print the sequence of numbers up to n.

Instructions:

Input the number n between 1 and 5 from the user.

If the number is 1, print 1.

If the number is 2, print 1,2.

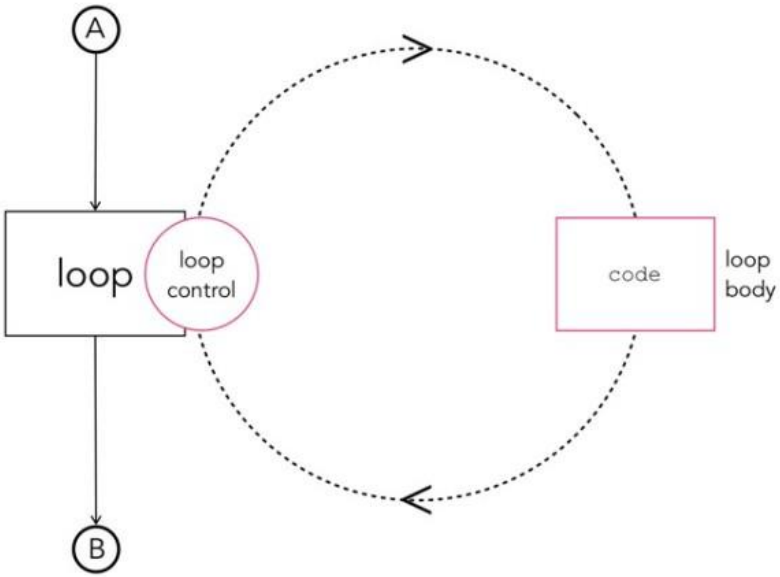
If the number is 3, print 1,2,3.

If the number is 4, print 1,2,3,4.

If the number is 5, print 1,2,3,4,5.

```
Enter a number: 5
1 2 3 4 5
```

Next session



Control structures

-Looping

for Loop

while Loop