پایتون جامع

محمد حسین مقدم خواه

Mhmk.nova@proton.me

جلسه سوم

Topics

- Conditionals
- While Loop
- List Datatype
- For Loop

Control Flow

- If - Elif - Else Structure
- Shorthand If
- Shorthand If-Else
- Match Case (Python +3.10)

References:

- https://www.geeksforgeeks.org/python-if-else/
- https://www.geeksforgeeks.org/python-match-case-statement/
- https://www.w3schools.com/python/python_conditions.asp

Exrecises

- https://csiplearninghub.com/python-if-else-conditional-statement-practice/

If – Elif – Else Structure

```python
num_1 = 10
num_2 = 20

if num_1 > num_2:
    print("num_1 is greater than num_2")
elif num_1 < num_2:
    print("num_2 is greater than num_1")
else:
    print("num_1 and num_2 are equal")

"""
OUT => num_2 is greater than num_1
"""
```

```python
num_1 = 10
num_2 = 20

if num_1 > num_2: print("num_1 is greater than num_2")
"""
OUT => num_2 is greater than num_1
"""
```

Shorthand If Statement

اگر عمل متقابل شرط ما یک خطی باشد میتوانیم شرط خود را در یک خط بنویسیم.

```python
num_1 = 10
num_2 = 20

print("num_1 is greater than num_2") if num_1 > num_2 else print("num_2 is greater than num_1")
"""
OUT => num_2 is greater than num_1
"""
```

Shorthand If-Else

همچنین در پایتون برای حالت شرط If – Elseهم حالت خلاصه داریم.

Match Case (Python +3.10)

در پایتون 3.10به بالا مفهوم Match – Case نیز اضافه شده که مانند ساختار Switch – Case در سایر زبان ها است.

```python
usr_input = int(input("Enter a number: "))

match usr_input:
    case 1:
        print("One")
    case 2:
        print("Two")
    case 3:
        print("Three")
    case _:
        print("Unknown")

"""
IN => 2
OUT => Two
-----------
IN => 5
OUT => Unknown
"""
```

While Loop Statement

References:

- GeekForGeeks
- W3School

Exrecises

While Loop:

نوعی حلقه کنترل شده است که تا زمانی که شرط آن Trueباشد، حلقه ادامه میابد.

همچنین برای انجام عملیات در پایان حلقه، از elseاستفاده میکنیم.

```python
num = 10

while num > 0:
    print(num)
    num -= 1
else:
    print("Done")


"""
OUT => 10, 9, 8 , 7, 6, 5, 4, 3
"""
```

Break & Continue

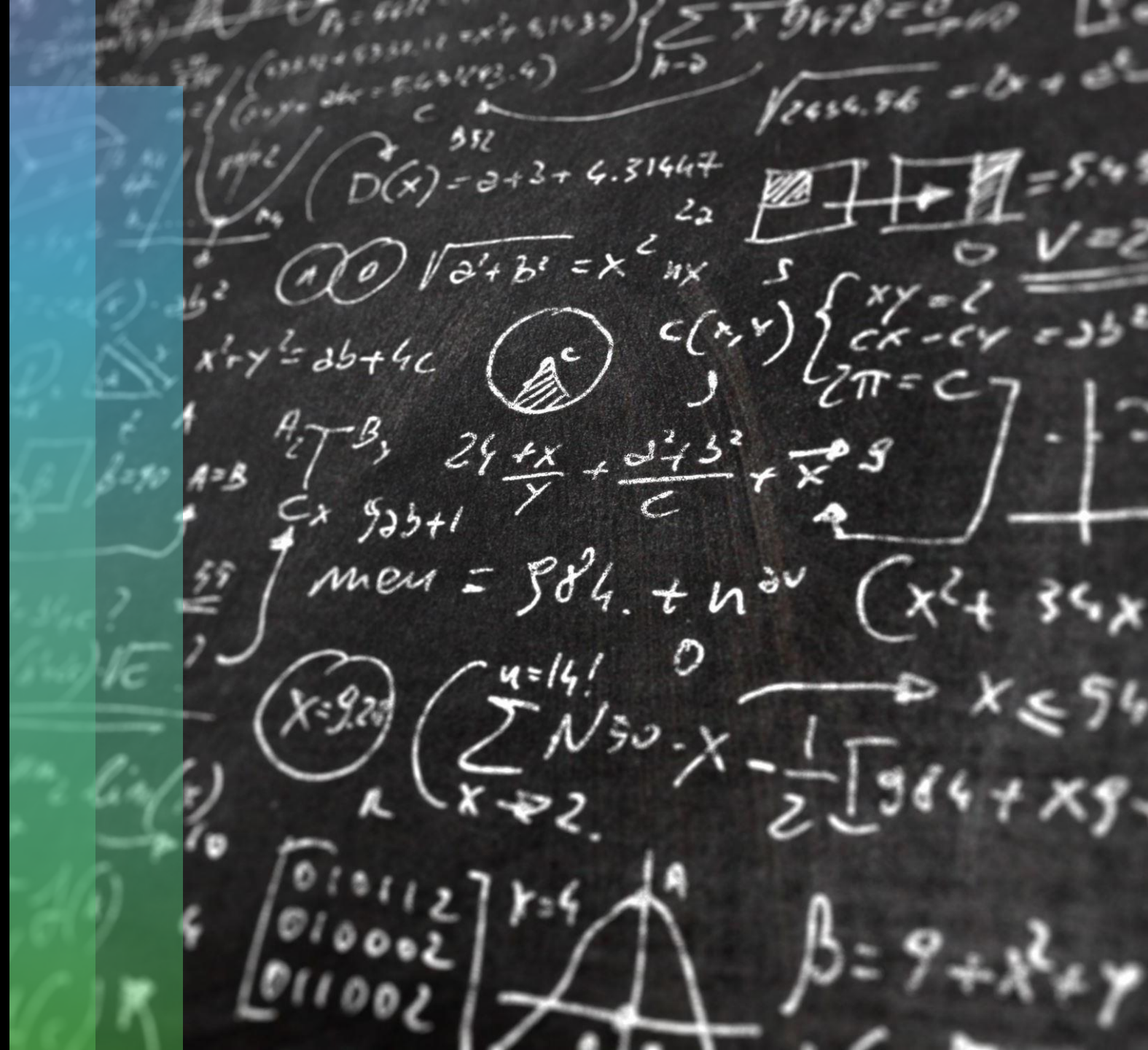
از breakبرای شکستن حلقه و
continueبرای پرش کیس فعلی و رفتن
روی کیس بعدی استفاده میشود.

```python
num = 10

while num > 0:
    print(num)
    num -= 1
    if num == 5:
        continue
    elif num == 3:
        break

"""
OUT => 10, 9, 8, 7, 6, 4
"""
```

List

References:

- GeekForGeeks
- W3School



```python
list_1 = [1, 3.14, "hello", True, 3 + 4j, [1, 2, 3]]

print(list_1)

"""
OUT => [1, 3.14, "hello", True, (3 + 4j), [1, 2, 3]]
"""
```

List DataType

دیتا تایپ دیگری که در پایتون داریم، list است که آرایه ای از مقادیر مختلف است که:

- ترتیب دارند
- میتواند دیتا تایپ های مختلف را در خود جای دهد
- قابل تغییر است.

Range() Function

از این تابع برای ساخت لیست از یک مقدار تا مقدار دیگر استفاده میکنیم که ورودی های آن عبارتند از:

1. مقدار شروع
2. مقدار پایان
3. قدم پیشروی

```python
list_1 = list(range(1, 10, 2))

print(list_1)

"""
OUT => [1, 3, 5, 7, 9]
"""
```

Iterables DataTypes

دیتا تایپ هایی که میتوانیم روی آنها حلقه بزنیم که تا اینجا خوانده ایم عبارتند از:

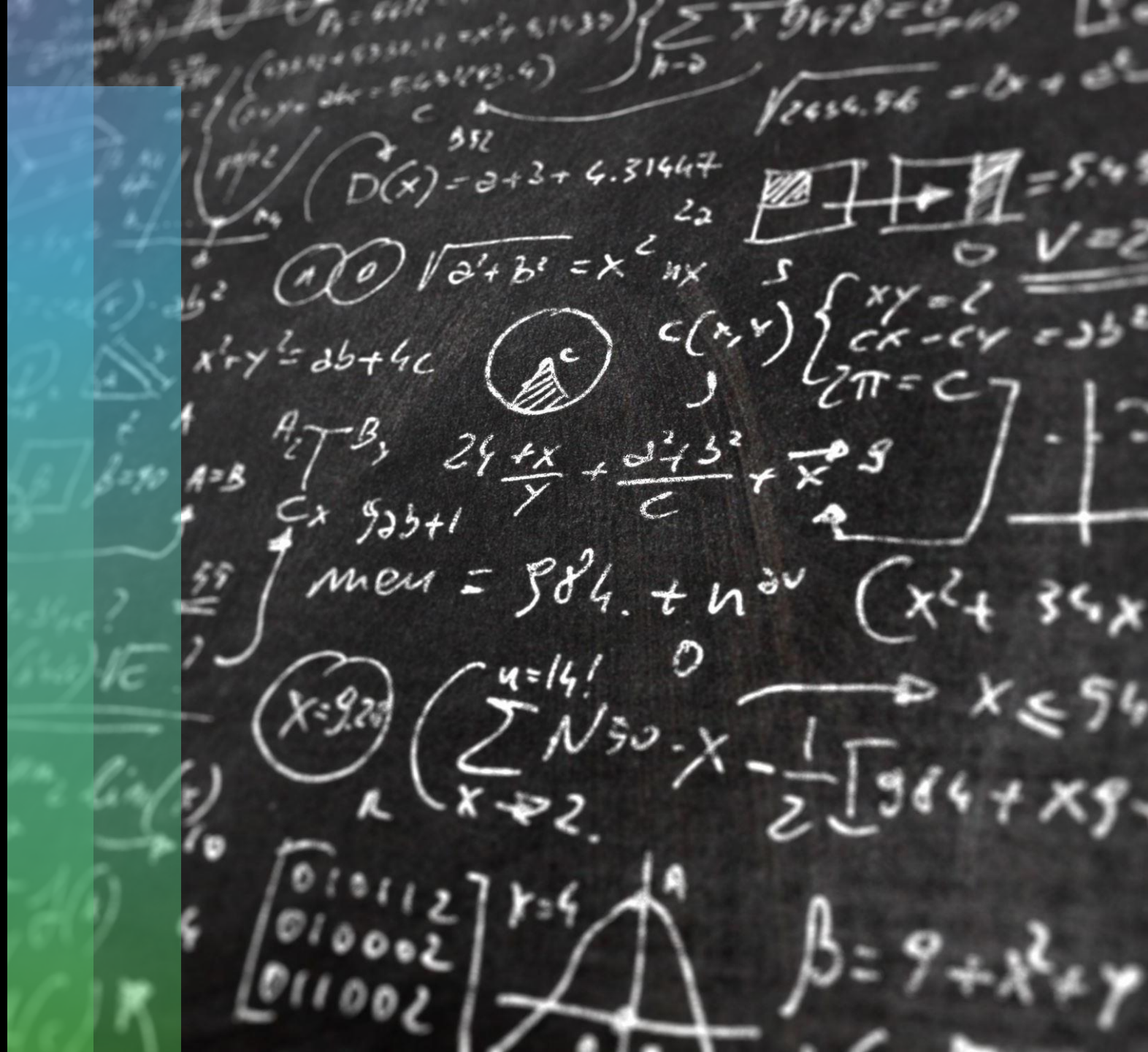
- String: در واقع این نوع داده، لیستی از کارکتر هاست
- List

For Loop Statement

References:

- GeekForGeeks
- W3School

  Exrecises



For Loop

از عملگر for برای ایجاد حلقه استفاده میکنیم.

همچنین در انتهای حلقه میتوانیم از else استفاده کنیم.

مانند بقیه ساختار های حلقه از break و continue هم میتوانیم استفاده کنیم.

```
for i in "Hello":
    print(i)
else:
    print("Done1")

for i in [1, 2, 3, 4, 5]:
    print(i)
else:
    print("Done2")

for i in range(1, 10):
    print(i)
else:
    print("Done3")

"""
OUT => H, e, l, l, o, Done1
1, 2, 3, 4, 5, Done2
1, 2, 3, 4, 5, 6, 7, 8, 9, Done3
"""
```

Session 3 Ended.

To Be Continued...