Control Transfer Statement & Functions

Session Objectives

- What control transfer statements are
- How to use the break statement
- How to use the continue statement
- How to use the pass statement
- Understand what functions are and why we use them.
- Learn to define functions, with parameters and arguments.

```
val = int(input("Enter the value to check wether a val is prime or not: "))
for num in range(2,val):
    if val % num == 0:
        print(f"{val} is not a prime number")
        break
else:
    print(f"{val} is a prime number")
```

```
Enter the value to check wether a val is prime or not:  17
17 is a prime number
```

```
for num in range(2,9) # [2,3,4.....8]

    Step 1. 9 % 2 == 0 : False
    num = 3
    Step 2. 9%3 == 0 : True
        print("")-> Console
        break
```

Memory

```
val = None
num = iterator[2,val]

val = 9      val = 7
num = 2
```

```
car_list = ['Taigun','Slavia','Verna','Thar','Innova','Defender','Lord Alto',
            'Safari','Harrier','Bolero','XUV700','Altroz']
for car in car_list:
    if car == 'Bolero': # breaking the loop abruptly
        break
    else:
        print(car , end = " ")
else:
    print() # Moving the statement to the next line
    print("Loop Run Successfully.")
```

```
Taigun Slavia Verna Thar Innova Defender Lord Alto Safari Harrier
```

```python
car_list = ['Taigun','Slavia','Verna','Thar','Innova','Defender','Lord Alto',
            'Safari','Harrier','Bolero','XUV700','Altroz']
for car in car_list:
    if car == 'Fronx': # This condition won't hit as this 'Fronx' is not available in the car_list
        break
    else:
        print(car , end = " ") # print the complete list
else:
    print() # Moving the statement to the next line
    print("Loop Run Successfully.")
```

```
Taigun Slavia Verna Thar Innova Defender Lord Alto Safari Harrier Bolero XUV700 Altroz
Loop Run Successfully.
```

```python
car_list = ['Taigun','Slavia','Verna','Thar','Innova','Defender','Lord Alto',
            'Safari','Harrier','Bolero','XUV700','Altroz']
for car in car_list:
    if (car == 'Innova') or (car == 'XUV700') or (car == 'Thar'): # skip the elements
        continue
    else:
        print(car , end = " ") # print the complete list excluding some cars mention above
else:
    print() # Moving the statement to the next line
    print("Loop Run Successfully.")
```

```
Taigun Slavia Verna Defender Lord Alto Safari Harrier Bolero Altroz
Loop Run Successfully.
```

```python
# pass
car_list = ['Taigun','Slavia','Verna','Thar','Innova','Defender','Lord Alto',
            'Safari','Harrier','Bolero','XUV700','Altroz']
for car in car_list:
    if (car == 'Innova') or (car == 'XUV700') or (car == 'Thar'):
        pass # Logic is missing -> will add it in future
    else:
        print(car , end = " ") # print the complete list excluding some cars mention above
else:
    print() # Moving the statement to the next line
    print("Loop Run Successfully.")
```

```
Taigun Slavia Verna Defender Lord Alto Safari Harrier Bolero Altroz
Loop Run Successfully.
```

```python
# Pass -> Nested List
car_list = [
    ['Taigun','Slavia','Verna'],
    ['Thar','Innova','Defender'],
    ['Lord Alto','Safari','Harrier'],
    ['Bolero','XUV700','Altroz']
]
flatten_car_list = []
for row in car_list :
    for car in row:
        # pass is actually a placeholder that can be filled with the logic in future
        pass # flatten_car_list.append(car)
print(flatten_car_list)
```

```
[]
```

```
nested_num_list = [
    [11,22,33,44,55],
    [2,5,7,9,11],
    [99,77,55,33,11],
    [10,20,30,40,50],
    [15,19,17,91,97]
]
even_list = []
odd_list = []
for row in nested_num_list:
    for val in row:
        if val % 2 == 0: # even
            even_list.append(val)
        else:
            odd_list.append(val)
else:
    print("Even List : " , even_list)
    print("Odd List : " , odd_list)
```

```
Even List :  [22, 44, 2, 10, 20, 30, 40, 50]
Odd List :  [11, 33, 55, 5, 7, 9, 11, 99, 77, 55, 33, 11, 15, 19, 17, 91, 97]
```

```
nested_num_list = [
    [11,22,33,44,55],
    [2,5,7,9,11],
    [99,77,55,33,11],
    [10,20,30,40,50],
    [15,19,17,91,97]
]
even_list = []
odd_list = []
for row in nested_num_list:
    for val in row:
        if val <=10 or val >= 80:
            pass # Placeholder for the logic
        elif val % 2 == 0: # even
            even_list.append(val)
        elif val % 2 == 1: # odd
            odd_list.append(val)
else:
    print("Even List : " , even_list)
    print("Odd List : " , odd_list)
```

```
Even List :  [22, 44, 20, 30, 40, 50]
Odd List :  [11, 33, 55, 11, 77, 55, 33, 11, 15, 19, 17]
```

```python
nested_num_list = [
    [11,22,33,44,55],
    [2,5,7,9,11],
    [99,77,55,33,11],
    [10,20,30,40,50],
    [15,19,17,91,97]
]
even_list = []
odd_list = []
another_list = []
for row in nested_num_list:
    for val in row:
        if val <=10 or val >= 80:
            another_list.append(val) # pass # Placeholder for the logic
        elif val % 2 == 0: # even
            even_list.append(val)
        elif val % 2 == 1: # odd
            odd_list.append(val)
else:
    print("Even List : " , even_list)
    print("Odd List : " , odd_list)
    print("Another List : " , another_list)
```

```
Even List :  [22, 44, 20, 30, 40, 50]
Odd List :  [11, 33, 55, 11, 77, 55, 33, 11, 15, 19, 17]
Another List :  [2, 5, 7, 9, 99, 10, 91, 97]
```

```python
# Prime Number -> Any number either divides by itself or by 1
# break
val = int(input("Enter the value to check wether a val is prime or not: "))
for num in range(2,val):
    if val % num == 0:
        print(f"{val} is not a prime number")
        break
else:
    print(f"{val} is a prime number")
```

```
Enter the value to check wether a val is prime or not:  15
15 is not a prime number
```

```python
val = int(input("Enter the value to check wether a val is prime or not: "))
for num in range(2,val):
    if val % num == 0:
        print(f"{val} is not a prime number")
        break
else:
    print(f"{val} is a prime number")
```

```
Enter the value to check wether a val is prime or not:  17
17 is a prime number
```

```
_dict = {
    "stud_1" : {'name' : 'Aman','age' : 29,'gender' : 'M','city' : 'Bhopal', 'Country' : 'India'},
    "stud_2" : {'name' : 'Sanchita','age' : 27,'gender' : 'F','city' : 'Varansi', 'Country' : 'India'},
    "stud_3" : {'name' : 'Surya','age' : 33,'gender' : 'M','city' : 'Mumbai', 'Country' : 'India'},
    "stud_4" : {'name' : 'Nihal','age' : 28,'gender' : 'M','city' : 'Chandigarh', 'Country' : 'India'},
    "stud_5" : {'name' : 'Lubhani','age' : 26,'gender' : 'F','city' : 'Pune', 'Country' : 'India'}
}
print(_dict.keys())
```

```
dict_keys(['stud_1', 'stud_2', 'stud_3', 'stud_4', 'stud_5'])
```

```
print(_dict['stud_1'].keys())
```

```
dict_keys(['name', 'age', 'gender', 'city', 'Country'])
```

```
print(_dict['stud_1'].values())
```

```
dict_values(['Aman', 29, 'M', 'Bhopal', 'India'])
```

```
_dict.items()
```

```
dict_items([('stud_1', {'name': 'Aman', 'age': 29, 'gender': 'M', 'city': 'Bhopal', 'Country': 'India'}), ('st
ud_2', {'name': 'Sanchita', 'age': 27, 'gender': 'F', 'city': 'Varansi', 'Country': 'India'}), ('stud_3', {'na
me': 'Surya', 'age': 33, 'gender': 'M', 'city': 'Mumbai', 'Country': 'India'}), ('stud_4', {'name': 'Nihal',
'age': 28, 'gender': 'M', 'city': 'Chandigarh', 'Country': 'India'}), ('stud_5', {'name': 'Lubhani', 'age': 2
6, 'gender': 'F', 'city': 'Pune', 'Country': 'India'})])
```

```
for stud_key in _dict.keys():
    print(_dict[stud_key])
```

```
{'name': 'Aman', 'age': 29, 'gender': 'M', 'city': 'Bhopal', 'Country': 'India'}
{'name': 'Sanchita', 'age': 27, 'gender': 'F', 'city': 'Varansi', 'Country': 'India'}
{'name': 'Surya', 'age': 33, 'gender': 'M', 'city': 'Mumbai', 'Country': 'India'}
{'name': 'Nihal', 'age': 28, 'gender': 'M', 'city': 'Chandigarh', 'Country': 'India'}
{'name': 'Lubhani', 'age': 26, 'gender': 'F', 'city': 'Pune', 'Country': 'India'}
```

```
for stud_key in _dict.keys():
    print(_dict[stud_key]['city'])
```

```
Bhopal
Varansi
Mumbai
Chandigarh
Pune
```

```
for student_detail in _dict.items():
    print(student_detail) # A complete inner dictionary
```

```
('stud_1', {'name': 'Aman', 'age': 29, 'gender': 'M', 'city': 'Bhopal', 'Country': 'India'})
('stud_2', {'name': 'Sanchita', 'age': 27, 'gender': 'F', 'city': 'Varansi', 'Country': 'India'})
('stud_3', {'name': 'Surya', 'age': 33, 'gender': 'M', 'city': 'Mumbai', 'Country': 'India'})
('stud_4', {'name': 'Nihal', 'age': 28, 'gender': 'M', 'city': 'Chandigarh', 'Country': 'India'})
('stud_5', {'name': 'Lubhani', 'age': 26, 'gender': 'F', 'city': 'Pune', 'Country': 'India'})
```

```python
for student_code , stud_info in _dict.items():
    print(student_code, end = " :")
    print(stud_info)
```

```
stud_1 :{'name': 'Aman', 'age': 29, 'gender': 'M', 'city': 'Bhopal', 'Country': 'India'}
stud_2 :{'name': 'Sanchita', 'age': 27, 'gender': 'F', 'city': 'Varansi', 'Country': 'India'}
stud_3 :{'name': 'Surya', 'age': 33, 'gender': 'M', 'city': 'Mumbai', 'Country': 'India'}
stud_4 :{'name': 'Nihal', 'age': 28, 'gender': 'M', 'city': 'Chandigarh', 'Country': 'India'}
stud_5 :{'name': 'Lubhani', 'age': 26, 'gender': 'F', 'city': 'Pune', 'Country': 'India'}
```

```python
for student_code , stud_info in _dict.items():
    print(student_code , end = " -> ")
    for person_key , person_detail in stud_info.items():
        print()
        print(end = "\t")
        print(person_key , end = " :")
        print(person_detail)
```

```
stud_1 ->
        name :Aman

        age :29

        gender :M

        city :Bhopal

        Country :India
stud_2 ->
        name :Sanchita

        age :27

        gender :F
```

```python
for student_code , stud_info in _dict.items():
    if student_code == 'stud_2':
        continue
    print(student_code , end = " -> ")
    for person_key , person_detail in stud_info.items():
            if person_key == 'city':
                print(person_detail)
```

```
stud_1 -> Bhopal
stud_3 -> Mumbai
stud_4 -> Chandigarh
stud_5 -> Pune
```

```python
for student_code , stud_info in _dict.items():
    if student_code == 'stud_2':
        continue
    print(student_code , end = " -> ")
    for person_key , person_detail in stud_info.items():
        if person_key == 'city':
            if _dict[student_code][person_key] == 'Bhopal':
                _dict[student_code][person_key] = 'Delhi'
            elif  _dict[student_code][person_key] == 'Mumbai':
                pass
            else:
                break
```

```
stud_1 -> stud_3 -> stud_4 -> stud_5 ->
```

```python
for student_code , stud_info in _dict.items():
    print(student_code , end = " -> ")
    for person_key , person_detail in stud_info.items():
        print()
        print(end = "\t")
        print(person_key , end = " :")
        print(person_detail)
```

```
stud_1 ->
        name :Aman

        age :29

        gender :M

        city :Delhi

        Country :India
stud_2 ->
        name :Sanchita

        age :27

        gender :F

        city :Varansi

        Country :India
stud_3 ->
        name :Surya

        age :33

        gender :M

        city :Mumbai

        Country :India
stud_4 ->
        name :Nihal

        age :28

        gender :M

        city :Chandigarh

        Country :India
stud_5 ->
        name :Lubhani

        age :26

        gender :F

        city :Pune

        Country :India
```

Functions :

Functions are reusable block of code that helps you :

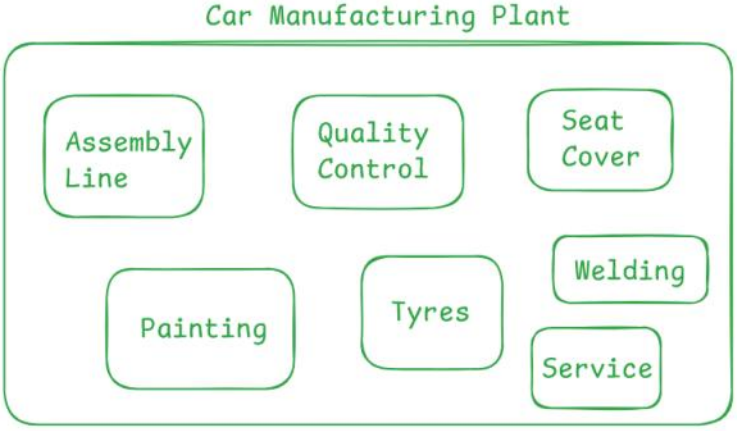
1. Organize your code.
2. Avoid Repoid (DRY): 'Do not Repeat Yourself'
3. Makes your program easy to understand and debug.

```
Syntax:
def function_name :
    # Function Logic [Print Vs Return]
```

```python
def greet():
    print("Hello World!")

greet()
greet()
greet()
greet()
```

```
Hello World!
Hello World!
Hello World!
Hello World!
```




```python
def squared(val):
    return val ** 2
def cubed(val):
    return val ** 3

sq1 = squared(10) # 100
print(sq1) # 100
sq2 = squared(11) # 121
print(sq2) # 221
sq3 = squared(21) # 441
print(sq3) # 441
cu1 = cubed(10) # 1000
print(cu1)
```

```
100
121
441
1000
```