# Conditional & Looping Statement

## 🎯 Session Objectives

- ✅ Understand what control statements are.
- ✅ Understand the importance of control statements.
- ✅ Understand the types of control statements.
- ✅ Understand what conditional statements are.
- ✅ Understand what nested conditional statements are.
- ✅ Understand conditional statements with shorthand notations.
- ✅ Understand what looping statements are
- ✅ Understand what a for loop is
- ✅ Understand what a while loop is

Control statements in Python are like traffic signals for your code — they decide which parts of your program should run and when.

Traffic light analogy:

Green → Go → Run code block

Red → Stop → Skip code block

Syntax : If

```
If condition:
    # code block
```

Syntax : If-else

```
if condition:
    # Code block to execute if the condition is true
else:
    # Code block to execute if the condition is false
```

Syntax : If-elif-else

```
if condition1:
    # Block if condition1 true
elif condition2:
    # Block if condition2 true
elif condition3:
    # Block if condition3 true
...
else:
    # Block if all conditions are false
```

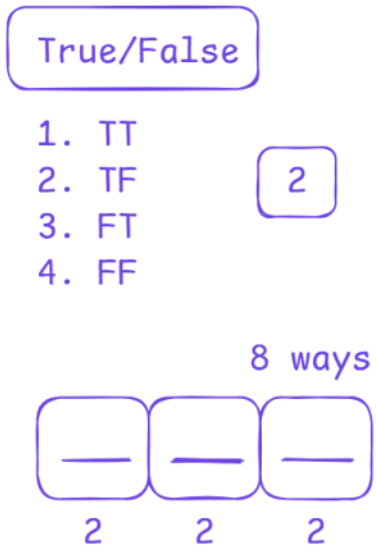
Syntax : Nested-if

```
if condition1:
    # Block if condition1 is true
    if condition2:
        # Block if both condition1 and condition2 are true
```

Syntax : Nested if-else

```
if condition1:
    if condition2:
        # Block if both condition1 & condition2 true
    else:
        # Block if condition1 true, condition2 false
else:
    # Block if condition1 is false

or
if condition:
    # Block if condition is true
else:
    if another_condition:
        # Block if another_condition is true
    else:
        # Block if another_condition is false
```




Syntax : Nested if-elif-else

```
if condition1:
    if condition2:
        # Block if both condition1 & condition2 true
    elif condition3:
        # Block if condition1 true, condition2 false, condition3 true
    else:
        # Block if condition1 true, others false
elif condition4:
    # Block if condition1 false, condition4 true
else:
    # Block if all conditions false
```

```python
# If Condition :
    # Block Code
val = int(input("Enter the Value: "))
if val > 100: # [Boolean Return]
    print(f"{val} is greater than 100.")
print("Outside the If Condition, Will Always Work....")
```

```
Enter the Value:  111
111 is greater than 100.
Outside the If Condition, Will Always Work....
```

```python
val = int(input("Enter the Value: "))
if val > 100: # [Boolean Return]
    print(f"{val} is greater than 100.")
print("Outside the If Condition, Will Always Work....")
```

```
Enter the Value:  11
Outside the If Condition, Will Always Work....
```

```python
val = int(input("Enter the Value: "))
if val > 100: # [Boolean Return]
    print(f"{val} is greater than 100.")
print("Outside the If Condition, Will Always Work....")
```

```
Enter the Value:  -1
Outside the If Condition, Will Always Work....
```

```python
val = int(input("Enter the Value: ")) # ValueError: invalid literal for int() with base 10: 'A'
if val > 100: # [Boolean Return]
    print(f"{val} is greater than 100.")
print("Outside the If Condition, Will Always Work....")
```

```python
num1 = int(input("Enter the value of num1: "))
num2 = int(input("Enter the value of num2: "))
if num1 > num2:
    print(f"{num1} is greater than {num2}")
if num2 > num1:
    print(f"{num2} is greater than {num1}")
if num1 == num2:
    print(f"{num1} is equal to {num2}")
if num1 != num2:
    print(f"{num1} is not equal to {num2}")
```

```
Enter the value of num1:  -99
Enter the value of num2:  77
77 is greater than -99
-99 is not equal to 77
```

```python
if condition:
    # Code block to execute if the condition is true
else:
    # Code block to execute if the condition is false
```

```python
_bool = True
print(int(_bool))
print(_bool)
```

```
1
True
```

```python
_bool = False
print(int(_bool))
print(_bool)
```

```
0
False
```

```python
# If any number is either positive or negative : True
# If any number is equal to 0,{},[],(),set() : False
if 0: # False
    print("Hello World")
else:
    print("Python Programming")
```

```
Python Programming
```

```python
if {}: # False
    print("Hello World")
else:
    print("Python Programming")
```

```
Python Programming
```

```python
if set(): # False
    print("Hello World")
else:
    print("Python Programming")
```

```
Python Programming
```

```python
if []: # False
    print("Hello World")
else:
    print("Python Programming")
```

```
Python Programming
```

```python
if (): # False
    print("Hello World")
else:
    print("Python Programming")
```

```
Python Programming
```

```python
if -9999: # True
    print("Hello World")
else:
    print("Python Programming")
```

```
Hello World
```

```python
if 777: # True
    print("Hello World")
else:
    print("Python Programming")
```

```
Hello World
```

```python
if [1,2,3,4]: # True
    print("Hello World")
else:
    print("Python Programming")
```

```
Hello World
```

```python
val = int(input("Enter the Value: "))
if val > 100: # [Boolean Return]
    print(f"{val} is greater than 100.")
else:
    print(f"{val} is less than or equal to 100.")
```

```
Enter the Value:  77
77 is less than or equal to 100.
```

```python
val = int(input("Enter the Value: "))
if val > 100: # [Boolean Return]
    print(f"{val} is greater than 100.")
else:
    print(f"{val} is less than or equal to 100.")
```

```
Enter the Value:  121
121 is greater than 100.
```

```python
# Checking for a multiple of 10
val = int(input("Enter the Value: "))
if val%10 == 0: # [Boolean Return]
    print(f"{val} is a multiple of 10.")
else:
    print(f"{val} is not a multiple of 10.")
```

```
Enter the Value:  1100
1100 is a multiple of 10.
```

```python
# Checking for a multiple of 10
val = int(input("Enter the Value: "))
if val%10 == 0: # [Boolean Return]
    print(f"{val} is a multiple of 10.")
else:
    print(f"{val} is not a multiple of 10.")
```

```
Enter the Value:  1234
1234 is not a multiple of 10.
```

```python
# Checking the type of an input
val = '11'
print(type(val)) # 'str'
if val == 11: # False
    print(f"{val} is a numeric value")
else:
    print("Please fix the data-type of the input, as it is not a numeric Type!")
```

```
<class 'str'>
Please fix the data-type of the input, as it is not a numeric Type!
```

```python
# Checking the type of an input
val = 11.1
print(type(val)) # 'float'
if type(val) is float:
    print(f"{val} is a float value")
else:
    print("Please fix the data-type of the input, as it is not a numeric Type!")
```

```
<class 'float'>
11.1 is a float value
```

```python
# Data Structures
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv','Harrier','Safari','Lord Alto','Thar',
            'Virtus','Defender','Innova','Baleno','Legender','ScorpioN','Grand Vitara']
fav_car1 = 'Virtus' # True
fav_car2 = 'Safari' # True
if fav_car1 in car_list or fav_car2 in car_list: # [TF,TT,FT]
    print("Looking to Evaluate if both favorite car exist in stock or not.")
    if fav_car1 in car_list and fav_car2 in car_list: # [TT]
        print(f"Congrats! Both Car {fav_car1} & {fav_car2} are available in stock")
    elif fav_car1 in car_list:
        print(f"We only have {fav_car1} in stock.")
    elif fav_car2 in car_list:
        print(f"We only have {fav_car2} in stock.")
else: # [FF]
    print("Both Cars doesn't exist on Stock. Will Update it soon....")
```

```
Looking to Evaluate if both favorite car exist in stock or not.
Congrats! Both Car Virtus & Safari are available in stock
```

```python
# Data Structures
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv','Harrier','Safari','Lord Alto','Thar',
            'Virtus','Defender','Innova','Baleno','Legender','ScorpioN','Grand Vitara']
fav_car1 = 'Mini Cooper' # False
fav_car2 = 'Safari' # True
if fav_car1 in car_list or fav_car2 in car_list: # [TF,TT,FT]
    print("Looking to Evaluate if both favorite car exist in stock or not.")
    if fav_car1 in car_list and fav_car2 in car_list: # [TT]
        print(f"Congrats! Both Car {fav_car1} & {fav_car2} are available in stock")
    elif fav_car1 in car_list:
        print(f"We only have {fav_car1} in stock.")
    elif fav_car2 in car_list:
        print(f"We only have {fav_car2} in stock.")
else: # [FF]
    print("Both Cars doesn't exist on Stock. Will Update it soon....")
```

```
Looking to Evaluate if both favorite car exist in stock or not.
We only have Safari in stock.
```

```python
# Data Structures
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv','Harrier','Safari','Lord Alto','Thar',
            'Virtus','Defender','Innova','Baleno','Legender','ScorpioN','Grand Vitara']
fav_car1 = 'Defender' # True
fav_car2 = 'Fortuner' # False
if fav_car1 in car_list or fav_car2 in car_list: # [TF,TT,FT]
    print("Looking to Evaluate if both favorite car exist in stock or not.")
    if fav_car1 in car_list and fav_car2 in car_list: # [TT]
        print(f"Congrats! Both Car {fav_car1} & {fav_car2} are available in stock")
    elif fav_car1 in car_list:
        print(f"We only have {fav_car1} in stock.")
    elif fav_car2 in car_list:
        print(f"We only have {fav_car2} in stock.")
else: # [FF]
    print("Both Cars doesn't exist on Stock. Will Update it soon....")
```

```
Looking to Evaluate if both favorite car exist in stock or not.
We only have Defender in stock.
```

```python
# Data Structures
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv','Harrier','Safari','Lord Alto','Thar',
            'Virtus','Defender','Innova','Baleno','Legender','ScorpioN','Grand Vitara']
fav_car1 = 'Mini Cooper' # False
fav_car2 = 'Fortuner' # False
if fav_car1 in car_list or fav_car2 in car_list: # [TF,TT,FT]
    print("Looking to Evaluate if both favorite car exist in stock or not.")
    if fav_car1 in car_list and fav_car2 in car_list: # [TT]
        print(f"Congrats! Both Car {fav_car1} & {fav_car2} are available in stock")
    elif fav_car1 in car_list:
        print(f"We only have {fav_car1} in stock.")
    elif fav_car2 in car_list:
        print(f"We only have {fav_car2} in stock.")
else: # [FF]
    print("Both Cars doesn't exist on Stock. Will Update it soon....")
```

```
Both Cars doesn't exist on Stock. Will Update it soon....
```

```python
# Data Structures
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv','Harrier','Safari','Lord Alto','Thar',
            'Virtus','Defender','Innova','Baleno','Legender','ScorpioN','Grand Vitara']
fav_car1 = 'Mini Cooper' # False
fav_car2 = 'Fortuner' # False

if fav_car1 in car_list and fav_car2 in car_list: # [TT]
    print(f"Congrats! Both Car {fav_car1} & {fav_car2} are available in stock")
elif fav_car1 in car_list: # [TF]
    print(f"We only have {fav_car1} in stock.")
elif fav_car2 in car_list: # [FT]
    print(f"We only have {fav_car2} in stock.")
else: # [FF]
    print(f"Both {fav_car1} & {fav_car2} doesn't exist on Stock. Will Update it soon....")
```

```
Both Mini Cooper & Fortuner doesn't exist on Stock. Will Update it soon....
```

```python
# Data Structures
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv','Harrier','Safari','Lord Alto','Thar',
            'Virtus','Defender','Innova','Baleno','Legender','ScorpioN','Grand Vitara']
fav_car1 = input("Enter your 1st Fav Car: ")
fav_car2 = input("Enter your 2nd Fav Car: ")

if fav_car1 in car_list and fav_car2 in car_list: # [TT]
    print(f"Congrats! Both Car {fav_car1} & {fav_car2} are available in stock")
elif fav_car1 in car_list: # [TF]
    print(f"We only have {fav_car1} in stock.")
elif fav_car2 in car_list: # [FT]
    print(f"We only have {fav_car2} in stock.")
else: # [FF]
    print(f"Both {fav_car1} & {fav_car2} doesn't exist on Stock. Will Update it soon....")
```

```
Enter your 1st Fav Car:  City
Enter your 2nd Fav Car:  Kylaq
We only have City in stock.
```

```python
# Data Structures
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv','Harrier','Safari','Lord Alto','Thar',
            'Virtus','Defender','Innova','Baleno','Legender','ScorpioN','Grand Vitara']
fav_car1 = input("Enter your 1st Fav Car: ")
fav_car2 = input("Enter your 2nd Fav Car: ")

if fav_car1 in car_list and fav_car2 in car_list: # [TT]
    print(f"Congrats! Both Car {fav_car1} & {fav_car2} are available in stock")
elif fav_car1 in car_list: # [TF]
    print(f"We only have {fav_car1} in stock.")
elif fav_car2 in car_list: # [FT]
    print(f"We only have {fav_car2} in stock.")
else: # [FF]
    print(f"Both {fav_car1} & {fav_car2} doesn't exist on Stock. Will Update it soon....")
```

```
Enter your 1st Fav Car:  Kylaq
Enter your 2nd Fav Car:  Harrier
We only have Harrier in stock.
```

```python
# Dictionary # Membership Operator [Return True if key exist]
student_details = {
    'name' : 'Rajat Singh Thakur',
    'age' : 29,
    'gender' : 'Male',
    'city' : 'Shimla',
    'state' : 'Himachal Pradesh',
    'country' : 'India',
    'course' : 'Data Analytics',
    'skills' : ['Excel','PowerBI','MySQL','Python']
}
key_to_search = input("Enter the valid key: ")
if key_to_search in student_details:
    print("You are Searching : ", student_details[key_to_search])
else:
    print("No Such Key Exist.")
```

```
Enter the valid key:  course
You are Searching :  Data Analytics
```

```python
# Dictionary # Membership Operator [Return True if key exist]
student_details = {
    'name' : 'Rajat Singh Thakur',
    'age' : 29,
    'gender' : 'Male',
    'city' : 'Shimla',
    'state' : 'Himachal Pradesh',
    'country' : 'India',
    'course' : 'Data Analytics',
    'skills' : ['Excel','PowerBI','MySQL','Python']
}
key_to_search = input("Enter the valid key: ")
if key_to_search in student_details:
    print("You are Searching : ", student_details[key_to_search])
else:
    print("No Such Key Exist.")
```

```
Enter the valid key:  skills
You are Searching :  ['Excel', 'PowerBI', 'MySQL', 'Python']
```

```python
# Dictionary # Membership Operator [Return True if key exist]
student_details = {
    'name' : 'Rajat Singh Thakur',
    'age' : 29,
    'gender' : 'Male',
    'city' : 'Shimla',
    'state' : 'Himachal Pradesh',
    'country' : 'India',
    'course' : 'Data Analytics',
    'skills' : ['Excel','PowerBI','MySQL','Python']
}
key_to_search = input("Enter the valid key: ")
if key_to_search in student_details:
    print("You are Searching : ", student_details[key_to_search])
else:
    print("No Such Key Exist.")
```

```
Enter the valid key:  email_address
No Such Key Exist.
```

```python
# Dictionary # Membership Operator [in/not in] [Return True if key exist]
student_details = {
    'name' : 'Rajat Singh Thakur',
    'age' : 29,
    'gender' : 'Male',
    'city' : 'Shimla',
    'state' : 'Himachal Pradesh',
    'country' : 'India',
    'course' : 'Data Analytics',
    'skills' : ['Excel','PowerBI','MySQL','Python']
}
key_to_search = input("Enter the valid key: ")
if key_to_search in student_details:
    print("You are Searching : ", student_details[key_to_search])
elif key_to_search not in student_details:
    print("No Such Key Exist.")
```

```
Enter the valid key:  Phone_Number
No Such Key Exist.
```

```python
num = 900
if num == 1000:
    print("num is equal to 1000")
elif num >= 700:
    print("num is greater than or equal to 700")
elif num >= 500:
    print("num is greater than or equal to 500")
else:
    print("num is less than 500.")
```

```
num is greater than or equal to 700
```

```python
# Find the max marks by comparing
marks1 = int(input("Enter Your Marks1 : ")) # 99
marks2 = int(input("Enter Your Marks2 : ")) # 92
marks3 = int(input("Enter Your Marks3 : ")) # 97
# Math.max(marks1, marks2, marks3) : Max marks will be return

if marks1 > marks2 and marks1 > marks3 : # 'marks1' is a winner 🔥
    print(f"{marks1} is greater than {marks2} & {marks3}.")
elif marks2 > marks1 and marks2 > marks3 : # 'marks2' is a winner 🔥
    print(f"{marks2} is greater than {marks1} & {marks3}.")
else:  # 'marks3' is a winner 🔥
    print(f"{marks3} is greater than {marks1} & {marks2}.")
```

```
Enter Your Marks1 :  99
Enter Your Marks2 :  92
Enter Your Marks3 :  97
99 is greater than 92 & 97.
```

```python
# Find the max marks by comparing
marks1 = int(input("Enter Your Marks1 : ")) # 99
marks2 = int(input("Enter Your Marks2 : ")) # 92
marks3 = int(input("Enter Your Marks3 : ")) # 97
# Math.max(marks1, marks2, marks3) : Max marks will be return

if marks1 > marks2 and marks1 > marks3 : # 'marks1' is a winner 🔥
    print(f"{marks1} is greater than {marks2} & {marks3}.")
elif marks2 > marks1 and marks2 > marks3 : # 'marks2' is a winner 🔥
    print(f"{marks2} is greater than {marks1} & {marks3}.")
else:  # 'marks3' is a winner 🔥
    print(f"{marks3} is greater than {marks1} & {marks2}.")
```

```
Enter Your Marks1 :  92
Enter Your Marks2 :  99
Enter Your Marks3 :  97
99 is greater than 92 & 97.
```

```python
# Find the max marks by comparing
marks1 = int(input("Enter Your Marks1 : "))
marks2 = int(input("Enter Your Marks2 : "))
marks3 = int(input("Enter Your Marks3 : "))
# Math.max(marks1, marks2, marks3) : Max marks will be return

if marks1 > marks2 and marks1 > marks3 : # 'marks1' is a winner 🔥
    print(f"{marks1} is greater than {marks2} & {marks3}.")
elif marks2 > marks1 and marks2 > marks3 : # 'marks2' is a winner 🔥
    print(f"{marks2} is greater than {marks1} & {marks3}.")
else:  # 'marks3' is a winner 🔥
    print(f"{marks3} is greater than {marks1} & {marks2}.")
```

```
Enter Your Marks1 :  77
Enter Your Marks2 :  88
Enter Your Marks3 :  99
99 is greater than 77 & 88.
```

```python
# Find the min marks by comparing
marks1 = int(input("Enter Your Marks1 : "))
marks2 = int(input("Enter Your Marks2 : "))
marks3 = int(input("Enter Your Marks3 : "))
# Math.min(marks1, marks2, marks3) : min marks will be return

if marks1 < marks2 and marks1 < marks3 : # 'marks1' is minimum 🔥
    print(f"{marks1} is less than {marks2} & {marks3}.")
elif marks2 < marks1 and marks2 < marks3 : # 'marks2' is minimum 🔥
    print(f"{marks2} is less than {marks1} & {marks3}.")
else:  # 'marks3' is minimum 🔥
    print(f"{marks3} is less than {marks1} & {marks2}.")
```

```
Enter Your Marks1 :  33
Enter Your Marks2 :  55
Enter Your Marks3 :  29
29 is less than 33 & 55.
```

```
# Polymorphism : Poly [Many] + forms [🎭]
Math.min(marks1, marks2)
Math.min(marks1, marks2, marks3)
Math.min(marks1, marks2, marks3, marks4)
Math.min(marks1, marks2, marks3, marks4, marks5)
Math.min(marks1, marks2, marks3, marks4, marks5 ,.....)
```

```python
# ShortHand Conditional Statement
# if condition: statement
marks = int(input("Enter a valid marks: "))
if marks >= 90 : print("Excellent")
```

```
Enter a valid marks:  99
Excellent
```

```python
# if condition: statement
marks = int(input("Enter a valid marks: "))
if marks >= 90 : print("Excellent")
```

```
Enter a valid marks:  77
```

```python
# if-else shorthand
# result = value1 if condition else value2
marks = int(input("Enter a valid marks: "))
print("Pass") if marks>=33 else print("Fail")
```

```
Enter a valid marks:  91
Pass
```

```python
# if-else shorthand
# result = value1 if condition else value2
marks = int(input("Enter a valid marks: "))
print("Pass") if marks>=33 else print("Fail")
```

```
Enter a valid marks:  29
Fail
```

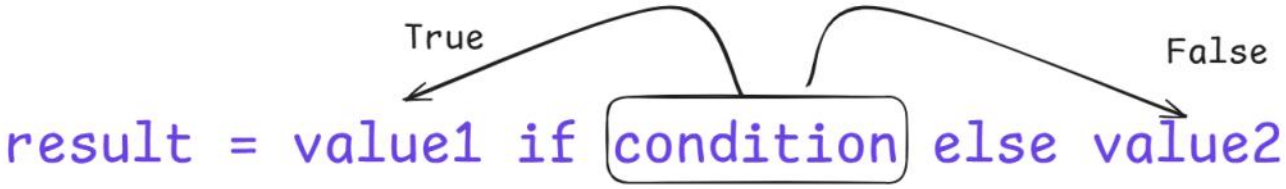
```
# if-else shorthand
# result = value1 if condition else value2
marks = int(input("Enter a valid marks: "))
print("Pass") if marks>=33 # SyntaxError: expected 'else' after 'if' expression
# if marks>=33 : print("Pass")
```

## Shorthand Conditional Statements :

if : if condition: statement

if-else : result = value1 if condition else value2

if-elif-else : result = (value1 if condition1 else value2) if condition2 else value3




```
# Comparing the Value ['ASCII']
val1 = 'Apple'
val2 = 'apple'
result = val1 if val1>val2 else val2 # val2 [False]
print(result) # 'apple'
```

```
apple
```

```
# Comparing the Value
val1 = 100
val2 = 100
result = val1 if val1 > val2 else val2 # val2 [False]
print(result) # 100
```

```
100
```

```python
# Comparing the Value
height1 = 170
height2 = 189
result = height2 if height2 > height1 else height1 # True
print(result) # height2 [189cm]
```

```
189
```

```python
val = int(input("Enter the value: "))
result = "Even" if val % 2 == 0  else "Odd"
print(result)
```

```
Enter the value:  77
Odd
```

```python
val = int(input("Enter the value: "))
result = "Even" if val % 2 == 0  else "Odd"
print(result)
```

```
Enter the value:  22
Even
```

```python
val = int(input("Enter the value: "))
if val % 2 == 0:
    print("Even")
else:
    print("Odd")
```

```
Enter the value:  99
Odd
```

```python
# if-elif-else :
# result = (value1 if condition1 else value2) if condition2 else value3
value1 = "Python"
value2 = "Java"
value3 = "JavaScript"
condition1 = True
condition2 = True
result = (value1 if condition1 else value2) if condition2 else value3
print(result) # 'Python'
```

```
Python
```

```python
# if-elif-else :
# result = (value1 if condition1 else value2) if condition2 else value3
value1 = "Python"
value2 = "Java"
value3 = "JavaScript"
condition1 = False
condition2 = True
result = (value1 if condition1 else value2) if condition2 else value3
print(result) # 'Java'
```

```
Java
```

```python
# if-elif-else :
# result = (value1 if condition1 else value2) if condition2 else value3
value1 = "Python"
value2 = "Java"
value3 = "JavaScript"
condition1 = True
condition2 = False
result = (value1 if condition1 else value2) if condition2 else value3
# value1 if condition2 else value3
print(result) # "JavaScript"
```

```
JavaScript
```

```python
# if-elif-else :
# result = (value1 if condition1 else value2) if condition2 else value3
value1 = "Python"
value2 = "Java"
value3 = "JavaScript"
condition1 = False
condition2 = False
result = (value1 if condition1 else value2) if condition2 else value3
print(result) # "JavaScript"
```

```
JavaScript
```

```python
# Grading System [Shorthand Property]
score = int(input("Enter a valid score: "))
grade = 'A' if score>=90 else ('B' if score>=80 else ('C' if score>=70 else ('D' if score>=50 else 'E')))
print(grade)
```

```
Enter a valid score:  77
C
```

```python
# Grading System [LongHand Property]
score = int(input("Enter a valid score: "))
if score >= 90:
    grade = 'A'
elif score >= 80:
    grade = 'B'
elif score >= 70:
    grade = 'C'
elif score >= 50:
    grade = 'D'
else:
    grade = 'E'
print(grade)
```

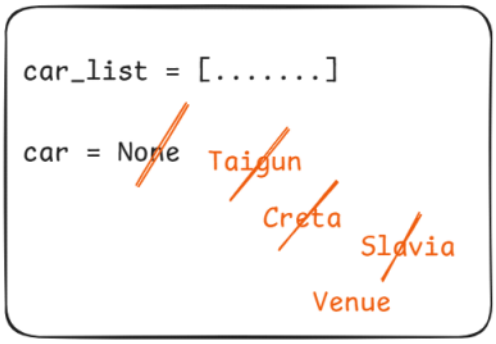
```
Enter a valid score:  77
C
```

## Looping Statement

```
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv',
            'Harrier','Safari','Lord Alto','Thar','Virtus','Defender',
            'Innova','Baleno','Legender','ScorpioN','Grand Vitara']
```

car

Memory




```
for car in car_list:
    print(car , end = " ")
```

Console:

```
Taigun Creta Slavia Venue
```

```
# Looping System
car_list = ['Taigun','Creta','Slavia','Venue','Sierra','City','Curv',
            'Harrier','Safari','Lord Alto','Thar','Virtus','Defender',
            'Innova','Baleno','Legender','ScorpioN','Grand Vitara']
# Manual Technique : Iterating on every element using indexing....
# What if we have to iterate from more then 1000s of elements in a list
print(car_list[0])
print(car_list[1])
print(car_list[2])
print(car_list[3])
print(car_list[4])
print(car_list[5])
print(car_list[6])
print(car_list[7])
print(car_list[8])
```

```
Taigun
Creta
Slavia
Venue
Sierra
City
Curv
Harrier
Safari
```

```python
# Loop [For , While]
for car in car_list:
    print(car , end = " ") # 'Taigun' 'Creta' 'Slavia' .....
```

```
Taigun Creta Slavia Venue Sierra City Curv Harrier Safari Lord Alto Thar Virtus Defender Innova Baleno Legende
r ScorpioN Grand Vitara
```

```python
# Loop [For , While]
for x in car_list:
    print(x , end = " ")
```

```
Taigun Creta Slavia Venue Sierra City Curv Harrier Safari Lord Alto Thar Virtus Defender Innova Baleno Legende
r ScorpioN Grand Vitara
```

```python
# Even and Odd Number
_list = [11,21,23,44,11,221,12,19,22,29,77,10,100,-50,500,51,False,True]
# False = 0
# True = 1
for val in _list:
    if val % 2 == 1: # Odd
        print(f"{val} is an Odd Element.")
    elif val % 2 == 0: # 'Even'
        print(f"{val} is an Even Element.")
```

```
11 is an Odd Element.
21 is an Odd Element.
23 is an Odd Element.
44 is an Even Element.
11 is an Odd Element.
221 is an Odd Element.
12 is an Even Element.
19 is an Odd Element.
22 is an Even Element.
29 is an Odd Element.
77 is an Odd Element.
10 is an Even Element.
100 is an Even Element.
-50 is an Even Element.
500 is an Even Element.
51 is an Odd Element.
False is an Even Element.
True is an Odd Element.
```

```python
# Even and Odd Number
_list = [11,21,23,44,11,221,12,19,22,29,77,10,100,-50,500,51,False,True]
# False = 0
# True = 1
# List -> .append() , .insert() , .extend()
even_list = []
odd_list = []
for val in _list:
    if val % 2 == 1: # Odd
        odd_list.append(val)
    elif val % 2 == 0: # 'Even'
        even_list.append(val)

print(even_list)
print(odd_list)
```

```
[44, 12, 22, 10, 100, -50, 500, False]
[11, 21, 23, 11, 221, 19, 29, 77, 51, True]
```

```python
# Even and Odd Number
_list = [11,21,23,44,11,221,12,19,22,29,77,10,100,-50,500,51,False,True]
# False = 0
# True = 1
# List -> .append() , .insert() , .extend()
# Set -> .add() , .update()
# Dictionary -> .setdefault(), .update()
even_set = set()
odd_set = set()
for val in _list:
    if val % 2 == 1: # Odd
        odd_set.add(val)
    elif val % 2 == 0: # 'Even'
        even_set.add(val)

print(even_set)
print(odd_set) # No 11 as duplicates allowed, Also final set will be shuffled
```

```
{False, 100, 10, 44, 12, -50, 500, 22}
{True, 11, 77, 29, 19, 51, 21, 23, 221}
```

```python
num_list = [1,2,3,4,5]
_mult = 1
_sum = 0
for val in num_list:
    _mult *= val # _mult = _mult * val
    _sum += val # _sum = _sum + val

print(_mult) # 120
print(_sum) # 15
```

```
120
15
```

```
min = +inf [1st element]
max = -inf [1st element]
```