

B18/25

Task 2: Implement Conditional, Control and looping statements

Aim: To implement conditional, control and looping statements using python

2.1 you are developing a simple grade management system for a school. The system needs to determine the grade of a student based on their score in a test. The grading system follows these rules:

If the score is 90 or above, the grade is "A".

If the score is between 80 and 89, the grade is "B".

If the score is between 70 and 79, the grade is "C".

If the score is between 60 and 69, the grade is "D".

If the score is below 60, the grade is "F".

Algorithm:

1) start

2) Get the input mark from the user

3) with the use of an if-elif-else statement do

* If the marks ≥ 90 print grade "A"

* If the marks is between 80 and 89 print grade "B".

* If the marks is between 70 and 79 print grade "C".

* If the marks is between 60 and 69 print grade "D".

* If the mark is below 60, print grade "F".

4) Stop

Program

```
Score = int(input("Enter the score:"))
```

```
if Score  $\geq$  90:
```

```
    print("The Grade is A")
```

```
elif (Score  $\leq$  89 and Score  $\geq$  80):
```

```
    print("The Grade is B")
```

```
elif (Score  $\leq$  79 and Score  $\geq$  70):
```

```
    print("The Grade is C")
```

```
elif (Score  $\leq$  69 and Score  $\geq$  60):
```

```
    print("The Grade is D")
```

```
else:
```

```
    print("The Grade is F")
```

Result: Thus implement conditional control and looping statement using python

Output:

Enter the score: 60

The Grade is D

DATE	SCORE
1	60
2	65
3	70
4	75
5	80
6	85
7	90
8	95
9	100
10	105
11	110
12	115
13	120
14	125
15	130
16	135
17	140
18	145
19	150
20	155
21	160
22	165
23	170
24	175
25	180
26	185
27	190
28	195
29	200
30	205
31	210
32	215
33	220
34	225
35	230
36	235
37	240
38	245
39	250
40	255
41	260
42	265
43	270
44	275
45	280
46	285
47	290
48	295
49	300
50	305
51	310
52	315
53	320
54	325
55	330
56	335
57	340
58	345
59	350
60	355
61	360
62	365
63	370
64	375
65	380
66	385
67	390
68	395
69	400
70	405
71	410
72	415
73	420
74	425
75	430
76	435
77	440
78	445
79	450
80	455
81	460
82	465
83	470
84	475
85	480
86	485
87	490
88	495
89	500
90	505
91	510
92	515
93	520
94	525
95	530
96	535
97	540
98	545
99	550
100	555

2.2 The electronics maintenance team at a data center needs a tool to assess the health status of UPS backup batteries based on their current charge percentage as input and categorizes the battery health using the following conditions:

- * If the percentage is greater than (or) equal to 90, display:
⇒ "Excellent Battery Health"
- * If the percentage is between 70 and 89, display:
⇒ "Good Battery Health"
- * If the percentage is between 40 and 69, display:
⇒ "Average Battery Health"
- * If the percentage is below 40, display:
⇒ "Poor Battery Health"

Task: Write a python program that uses Ladderized if-elif-else statements

Algorithm:

- 1) Accept battery percentage from the user.
- 2) Use Ladderized if-elif-else to determine the health category:

- * If percentage $\geq 90 \rightarrow$ "Excellent Battery Health".
- * If $70 \leq \text{percentage} < 90 \rightarrow$ "Good Battery Health".
- * If $40 \leq \text{percentage} < 70 \rightarrow$ "Average Battery Health".
- * If percentage $< 40 \rightarrow$ "Poor Battery Health".

Program:

```
# Battery Health checker
percentage = int(input("Enter battery percentage"))
if percentage >= 90:
    print("Excellent Battery Health")
elif percentage >= 70:
    print("Good Battery Health")
elif percentage >= 40:
    print("Average Battery Health")
else:
    print("Poor Battery Health")
```

Result: Thus the electronics maintenance team at a data center needs a tool to assess the health status of UPS backup batteries based on their current.

Test 2: Implement conditional, control and looping statements
 To implement conditional, control and looping statements

You are developing a simple grade management system for a school. The system needs to determine the grade of a student based on their score in a test. The grading system follows these rules:

If the score is 90 or above, the grade is "A".
 If the score is between 80 and 89, the grade is "B".
 If the score is between 70 and 79, the grade is "C".
 If the score is between 60 and 69, the grade is "D".
 If the score is below 60, the grade is "F".

Enter battery percentage: 85

Good Battery Health

1) Start
 2) Get the input mark from the user
 3) If the mark is greater than or equal to 90, the grade is "A".
 * If the mark is greater than or equal to 80 and less than 90, the grade is "B".
 * If the mark is greater than or equal to 70 and less than 80, the grade is "C".
 * If the mark is greater than or equal to 60 and less than 70, the grade is "D".
 * If the mark is less than 60, the grade is "F".

2nd step
 Program
 Score - int input ("Enter the score: ")
 if (score >= 90)
 print ("The grade is A")
 else if (score >= 80 and score < 90)
 print ("The grade is B")
 else if (score >= 70 and score < 80)
 print ("The grade is C")
 else if (score >= 60 and score < 70)
 print ("The grade is D")
 else
 print ("The grade is F")

Test 3: Implement conditional, control and looping statements

Sample Input:

Enter height of visitor 1 in cm: 150

Enter height of visitor 2 in cm: 110

Enter height of visitor 3 in cm: 100

Enter height of visitor 4 in cm: 90

Enter height of visitor 5 in cm: 120

Sample Output:

Allowed

Not Allowed

Allowed

Not Allowed

Allowed

2.3 You're coding a system at an amusement park that checks the height of each visitor.

* If the height is 120cm or more, print "Allowed".

* Otherwise, print "Not allowed".

Repeat this for 5 visitors.

Algorithm:

1) Start the program

2) Set the total number of visitors to 5

3) Loop from visitor 1 to visitor 5:

* Accept the height of the visitor as input (in cm)

* If height is greater than or equal to 120, print "Allowed".

* else, print "Not allowed".

4) End the loop after 5 visitors have been checked

5) Stop the program

Program

```
for i in range(1,6):
```

```
    height = int(input(f"Enter height of visitor {i} in cm:"))
```


```
    if height >= 120:
```

```
        print("Allowed to ride")
```

```
    else
```

```
        print("Not allowed to ride")
```

VEL TECH	
EX NO.	2
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	20
SIGN WITH DATE	

Result: 

Thus, Python program was successfully implemented using conditional statements (if-else) control flow and looping statements.