Decision Making Statements (if, if-else, if-elif-else)

Statements	Syntax	Example	Definition

if	if condition:
	statement1
	statement2
		i = 10
		if (i > 15):
		print ("10 is less than 15")
		print ("I am Not in if")
		
		Output:
		`I am Not in if`
			if statement is the most simple decision making statement. It is used to decide whether a certain statement or block of statements will be executed or not

If - else	if (condition):
	statement1
	else:
	statement2
		i = 20;
		if (i < 15):
		print ("i is smaller than 15")
		print ("i'm in if Block")
		else:
		print ("i is greater than 15")
		print ("i'm in else Block")
		print ("i'm not in if and not in else Block")
		
		Output:
		i is greater than 15
		i'm in else Block
		i'm not in if and not in else Block
			We can use the else statement with if statement to execute a block of code when the condition is false.

nested-if	if (condition1):
	statement
	if (condition2):
	statement
	# if Block is end here
	# if Block is end here
		i = 10
		if (i == 10):
		if (i < 15):
		print ("i is smaller than 15")
		if (i < 12):
		print ("i is smaller than 12 too")
		else:
		print ("i is greater than 15")
		
		Output:
		i is smaller than 15
		i is smaller than 12 too
			A nested if is an if statement that is the target of another if statement. Nested if statements means an if statement inside another if statement.

if-elif-else	if (condition):
	  statement
	elif (condition):
	  statement
	  .
	  .
	else:
	  statement
		i = 20
		if (i == 10):
		  print ("i is 10")
		elif (i == 15):
		  print ("i is 15")
		elif (i == 20):
		  print ("i is 20")
		else:
		  print ("i is not present")
		
		Output:
		i is 20
			Here, a user can decide among multiple options. The if statements are executed from the top down. As soon as one of the conditions controlling the if is true, the statement associated with that if is executed, and the rest of the ladder is bypassed.

1. Predict the output:

```
a, b = 1, 2
if a==1:
        if b==2:
            print("a is", a, "and b is", b)
print("End of the program")
```

```
output: a is 1 and b is 2
End of the program
```

2. For the given below program:

```
print("Enter the Number: ")
num = int(input())
R = num%2
if num%2==0:
    print("Bennett")
    print(R)
else:
    print("University")
    print(R)
```

if the input is -13, then what will be the output

```
output: University
        1
```

3. Predict the output:

```
#Let's Play a game!!

First_Person = int(input("Enter a number between 1 and 10: "))
Second_Person = int(input("Enter a number between 1 and 10: "))
if (First_Person >= 1) and (First_Person <= 10):
    if (Second_Person >= 1) and (Second_Person <= 10):
        print("Your secret number is: ", First_Person * Second_Person)
    else:
        print("Second person entered value is out of range!")
else:
    print("First person entered value is out of range!")
```

if `First_Person entered` input is -1 and `Second_Person entered` input is 11, then what will be the output

output: `First person entered value is out of range!`

4. Write an algorithm that perform simple grading scheme according to the given below table. Now, accept only valid inputs for our grade conversion script and displays an error message otherwise. That is, if input is greater than 100 or less than 0, then display error message.

LETTER GRADE S	RANGE OF NUMERIC GRADE
A+	All grades above 95
A	All grades above 90 and below 96
B	All grades above 80 and below 91
C	All grades above 70 and below 81
D	All grades above 60 and below 71
E	All grades above 50 and below 61
F	All grades below 51

Solution:

5. Enter the number, check whether the given input is divisible by 2 and 4 both.

Use: bitwise operator, right/left shift operation

Solution:

```
n = int(input("enter the number: ", ))
if (((n >> 2) << 2) == n) and (((n >> 1) << 1) == n):
              print("Number is divisible by 2 and 4 both")
else:
    print("Not divisible by 2 and 4")
```

6. Enter three values as a=10, b=20 and c=30 and calculate the average of these three numbers. Then, check the following conditions:

I. If the average value is greater than the individual value
II. If the average value is greater than a and b
III. If the average value is greater than a and c
IV. If the average value is greater than b and c

Solution:

```
a=10
b=20
c=30
avg=(a+b+c)/3
print("avg = ", avg)
if avg>a and avg>b and avg>c:
        print("Average is greater than a, b and c")
elif avg>a and avg>b:
        print("Average is greater than a and b")
elif avg>a and avg>c:
        print("Average is greater than a and c")
elif avg>b and avg>c:
        print("Average is greater than b and c")
else:
        print("Conditions are not satisfied")
```

7. Find the output of the following:

a)

```
a=20
a*=5!=5 or 6>=10>>2
```

```
print(a)
Output: 20
```

b)

```
a=10
a/=5!=5 and 6>=100>>2
print(a)
Traceback (most recent call last):
File "<string>", line 2, in <module>
ZeroDivisionError: division by zero
```

c)

```
a=50<=55 and 6>=10 and 2
print(a)
Output: False
```

d)

```
a=20
a = 50 or 55 and 6<=10 and 20%2
print(a)
Output: 50
```

e)

```
a=20
a = int(5^5 or 55 and 6>=10 and 2*2)
print(a)
Output: 0
```

8. What will be the output of the following programs:

a)

```
Thislist = ["Apple", "Banna", "Cherry"]
mylist = Thislist.copy()
print(Thislist)
print(mylist)
Output: ['Apple', 'Banna', 'Cherry']
        ['Apple', 'Banna', 'Cherry']
```

b)

```
Thislist = ['B', 'E', 'N', 'N', 'E','T','T']
print("Initial List: ", Thislist)
Sliced_Thislist = Thislist[:5]
print(Sliced_Thislist)
Output: ['B', 'E', 'N', 'N', 'E']
```

c)

```
Thislist = ['B', 'E', 'N', 'N', 'E','T','T']
print("Initial List: ", Thislist)
Sliced_Thislist = Thislist[:-5]
print(Sliced_Thislist)
output: ['B', 'E']
```

d)

```
lst=[[1,2,3],'hello',[3,4,5,6]]
print(lst[1][1])
print(lst[2][1])
print(lst[1])
print(lst[0])
lst[2][:2]=[1,1]
print(lst)
print(lst[0][1])
output: e
        4
        hello
        [1, 2, 3]
        [[1, 2, 3], 'hello', [1, 1, 5, 6]]
        2
```

9. What will be output of the following statements?

```
a = 10
b = 4
print(a ^ b)
print(a >> 2)
print(a << 2)
print(a & b)
print(a | b)
print(~a)


output: 14
        2
        40
        0
        14
        -11
```

Explain all the output.