



Inspiring Excellence

<b>Course Code:</b>	<b>CSE111</b>
<b>Course Title:</b>	<b>Programming Language II</b>
<b>Lab No:</b>	<b>09</b>
<b>Topic:</b>	<b>OOP (Class variable and class method)</b>
<b>Number of tasks:</b>	<b>5</b>

## Task 1

We know that Nike is opening their official outlets in Bangladesh. So let's construct a NikeBangladesh class so that they can keep track of their inventory and sales here,

### Hint:

**productSold()/restockProducts():** takes in a dictionary with product name and quantity, and updates the instance and class variables accordingly

Driver Code	Output
<pre>print("xxxxxxxxxxxxxxxx1xxxxxxxxxxxxxxxx") NikeBangladesh.status() dhaka = NikeBangladesh("Dhaka Banani") chittagong = NikeBangladesh("Chittagong GEC") print("xxxxxxxxxxxxxxxx2xxxxxxxxxxxxxxxx") dhaka.details() print("xxxxxxxxxxxxxxxx3xxxxxxxxxxxxxxxx") chittagong.details() print("xxxxxxxxxxxxxxxx4xxxxxxxxxxxxxxxx") dhaka.restockProducts( {"Air Jordan":1200,"Cortez":200,"Zoom Kobe":200}) chittagong.restockProducts( {"Air Jordan":1000,"Cortez":250,"Zoom Kobe":100}) print("xxxxxxxxxxxxxxxx5xxxxxxxxxxxxxxxx") NikeBangladesh.status() print("xxxxxxxxxxxxxxxx6xxxxxxxxxxxxxxxx") dhaka.productSold({"Air Jordan":760,"Cortez":90}) chittagong.productSold({"Air Jordan":520,"Zoom Kobe":70}) print("xxxxxxxxxxxxxxxx7xxxxxxxxxxxxxxxx") NikeBangladesh.status()</pre>	<pre>xxxxxxxxxxxxxxxx1xxxxxxxxxxxxxxxx Nike Bangladesh Status: Branches Opened: [] Currently Stocked {'Air Jordan': 0, 'Cortez': 0, 'Zoom Kobe': 0} Sold: 0 xxxxxxxxxxxxxxxx2xxxxxxxxxxxxxxxx Nike Dhaka Banani outlet: Products Currently Stocked: {'Air Jordan': 0, 'Cortez': 0, 'Zoom Kobe': 0} Sold: 0 xxxxxxxxxxxxxxxx3xxxxxxxxxxxxxxxx Nike Chittagong GEC outlet: Products Currently Stocked: {'Air Jordan': 0, 'Cortez': 0, 'Zoom Kobe': 0} Sold: 0 xxxxxxxxxxxxxxxx4xxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxx5xxxxxxxxxxxxxxxx Nike Bangladesh Status: Branches Opened: ['Dhaka Banani', 'Chittagong GEC'] Currently Stocked {'Air Jordan': 2200, 'Cortez': 450, 'Zoom Kobe': 300} Sold: 0 xxxxxxxxxxxxxxxx6xxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxx7xxxxxxxxxxxxxxxx Nike Bangladesh Status: Branches Opened: ['Dhaka Banani', 'Chittagong GEC'] Currently Stocked {'Air Jordan': 920, 'Cortez': 360, 'Zoom Kobe': 230} Sold: 1440</pre>

## Task 2

Write the **Player** class so that the given code provides the expected output.

[You are not allowed to change the code below]

*# Write your code here*

```
print("Total number of players:", Player.total)
print("-----")
p1 = Player()
p1.set_name("Neymar")
p1.set_team("Brazil")
print(p1.player_detail())
print('=====')
Player.info()
print("-----")
p2 = Player("Ronaldo")
p2.set_number(7)
p2.set_team("Portugal")
print(p2.player_detail())
print('=====')
Player.info()
print("-----")
p3 = Player("Messi")
p3.set_team("Argentina")
print(p3.player_detail())
print('=====')
Player.info()
print("-----")
p4 = Player("Mbappe", 10, "France")
print(p4.player_detail())
print('=====')
Player.info()
```

*Output:*

Total number of players: 0

-----

Player Name: Neymar

Jersey Number: 10

Country: Brazil

=====

Total number of players: 1

Players enlisted so far: Neymar

-----

Player Name: Ronaldo

Jersey Number: 7

Country: Portugal

=====

Total number of players: 2

Players enlisted so far: Neymar, Ronaldo

-----

Player Name: Messi

Jersey Number: 10

Country: Argentina

=====

Total number of players: 3

Players enlisted so far: Neymar, Ronaldo, Messi

-----

Player Name: Mbappe

Jersey Number: 10

Country: France

=====

Total number of players: 4

Players enlisted so far: Neymar, Ronaldo, Messi, Mbappe

## Task 3

Write the **SultansDine** class so that the given code provides the expected output.

[You are not allowed to change the code below]

<pre># Write your code here  SultansDine.details() print('#####') dhanmondi = SultansDine('Dhanmondi') dhanmondi.sellQuantity(25) dhanmondi.branchInformation() print('-----') SultansDine.details()  print('=====')  baily_road = SultansDine('Baily Road') baily_road.sellQuantity(15) baily_road.branchInformation() print('-----') SultansDine.details()  print('=====')  gulshan = SultansDine('Gulshan') gulshan.sellQuantity(9) gulshan.branchInformation() print('-----') SultansDine.details()</pre>	<p><b>Output:</b></p> <p>Total Number of branch(s): 0 Total Sell: 0 Taka ##### Branch Name: Dhanmondi Branch Sell: 10000 Taka ----- Total Number of branch(s): 1 Total Sell: 10000 Taka Branch Name: Dhanmondi, Branch Sell: 10000 Taka Branch consists of total sell's: 100.00% =====</p> <p>Branch Name: Baily Road Branch Sell: 5250 Taka ----- Total Number of branch(s): 2 Total Sell: 15250 Taka Branch Name: Dhanmondi, Branch Sell: 10000 Taka Branch consists of total sell's: 65.57% Branch Name: Baily Road, Branch Sell: 5250 Taka Branch consists of total sell's: 34.43% =====</p> <p>Branch Name: Gulshan Branch Sell: 2700 Taka ----- Total Number of branch(s): 3 Total Sell: 17950 Taka Branch Name: Dhanmondi, Branch Sell: 10000 Taka Branch consists of total sell's: 55.71% Branch Name: Baily Road, Branch Sell: 5250 Taka Branch consists of total sell's: 29.25% Branch Name: Gulshan, Branch Sell: 2700 Taka Branch consists of total sell's: 15.04%</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Subtasks:

1. Create **SultansDine** class
2. Create 2 class variable and 1 class list
3. Create 1 class method
4. Calculation of branch sell is given below

- a. If sellQuantity < 10:
    - i. Branch\_sell = quantity \* 300
  - b. Else if sellQuantity < 20:
    - i. Branch\_sell = quantity \* 350
  - c. Else
    - i. Branch\_sell = quantity \* 400
5. Calculation of branch's sell percentage = (branch's sell / total sell) \* 100

## Task 4

1	<code>class FinalT6A:</code>
2	<code>temp = 3</code>
4	<code>def __init__(self, x, p):</code>
5	<code>self.sum, self.y = 0, 2</code>
6	<code>FinalT6A.temp += 3</code>
7	<code>self.y = self.temp - p</code>
8	<code>self.sum = self.temp + x</code>
9	<code>print(x, self.y, self.sum)</code>
11	<code>def methodA(self):</code>
12	<code>x, y = 0, 0</code>
13	<code>y = y + self.y</code>
14	<code>x = self.y + 2 + self.temp</code>
15	<code>self.sum = x + y + self.methodB(self.temp, y)</code>
16	<code>print(x, y, self.sum)</code>
18	<code>def methodB(self, temp, n):</code>
19	<code>x = 0</code>
20	<code>FinalT6A.temp += 1</code>

21	<code>self.y = self.y + (FinalT6A.temp)</code>
22	<code>FinalT6A.temp -= 1</code>
23	<code>x = x + 2 + n</code>
24	<code>self.sum = self.sum + x + self.y</code>
25	<code>print(x, self.y, self.sum)</code>
26	<code>return self.sum</code>

<pre>q1 = FinalT6A(2,1) q1.methodA() q1.methodA()</pre>	x	y	sum

## Task 5

1	<code>class msgClass:</code>
2	<code>def __init__(self):</code>
3	<code>self.content = 0</code>
4	
5	<code>class Quiz3:</code>
6	<code>x = 0</code>
7	<code>def __init__(self, k = None):</code>
8	<code>self.sum, self.y = 0, 0</code>
9	<code>if k is None:</code>
10	<code>self.sum = 5</code>
11	<code>Quiz3.x = 2</code>

12	<code>self.y = 2</code>
13	<code>else:</code>
14	<code>self.sum = self.sum + k</code>
15	<code>self.y = 3</code>
16	<code>Quiz3.x += 2</code>
17	<code>def methodA(self):</code>
18	<code>    x = 1</code>
19	<code>    y = 1</code>
20	<code>    msg = [None]</code>
21	<code>    myMsg = msgClass()</code>
22	<code>    myMsg.content = Quiz3.x</code>
23	<code>    msg[0] = myMsg</code>
24	<code>    msg[0].content = self.y + myMsg.content</code>
25	<code>    self.y = self.y + self.methodB(msg[0])</code>
26	<code>    y = self.methodB(msg[0]) + self.y</code>
27	<code>    x = y + self.methodB(msg, msg[0])</code>
28	<code>    self.sum = x + y + msg[0].content</code>
29	<code>    print(x, y, self.sum)</code>
30	<code>def methodB(self, *args):</code>
31	<code>    if len(args) == 2:</code>
32	<code>        mg2, mg1 = args</code>
33	<code>        x = 2</code>
34	<code>        self.y = self.y + mg2[0].content</code>
35	<code>        mg2[0].content = self.y + mg1.content</code>

36	<code>x = x + 2 + mg1.content</code>
37	<code>self.sum = self.sum + x + self.y</code>
38	<code>mg1.content = self.sum - mg2[0].content</code>
39	<code>print(Quiz3.x, self.y, self.sum)</code>
40	<code>return self.sum</code>
41	
42	<code>elif len(args) == 1:</code>
43	<code>mg1, = args</code>
44	<code>x = 1</code>
45	<code>y = 2</code>
46	<code>y = self.sum + mg1.content</code>
47	<code>self.y = y + mg1.content</code>
48	<code>x = Quiz3.x + 5 + mg1.content</code>
49	<code>self.sum = self.sum + x + y</code>
50	<code>Quiz3.x = mg1.content + x + 3</code>
51	<code>print(x, y, self.sum)</code>
52	<code>return y</code>

<code>a1 = Quiz3()</code>	<b>x</b>	<b>y</b>	<b>sum</b>
<code>a2 = Quiz3(5)</code>			
<code>msg = msgClass()</code>			
<code>a1.methodA()</code>			
<code>a2.methodB(msg)</code>			