

| **Course Code:** | **CSE111** |
| --- | --- |
| **Course Title:** | **Programming Language II** |
| **Lab No:** | **09** |
| **Topic:** | **OOP (Class variable and class method)** |
| **Number of tasks:** | **5** |

**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** |
| --- | --- |
| print("xxxxxxxxxxxxxx1xxxxxxxxxxxxxxxx")  NikeBangladesh.status()  dhaka = NikeBangladesh("Dhaka Banani")  chittagong = NikeBangladesh("Chittagong GEC")  print("xxxxxxxxxxxxxx2xxxxxxxxxxxxxxxx")  dhaka.details()  print("xxxxxxxxxxxxxx3xxxxxxxxxxxxxxxx")  chittagong.details()  print("xxxxxxxxxxxxxx4xxxxxxxxxxxxxxxx")  dhaka.restockProducts(  {"Air Jordan":1200,"Cortez":200,"Zoom Kobe":200})  chittagong.restockProducts(  {"Air Jordan":1000,"Cortez":250,"Zoom Kobe":100})  print("xxxxxxxxxxxxxx5xxxxxxxxxxxxxxxx")  NikeBangladesh.status()  print("xxxxxxxxxxxxxx6xxxxxxxxxxxxxxxx")  dhaka.productSold({"Air Jordan":760,"Cortez":90})  chittagong.productSold({"Air Jordan":520,"Zoom Kobe":70})  print("xxxxxxxxxxxxxx7xxxxxxxxxxxxxxxx")  NikeBangladesh.status() | xxxxxxxxxxxxxx1xxxxxxxxxxxxxxxx  Nike Bangladesh Status:  Branches Opened: []  Currently Stocked  {'Air Jordan': 0, 'Cortez': 0, 'Zoom Kobe': 0}  Sold: 0  xxxxxxxxxxxxxx2xxxxxxxxxxxxxxxx  Nike Dhaka Banani outlet:  Products Currently Stocked:  {'Air Jordan': 0, 'Cortez': 0, 'Zoom Kobe': 0}  Sold: 0  xxxxxxxxxxxxxx3xxxxxxxxxxxxxxxx  Nike Chittagong GEC outlet:  Products Currently Stocked:  {'Air Jordan': 0, 'Cortez': 0, 'Zoom Kobe': 0}  Sold: 0  xxxxxxxxxxxxxx4xxxxxxxxxxxxxxxx  xxxxxxxxxxxxxx5xxxxxxxxxxxxxxxx  Nike Bangladesh Status:  Branches Opened: ['Dhaka Banani', 'Chittagong GEC']  Currently Stocked  {'Air Jordan': 2200, 'Cortez': 450, 'Zoom Kobe': 300}  Sold: 0  xxxxxxxxxxxxxx6xxxxxxxxxxxxxxxx  xxxxxxxxxxxxxx7xxxxxxxxxxxxxxxx  Nike Bangladesh Status:  Branches Opened: ['Dhaka Banani', 'Chittagong GEC']  Currently Stocked  {'Air Jordan': 920, 'Cortez': 360, 'Zoom Kobe': 230}  Sold: 1440 |

**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]**

| ***# 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() | ***Output:***  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%  ================================  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%  ================================  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% |
| --- | --- |

**Subtaks:**

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
   1. If sellQuantity < 10:
      1. Branch\_sell = quantity \* 300
   2. Else if sellQuantity < 20:
      1. Branch\_sell = quantity \* 350
   3. Else
      1. Branch\_sell = quantity \* 400
5. Calculation of branch’s sell percentage = (branch’s sell / total sell) \* 100

**Task 4**

| **1** | **class FinalT6A:** |
| --- | --- |
| **2** | **temp = 3** |
| **4** | **def \_\_init\_\_(self, x, p):** |
| **5** | **self.sum, self.y = 0, 2** |
| **6** | **FinalT6A.temp += 3** |
| **7** | **self.y = self.temp - p** |
| **8** | **self.sum = self.temp + x** |
| **9** | **print(x, self.y, self.sum)** |
| **11** | **def methodA(self):** |
| **12** | **x, y = 0, 0** |
| **13** | **y = y + self.y** |
| **14** | **x = self.y + 2 + self.temp** |
| **15** | **self.sum = x + y + self.methodB(self.temp, y)** |
| **16** | **print(x, y, self.sum)** |
| **18** | **def methodB(self, temp, n):** |
| **19** | **x = 0** |
| **20** | **FinalT6A.temp += 1** |
| **21** | **self.y = self.y + (FinalT6A.temp)** |
| **22** | **FinalT6A.temp -= 1** |
| **23** | **x = x + 2 + n** |
| **24** | **self.sum = self.sum + x + self.y** |
| **25** | **print(x, self.y, self.sum)** |
| **26** | **return self.sum** |

| **q1 = FinalT6A(2,1)**  **q1.methodA()**  **q1.methodA()** | **x** | **y** | **sum** |
| --- | --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

**Task 5**

| **1** | **class msgClass:** |
| --- | --- |
| **2** | **def \_\_init\_\_(self):** |
| **3** | **self.content = 0** |
| **4** |  |
| **5** | **class Quiz3:** |
| **6** | **x = 0** |
| **7** | **def \_\_init\_\_(self, k = None):** |
| **8** | **self.sum, self.y = 0, 0** |
| **9** | **if k is None:** |
| **10** | **self.sum = 5** |
| **11** | **Quiz3.x = 2** |
| **12** | **self.y = 2** |
| **13** | **else:** |
| **14** | **self.sum = self.sum + k** |
| **15** | **self.y = 3** |
| **16** | **Quiz3.x += 2** |
| **17** | **def methodA(self):** |
| **18** | **x = 1** |
| **19** | **y = 1** |
| **20** | **msg = [None]** |
| **21** | **myMsg = msgClass()** |
| **22** | **myMsg.content = Quiz3.x** |
| **23** | **msg[0] = myMsg** |
| **24** | **msg[0].content = self.y + myMsg.content** |
| **25** | **self.y = self.y + self.methodB(msg[0])** |
| **26** | **y = self.methodB(msg[0]) + self.y** |
| **27** | **x = y + self.methodB(msg, msg[0])** |
| **28** | **self.sum = x + y + msg[0].content** |
| **29** | **print(x, y, self.sum)** |
| **30** | **def methodB(self, \*args):** |
| **31** | **if len(args) == 2:** |
| **32** | **mg2, mg1 = args** |
| **33** | **x = 2** |
| **34** | **self.y = self.y + mg2[0].content** |
| **35** | **mg2[0].content = self.y + mg1.content** |
| **36** | **x = x + 2 + mg1.content** |
| **37** | **self.sum = self.sum + x + self.y** |
| **38** | **mg1.content = self.sum - mg2[0].content** |
| **39** | **print(Quiz3.x, self.y, self.sum)** |
| **40** | **return self.sum** |
| **41** |  |
| **42** | **elif len(args) == 1:** |
| **43** | **mg1, = args** |
| **44** | **x = 1** |
| **45** | **y = 2** |
| **46** | **y = self.sum + mg1.content** |
| **47** | **self.y = y + mg1.content** |
| **48** | **x = Quiz3.x + 5 + mg1.content** |
| **49** | **self.sum = self.sum + x + y** |
| **50** | **Quiz3.x = mg1.content + x + 3** |
| **51** | **print(x, y, self.sum)** |
| **52** | **return y** |

| **a1 = Quiz3()**  **a2 = Quiz3(5)**  **msg = msgClass()**  **a1.methodA()**  **a2.methodB(msg)** | **x** | **y** | **sum** |
| --- | --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |