assesment4

June 7, 2023

```
[ ]: #Q1
```

```
[2]: my_list=[('sachin Tendular',34357),('Ricky Pointing',27483),('Jack␣
      ↪kallis',25534),('Virat Kohil',24936)]
```

```
[5]: sorted_data = sorted(my_list,key=lambda x:x[1],reverse=True)
     print(sorted_data)
```

```
[('sachin Tendular', 34357), ('Ricky Pointing', 27483), ('Jack kallis', 25534),
('Virat Kohil', 24936)]
```

```
[ ]: #Q2
```

```
[6]: my_list=[1,2,3,4,5,6,7,8,9,10]
```

```
[8]: list(map(lambda x : x**2,my_list))
```

```
[8]: [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

```
[ ]: #Q3
```

```
[9]: my_string=[1,2,3,4,5,6,7,8,9,10]
```

```
[15]: tuple(map(lambda x : str(x) ,my_string))
```

```
[15]: ('1', '2', '3', '4', '5', '6', '7', '8', '9', '10')
```

```
[ ]: # Q4
      "we need to invite the reduce function as it is not a in built function like␣
      ↪other so import the function from the libary"
```

```
[17]: from functools import reduce
```

```
[22]: my_list=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25]
```

```
[26]: (reduce(lambda x,y : x+y ,my_list))
```

```
[26]: 325

[ ]: #Q5

[27]: my_list=[2,3,6,9,27,60,90,120,55,46]

[40]: list(filter(lambda x :  x % 2== 0 and x % 3==0 ,my_list))

[40]: [6, 60, 90, 120]

[ ]: #Q6

[41]: my_plaindromes=['python','php','aba','radar','level']

[42]: list(filter(lambda x: x==x[::-1],my_plaindromes))

[42]: ['php', 'aba', 'radar', 'level']

[ ]: # as we got this output because to find plaindrome if we write the words from␣
      ↪front to back or back to front it should be same we can find this by slicing␣
      ↪operation

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```