

# پاسخ سوال برنامه نویسی

دوره استادی پایتون درسمن

پایتون پیشرفته



## پاسخ تمرین شماره ۱:

```
1 class Counter:
2     def __init__(self, start, end):
3         self.start = start
4         self.end = end
5
6     def __iter__(self):
7         return self
8
9     def __next__(self):
10        self.start += 3
11        if self.start > self.end:
12            raise StopIteration
13        return self.start
14
15 counter1 = Counter(10,100)
16 for number in counter1:
17     print(number)
```



## پاسخ تمرین شماره ۲:

```
9 def remove_Number(func):
10     def inner(*args,**kwargs):
11         number_List1 = list(*args)
12         number_List2 = []
13         for number in number_List1:
14             if type(number) == int and number > 0:
15                 number_List2.append(number)
16         func(number_List2)
17     return inner
18
19 @remove_Number
20 def my_Factorial(number_List):
21     number_List_New = []
22     for number in number_List:
23         multi = 1
24         for i in range(number,0,-1):
25             multi*=i
26         number_List_New.append(multi)
27     print(number_List_New)
28
29 number_List = [4,3,8,0,-3,-45,2,10,-16,23,9,1,-6,55,3.4,6,11.5]
30 my_Factorial(number_List)
```



## پاسخ تمرین شماره ۳:

```
112 from datetime import date,datetime,time,timedelta
113 from khayyam import *
114
115 def jalali_To_Gregorian_Converter1():
116     date1 = input("Enter Date (For Example: 1399-06-20):")
117     date2 = datetime.strptime(date1,"%Y-%m-%d")
118     date3 = JalaliDate(date2.year,date2.month,date2.day)
119     gregorian_Date = date3.to_date()
120     gregorian_Date1 = date.today()
121     days = abs(gregorian_Date-gregorian_Date1).total_seconds()/(24*60*60)
122     yield gregorian_Date
123     yield days
124
125 final_List = list(jalali_To_Gregorian_Converter1())
126 for item in final_List:
127     print(item)
128     print(50*"")
```



## پاسخ تمرین شماره ۴:

```

1 import operator
2 # -----
3 # example
4 str = """
5 Today, Richard Rael and Tony Riggs tell the story of American astronomer edwin hubble. He changed our
6 ideas about the universe and how it developed.
7
8 Edwin hubble made his most important discoveries in the nineteen twenties. Today, other astronomers
9 continue the work he began. Many of them are using the Hubble space telescope that is named after him.
10
11 Edwin Hubble was born in eighteen eighty-nine in Marshfield, Missouri. He spent his early years in the
12 state of Kentucky. Then he moved with his family to Chicago, Illinois. He attended the University of
13 Chicago. He studied mathematics and astronomy.
14 """
15 # -----
16 sentence_List = str.split(".")
17 temp_List = []
18 for sentence in sentence_List:
19     if operator.contains(sentence, "Edwin Hubble"):
20         print(sentence)
21     if operator.contains(sentence, "edwin hubble"):
22         print(sentence)
23         new_Sentence = sentence.replace("edwin hubble", "Edwin Hubble")
24         temp_List.append(new_Sentence)
25     elif operator.contains(sentence, "Edwin hubble"):
26         print(sentence)
27         new_Sentence1 = sentence.replace("Edwin hubble", "Edwin Hubble")
28         temp_List.append(new_Sentence1)
29     elif operator.contains(sentence, "edwin Hubble"):
30         print(sentence)
31         new_Sentence2 = sentence.replace("edwin Hubble", "Edwin Hubble")
32         temp_List.append(new_Sentence2)
33     else:
34         temp_List.append(sentence)
35 new_Str = ".".join(temp_List)
36 print(150*"")

```



پاسخ تمرین شماره ۵:

```
1 import operator
2 import itertools
3 # -----
4 """Math = M, Computer = C, Industrial Engineering = IE, Electrical Engineering = EE """
5
6 expert_List1 = [("Ali", "Ahmadi", "M", 35), ("Sima", "Sadri", "C", 39), ("Ahmad", "Moradi", "M", 30), ("Ftemeh",
7 "Majd", "C", 29), ("Sara", "Biglar", "IE", 27), ("Reza", "Rahnama", "EE", 45)]
8
9 expert_List2 = [("Mina", "Gohari", "EE", 40), ("Iman", "Shams", "M", 26), ("Farzad", "Yeganeh", "M", 41), ("Ali",
10 "Imani", "C", 33), ("Aref", "Alameh", "M", 32), ("Narges", "Sohrabi", "C", 35)]
11
12 # =====
13 def sorted_Expert_List():
14     expert_List_Final = list(itertools.chain.from_iterable([expert_List1, expert_List2]))
15     sorted_expert_List_Final = sorted(expert_List_Final, key = operator.itemgetter(2))
16     return sorted_expert_List_Final
17
18 # =====
19 def create_Group():
20     math_List = list(itertools.compress(sorted_Expert_List(), [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]))
21     math_Group = list(itertools.combinations(math_List, 3))
22     return math_Group
23
24 # =====
25 print("Name \tFamily\tMajor\tAge")
26 for expert in sorted_Expert_List():
27     for information in expert:
28         print(information, end="\t")
29     print()
30 for group in create_Group():
31     print(group)
```



پاسخ تمرین شماره ۶:

```
39 import collections
40
41 Mobile = collections.namedtuple("Mobile", "Brand Model Price Colors")
42
43 m1 = Mobile("Samsung", "xxx", 6000000, ["red", "black"])
44 m2 = Mobile("Apple", "xxx", 20000000, ["blue", "black", "red"])
45 m3 = Mobile("Nokia", "xxx", 8000000, ["red", "black", "white", "pink"])
46 m4 = Mobile("Samsung", "xxx", 12000000, ["red", "white"])
47
48 mobile_List = []
49 for mobile in (m1, m2, m3, m4):
50     mobile_List.append(mobile._asdict())
51 for mobile in mobile_List:
52     print(mobile)
```



### پاسخ تمرین شماره ۷:

```
16 import collections
17 # -----
18 student_List1 = [{"Name": "Ali Rezaee", "Age": 25}, {"Name": "Reza Ahmadi", "Age": 28}, {"Name": "Sara Akbari",
19 "Age": 25}, {"Name": "Bahar Najafi", "Age": 23}, {"Name": "Iman Mohamadi", "Age": 25}, {"Name": "Sima Shaker",
20 "Age": 25}, {"Name": "Negin Ghazi", "Age": 29}, {"Name": "Maryam Yaghoubi", "Age": 25}, {"Name": "Mitra Sharif",
21 "Age": 23}, {"Name": "Ahmad Moradi", "Age": 25}]
22 # -----
23 age_List1 = []
24 for student in student_List1:
25     age_List1.append(student["Age"])
26 # =====
27 age_List2 = []
28 for student in student_List2:
29     age_List2.append(student["Age"])
30 # =====
31 i = 1
32 for list in (age_List1, age_List2):
33     common_Age = collections.Counter(list).most_common(1)
34     print(f"Group{i}: {common_Age[0]}")
35     i += 1
```



### پاسخ تمرین شماره ۸:

```
71 import collections
72 # -----
73 class MyString(collections.UserString):
74     def reverse(self):
75         tempStr = ""
76         for i in range(len(self.data)-1, -1, -1):
77             tempStr += self.data[i]
78         return tempStr
79
80 str1 = MyString("hello")
81 print(str1.reverse())
82 # =====
83 class MyString(collections.UserString):
84     def insert(self, word1, word2):
85         index = self.data.find(word1)
86         return self.data[:index] + word2 + self.data[index:]
87
88 str1 = "Python is an open source language."
89 mystring = MyString(str1)
90 print(mystring.insert("language", "programming"))
```

