Python Programming Exercises

Q1) Create a program to check if a number is odd or even.
A.OddOrEven.py
Q2) Create a program that asks the user to enter their name and their age. Print the year the person's age is 100.
A.AgeCalculator.py
Q3) Create a program to generate squares of a given list of numbers.
A.Squares.py
Q4) Create a program to extract numbers less than 5 from the input list.
A.LessThan5.py
Q5) Create a program to generate a list of common elements in two lists.
A.CommonElement.py
Q6) Create a Simple calculator for addition and subtraction.
A.SimpleCalculator.py
Q7) A Guess to match random integers generated by a Python program. User will type a number, if it matches with a number generated by the program, it's a success.
A.GuessingGame.py
Q8) Create a program to generate Pyramid of stars like
*
* *
* * *
* * * *
* * * * *
A.Pyramid.py
Q9) Given a list of elements or lists, create a program to iterate through this and print elements of the list (the output should be elements only not "lists").
A.FunRecursion.py
Q10) Check if a string is a palindrome or not.
(A palindrome is a string that reads the same forwards and backwards.)
A.palindrome.py

Q11) Create an empty list called “emp_names”, add these items to it.
aslam, veda, peter, Daniel, William, maria
a). Add a name “Sanmoy” in between Daniel and William
b). Remove the name “Daniel” from list, again add it at same position
c). Print all the names which ever have the alphabet “e” in them.

Q12) Convert above list to dictionary with name - “emp_dict” such that keys are - aslam, veda, peter, Daniel, William, maria, and respective values are - Saudi, India, Britan, Israel, USA, Russia
a). Add these new key value pairs to above dictionary - (Oliver, Australia), (Fernanda, Brazil), (Aika, Japan), (Daichi, Korea), (Zhang wei, China)
b).For above dictionary, extend the value list by adding these ages, that means new list contains two items for values, they are - country_name, list of age for every person.
Here is the info of person & respective age
aslam-25,veda-23,peter-19,Daniel-38,william-50,maria-32,Oliver-21,Fernanda-26,Aika-42,Daichi-18,Zhang We-35

c). Collect all the ages from above dictionary to ages tuple, and person names to person_names tuple

Q13) Take above person_names Tuple and create a new Tuple with name "prefix_names" which should contain all the names but with a prefix "Mr. or Ms. " as a prefix for each name.
Output: The tuple should look like (Mr or Ms. Aslam, Mr or Ms. Veda, Mr or Ms. Peter etc)

Must: Version1 code: For this exercise you should define your own function called "addPrefix(person_name)".
Version 2 code :For the same exercise you should achieve the same functionality using lambda function.

Q14) Collect all the person_names from above exercises to your favourite collection, loop through them and print the names if they don't contain character "e" in them.
Use: break/continue

NOTE: Must use for/while loops & if else blocks where necessary.

Let me know if you have any questions.

You can mail your questions to rahul@datajango.com, yukesh@datajango.com, vamsi@datajango.com