Harbor Freight Tools Data Scientist Technical Assessment

Use the following table for the next 2 questions:

Store Number	Customer	Order Total
1	A	$45
1	B	$50
2	C	$43
2	D	$48
…	…	…

1. Write a SQL query to create a table that shows, for each store, the value of the highest order total for the store.

NOTE: In the given code, the word “Table” refers to the table provided above.

```
CREATE TABLE MaxValues AS

SELECT `Store Number`, MAX(`Order Total`)

FROM Table

GROUP BY  `Store Number`;
```

2. Write a SQL query to find the customer who spent the most for each store.

```
SELECT  `Store Number`, Customer
FROM (SELECT *,
RANK()  OVER (PARTITION BY  `Store Number`, ORDER BY `Order Total` DESC) AS
rank_in_store
FROM Table)
WHERE rank_in_store=1;
```

P.S: In the above codes I assumed that “Order Total” means the total (sum) of orders of that specific customer from that specific store. If I had access to a data dictionary saying that this column is only one purchase, I would sum the purchases up (using SELECT SUM(Order Total) GROUP BY Store Number, Customer) first

3. In whatever language you would like: write a program that prints the numbers from 1 to 100. But for multiples of three print “Three” instead of the number and for multiples of five print “Five”. For numbers which are multiple of both three and five print “ThreeFive”.

```
#Python

#Creating a list to store all the sequence

output = []

#creating a for loop to check all numbers from 1 to 100

for num in range(1, 101):

    #Creating a string to record the divisibility for each number

    ans_str = ""

    # Divisible by 3

    if num%3==0:

        ans_str += "Three"

    # Divisible by 5

    if num%5==0:

        ans_str += "Five"

    # Not divisible by 3 or 5 (str is empty)

    if not ans_str:

        ans_str = str(num)



    # Append the str for this number to the final output list

    output.append(ans_str)

print(output)
```

4. Now, write code so that the number/word pairs are part of the input, and I could pass an arbitrary amount of them to get a result. As an example, show how you can use your code to print “Seventeen” for all multiples of 17.

This code is written for an arbitrary range of numbers (Like 1 to 100, or 50 to 500) and arbitrary numbers that you want to be printed as words (Like Three Five Nine)

NOTE: In case the module num2words was not installed by command, you can install it from “Python Packages”

```
#Python
# importing the necessary Library
from num2words import num2words as n2w

#Writing the function for deliberate number of ranges or numbers
def game():
    # Getting the desired range of numbers
    try:
        a, b = [int(x) for x in
                input("Please enter the start and stop of your desired range separated with a space:
").split()]

        if a > b or a < 0:
            return ('Invalid format for the range')

    except ValueError:
        return ('Invalid format for the range')

    # Getting the numbers to be printed as word (like 3 and 5 in the previous problem)
    try:
        x = [int(x) for x in
             input("Now enter the number(s) you want to see as word(s) separated with a space:
").split()]
    except:
        return ('Invalid format for the printing numbers')

    # creating the output as a list
    output = []

    for num in range(a, b + 1):
        ans_str = ""
        #If the number is divisible by any number the concatenation will happen
        for j in range(0, len(x)):
            if num % x[j] == 0:
```

```
            ans_str += n2w(x[j])
        if not ans_str:
            ans_str = str(num)
        output.append(ans_str)
    print(output)


game()
```