Technical Case Study

*Coding Skills – Round 1*

# **Instructions**

Please read through the questions below. Please answer the following questions either using Python or SQL, whichever you prefer. Please keep in mind code efficiency when answering. The objective is to assess your coding skills and style. The interviewer(s) are likely to ask additional questions beyond those listed below to assess your understanding.

Please submit your answer 1 hour before the start of interview to give the interviewers time to evaluate your answers. Attempt as many questions as you can.

**IMPORTANT NOTE:** The final answers/results should be sent via one of these channels:

1. upload your test on GitLab and send the link / access
2. OR, convert the .py file to a .txt file and send file to dunnhumby point of contact

Do not send .py or. ipynb. Our systems block external .py and. ipynb files from being downloaded.

# **Section 1: general Python**

1. Given the list of basket values, do the following:
   1. Print out whether each basket is small (basket value < £5), medium (£5 ≤ basket value < £10) or large (basket value ≥ £10);
   2. Sum and print the value of the medium value baskets.

**basket\_values = [3.43,9.73,7.56,9.52,15.23,2.25,6.44,7.38]**

1. You are given the following nested dictionaries, which represent items in a basket.  Do the following:
   1. Return the product name for item 7527.
   2. Return the total value of this basket.
   3. Add another entry for a product that costs £4.95, has ID 7524 and name ‘poppy seeds’.

**basket = {'2624': {'price': 0.5, 'prod\_name': 'salt'},**

**'2894': {'price': 3.25, 'prod\_name': 'yeast'},**

**'7527': {'price': 2.5, 'prod\_name': 'flour'}}**

1. Below is the source code for a function called ‘get\_sql\_string’.

**1   def get\_sql\_string(stores):**

**2      store\_names = [x.split(', ')[0] for x in stores]**

**3      store\_names = [x.replace(' ', '\_') for x in store\_names]**

**4      store\_regions = [x.split(',')[1] for x in stores]**

**5      locations = store\_names + store\_regions**

**6      columns = ['sales\_' + x.lower() for x in locations]**

**7      return ', '.join(columns)**

* 1. There is a bug in line 4. What should the line be?
  2. Assuming this bug was fixed, what would be returned if the following command was executed:

**my\_stores = ['Fulham Palace Rd, Hammersmith', 'Crown St, Reading', 'Leavesden Green, Watford']**

**get\_sql\_string(my\_stores)**

1. Write a function that:
   1. accepts a list of strings as input.
   2. returns an alphabetically ordered list of unique strings.
   3. prints the string(s) with maximum length in the console.

# **Section 2: pandas/SQL data frames**

Suppose you have access to two data frames, *dept* and *employee*.  The first three rows of these are represented below.

 employee.head(3)



dept.head(3)



*Notes:*

* *MGR is the EMPNO of the Employee whom the observed Employee reports to.*
* *DEPTNO is a foreign key.*

Questions

1. Find the *n*th largest salary.
2. List the highest salary paid for each job.
3. In which year did most people join the company?  Display the year and the number of Employees.
4. Create a new column with the length of service of the Employees (in the form *n* years and *m* months).
5. List all the Employees who have at least one person reporting to them.