|  |  |  |  |
| --- | --- | --- | --- |
| **Job Title:** | Python Developer | **Contact:** | cprior@bink.com |
| This is a benchmark test to ensure that Python developers can show a good understanding of the fundamentals of reading, coding and delivering to a timeframe. | | | |
| **Rules** | | | |
| A link to a public Git repository with your final solution must be provided within 5 working days of receipt of the test.  Guidelines   1. To help understand how you approach the problem, we will assess your use of source control and how you build to the final solution, checking what is committed along each step 2. The code must be written in Python 3. 3. You may use any frameworks or libraries to complete this task, excluding data analysis libraries like Pandas. 4. Unit tests must be provided | | | |
| **Objective** | | | |
| A data file will be provided alongside this test. The dataset is a CSV which contains publicly available data about mobile phone masts in an area of the UK. This file contains un-normalised data (such as multiple variations of Tenant Name) – **treat these as individual tenants.**  Actions   1. Load the data file, process and output the data in the forms specified 2. Read in, process and present the data as specified in the requirements section 3. Demonstrate usage of list comprehension for at least one of the tasks 4. Allow user input to run all of your script, or specific sections | | | |
| **Requirements** | | | |
| 1. Read in the attached file    1. Produce a list sorted by “Current Rent” in ascending order    2. Obtain the first 5 items from the resultant list and output to the console 2. From the list of all mast data, create a new list of mast data with “Lease Years” = 25 years.    1. Output the list to the console, including all data fields    2. Output the total rent for all items in this list to the console 3. Create a dictionary containing tenant name and a count of masts for each tenant    1. Output the dictionary to the console in a readable form 4. List the data for rentals with “Lease Start Date” between 1st June 1999 and 31st August 2007    1. Output the data to the console with dates formatted as DD/MM/YYYY | | | |
| **Assessment** | | | |
| Your code will be reviewed and assessed according to the following:   1. Adherence to the requirements 2. Code quality – readability & structure of the code 3. Unit test coverage and relevance of the tests | | | |
| **Helpful Tips** | | | |
| If you struggle completing the test or have concerns over certain aspects that is okay – just highlight it to us when you submit your test. Explain what you couldn’t get working and steps you took to solve the problem. Whilst we want to see completed tests it is just as important for us to see how you approached an issue and attempted to find a solution.  Do not overthink your solution. Keep it simple and use what you know.  Don’t forget the README | | | |