CDL JavaScript Kata

1. We expect you to spend **around 1.5 hours** on this challenge - use your time wisely, we know it's valuable!
2. Ideally use JavaScript as we can run that kind of code fairly easily.
3. Tell us about key decisions you made and what you'd do if you had more time.
4. Keep your solution simple, making effective use of your chosen language.
5. (Intermediate /Senior role only) Do provide some tests, especially around the most important logic.
6. We're not looking for production-ready enterprise-scale code (no databases or servers please, and keep boilerplate to a minimum).

Implement the code for a checkout system that handles pricing schemes such as “apples cost 50 pence, three apples cost £1.30.”

Implement the code for a basket that calculates the total price of a number of items. In a normal store, things are identified using Stock Keeping Units, or SKUs. In our store, we’ll use individual letters of the alphabet (A, B, C, and so on). Our goods are priced individually. In addition, some items are multi-priced: buy ‘n’ of them, and they’ll cost you ‘y’ pence. For example, item ‘A’ might cost 50 pence individually, but this week we have a special offer: buy three ‘A’s and they’ll cost you £1.30.

Use these prices for the sample:

|  |  |  |
| --- | --- | --- |
| Item | Unit Price (pence) | Special Price (pence) |
| A | 50 | 3 for 130 |
| B | 30 | 2 for 45 |
| C | 20 |  |
| D | 15 |  |

Our checkout accepts items in any order, so that if we scan a B, an A, and another B, we’ll recognize the two B’s and price them at 45 (for a total price so far of 95). Because the pricing changes frequently, we need to be able to pass in a set of pricing rules each time we start handling a checkout transaction.

The solution should allow for items to input within a simple user interface, and allow for a final total to be calculated and for a running total after each item is added to the basket

Please use JavaScript for the solution and place this onto a publicly accessible Github repository and let us know where it is.

Finally, please commit changes frequently. It is more important to see the progress than the finished product.