UnisLink Technical Assessment

# Assignment: Discounts on apparel

A good friend of yours has recently started an apparel boutique and she relies on you to implement a technology solution to calculate the discount on the products her customers buy from her. She has a rich collection of apparels and accessories. Will you help your friend?

## The problem in detail

The objective is to calculate the total discount on the items that a customer buys from the retailer in one shopping session. The retailer has several categories of products. In fact, categories have subcategories which themselves can have subcategories. Below is a diagram. Casuals is a subcategory of Trousers, which by itself is a subcategory of Men's wear. Some categories have discounts.

Men's wear
- |- Shirts
- |- Trousers
  - |- Casuals (30% off)
  - |- Jeans (20% off)

Women's wear (50% off)
- |- Dresses
- |- Footwear

Each product belongs to a brand which by themselves are running discounts. Below is the list:

**Brands Discounts:**

| | |
|---|---|
| Wrangler | 10% |
| Arrow | 20% |
| Vero Moda | 60% |
| UCB | None |
| Adidas | 5% |
| Provogue | 20% |

This way, a product can have three types of discounts applicable:

1. Discount on the brand
2. Discount on the category
3. Discount on the ancestor category (e.g. Footwear doesn't have a discount, but it's parent category Women's wear has 50% off). It is worth noting, that it is an ancestor: not just a direct parent, anyone in the lineage. The discount that is applied is the greatest of the above three.

# UnisLink Technical Assessment

## Input data for data loading:

The following input data will be provided as pipe-delimited files and need to be imported. The data may be modified during the interview to allow for improvisations and better discussion of concepts.

**Inventory file**

SKUID|Brand|Category|ListPrice
1|Arrow|Shirts|800
2|Vero Moda|Dresses|1400
3|Provogue|Footwear|1800
4|Wrangler|Jeans|2200
5|UCB|Shirts|1500

**Discounts file**

DiscountType|DiscountOn|Discount%
C|Casuals|30
C|Jeans|20
C|Women's wear|50
B|Wrangler|10
B|Arrow|20
B|Vero Moda|60
B|UCB|0
B|Adidas|5
B|Provogue|20

DiscountType => C = category; B = brand

A product can have three types of discounts applicable:
1. Discount on the brand
2. Discount on the category
3. Discount on the ancestor category (e.g. Footwear doesn't have a discount, but it's parent category Women's wear has 50% off). It is worth noting, that it is an ancestor: not just a direct parent, anyone in the lineage.

The discount that is applied is the greatest of the above three.

# UnisLink Technical Assessment

## A sample scenario worked out:

Let us say a customer adds 2 items to his/her cart:
a. 1 Provogue Footwear
b. 1 Wrangler Jeans

the sample XML input of customer shopping cart for procedure will be

```
<cart>
   <items>
      <item skuid=3 unitsbought=1 />
      <item skuid=4 unitsbought=1 />
    </items>
</cart>
```

The discounted price is the expected output for the procedure. In this case it will be 2660

For SKU ID 3 (i.e. Provogue Footwear). The list price is 1800. The applicable discounts are:
1. Discount on brand: 20%
2. Discount on category (Footwear): 0%
3. Discount on parents (Women's wear): 50%
So, the discount that is applied 50% and the final price for SKUID = 3 is 50% of 1800 = 900

For the Jeans of Wrangler Brand, the list price is 2200 and the discounts are:
1. Discount on brand: 10%
2. Discount on category (Jeans): 20%
3. Discount on parents (Trousers, Men's wear): None
So, the discount that is applied 20%. The final price for the product is 80% of 2200 = 1760

So the Expected output = 1760 + 900 = 2660

## Deliverables expected:

1. Data model for the entities involved in the problem - **DDL scripts of tables**
2. A utility to load data Inventory and Discount files; **Write a Python program that will accept two parameters - Source File name and Target Table name. The columns should be configurable. i.e. if in future a new column needs to be added, there should be no change to the Py program.**
3. A SQL program that takes XML input of shopping cart items and displays the price before discount, price after discount and total savings for the buyer – **SQL server Stored Procedure**