In [49]:
```
import pandas as pd
pd.options.display.float_format = '{:20,.2f}'.format
pd.set_option('display.max_rows', 5000)
pd.set_option('display.max_columns', 5000)
pd.set_option('display.width', 1000)
pd.set_option('display.max_colwidth', -1)
```

In [50]:
```
abt_buy_path = (r'/home/ubuntu/jupyter/ServerX/1_Standard Data Integrat
                r'/Unprocessed Data/product_samples/abt_buy/')
```

Read csv data

In [51]:
```
abt_df = pd.read_csv(abt_buy_path + 'Abt.csv', sep=',', quotechar='"', 
```

In [122]:
```
abt_df.columns
```

Out[122]:
```
Index(['id', 'name', 'description', 'price', 'source', 'producer'], dt
ype='object')
```

In [52]:
```
buy_df = pd.read_csv(abt_buy_path + 'Buy.csv', sep=',', quotechar='"', 
```

In [123]:
```
buy_df.columns
```

Out[123]:
```
Index(['id', 'name', 'description', 'producer', 'price', 'source'], dt
ype='object')
```

In [53]:
```
abt_df.columns
```

Out[53]:
```
Index(['id', 'name', 'description', 'price'], dtype='object')
```

In [54]:
```
buy_df.columns
```

Out[54]:
```
Index(['id', 'name', 'description', 'manufacturer', 'price'], dtype='o
bject')
```

In [55]:
```
# convert all price values to comparable float prices
abt_df['price'] = abt_df['price'].apply(lambda x: float(str(x).replace(
buy_df['price'] = buy_df['price'].apply(lambda x: float(str(x).replace(
```

In [56]:
```
abt_df['source'] = 'abt'
abt_df['producer'] = None
buy_df['source'] = 'buy'
```

In [57]:
```
buy_df.rename(columns={'manufacturer':'producer'}, inplace=True)
```

In [58]:
```
abt_buy_df = pd.concat([abt_df, buy_df])
```

```
/home/ubuntu/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.
py:1: FutureWarning: Sorting because non-concatenation axis is not ali
gned. A future version
of pandas will change to not sort by default.

To accept the future behavior, pass 'sort=False'.

To retain the current behavior and silence the warning, pass 'sort=Tru
e'.

  """Entry point for launching an IPython kernel.
```

In [59]:
```
abt_buy_df.rename(columns={'id':'Id'}, inplace=True)
```

In [60]:
```
abt_buy_df[abt_buy_df['Id'] == 6284]
```

Out[60]:

	description	Id	name	price	producer	source
4	Bose 161 Bookshelf Speakers In White - 161WH/ Articulated Array Speaker Design/ High-Excursion Twiddler Drivers/ Magnetically Shielded/ Priced Per Pair/ White Finish	6284	Bose 27028 161 Bookshelf Pair Speakers In White - 161WH	158.00	None	abt

In [61]:
```
abt_buy_df.columns
```

Out[61]:
```
Index(['description', 'Id', 'name', 'price', 'producer', 'source'], dt
ype='object')
```

In [62]:
```
abt_buy_df = abt_buy_df[['Id', 'description',  'name', 'price', 'produc
```

In [63]:
```
processed_data_path = (r'/home/ubuntu/jupyter/ServerX/1_Standard Data I
                       r'/Processed Data/product_samples/')
```

In [64]:
```
abt_buy_df.to_csv(processed_data_path + 'abt_buy_all.csv', sep=',', quo
```

Prepration of Mapping File

In [65]:
```
match_file_name = 'abt_buy_perfectMapping.csv'
```

In [66]:
```
abt_buy_match_df = pd.read_csv(abt_buy_path + match_file_name, sep=',',
```

In [67]:
```
abt_buy_match_df.columns
```

Out[67]:
```
Index(['idAbt', 'idBuy'], dtype='object')
```

In [68]:
```
abt_buy_match_df.rename(columns={'idAbt':'abt_id', 'idBuy':'buy_id'}, i
```

In [69]:
```
abt_buy_match_df.to_csv(processed_data_path + 'abt_buy_match.csv', sep=
```

In [115]:
```
def get_abt_name(abt_id):

    relevant_entry = list(abt_df[abt_df['id'] == abt_id]['name'])
    if not len(relevant_entry) == 1:
        raise Exception('id duplicate')
    else:
        return relevant_entry[0]

def get_abt_desc(abt_id):

    relevant_entry = list(abt_df[abt_df['id'] == abt_id]['description'])
    if not len(relevant_entry) == 1:
        raise Exception('id duplicate')
    else:
        return relevant_entry[0]

def get_buy_name(buy_id):

    relevant_entry = list(buy_df[buy_df['id'] == buy_id]['name'])
    if not len(relevant_entry) == 1:
        raise Exception('id duplicate')
    else:
        return relevant_entry[0]

def get_buy_desc(buy_id):

    relevant_entry = list(buy_df[buy_df['id'] == buy_id]['description'])
    if not len(relevant_entry) == 1:
        raise Exception('id duplicate')
    else:
        return relevant_entry[0]
```

In [83]:
```
get_abt_name(552)
```

Out[83]:
```
'Sony Turntable - PSLX350H'
```

In [116]:
```
abt_buy_match_df['abt_name'] = abt_buy_match_df['abt_id'].apply(lambda x
abt_buy_match_df['abt_desc'] = abt_buy_match_df['abt_id'].apply(lambda x
abt_buy_match_df['buy_name'] = abt_buy_match_df['buy_id'].apply(lambda x
abt_buy_match_df['buy_desc'] = abt_buy_match_df['buy_id'].apply(lambda x
```

In [ ]:
```
abt_buy_match_df['abt']
```

In [121]:
```
len(abt_buy_match_df)
```

Out[121]:
```
1097
```

In [120]:
```
abt_buy_match_df[['abt_name', 'buy_name', 'abt_desc', 'buy_desc']]
```

Out[120]:

	abt_name	buy_name	
0	Linksys EtherFast 8-Port 10/100 Switch - EZXS88W	Linksys EtherFast EZXS88W Ethernet Switch - EZXS88W	Linksys EtherFast 8-Port 10/100 Switch - EZXS88\ Perfect For Optimizing 10BaseT And 100E Network/ Speeds Of Up To 200Mbps In Full Duplex Const
1	Linksys EtherFast10/100 5-Port Auto-Sensing Switch - EZXS55W	Linksys EtherFast EZXS55W Ethernet Switch	Linksys EtherFast10/100 5-Port Auto-Sensing S\ Autosensing Ports With Both Half And Full Duple: Your 10BaseT And 100BaseTX Network Hardware 10Mbps, 20I
2	Netgear ProSafe 5 Port 10/100 Desktop Switch - FS105	Netgear ProSafe FS105 Ethernet Switch - FS105NA	Netgear ProSafe 5 Port 10/100 Desktop Switch 10/100
3	Belkin F3H982-10 Pro Series High Integrity 10 Feet Monitor Cable - F3H98210	Belkin Pro Series High Integrity VGA/SVGA Monitor Extension Cable - F3H982-10	Belkin F3H982-10 Pro Series High Integrity 10 I Recommended For Monitors 17' And Larger/ Desig Imaging And High Speed/ Double Shieldin

In [70]:
```
len(abt_buy_match_df)
```

Out[70]:
```
1097
```

In [25]:
```
abt_buy_match_df.head(10)
```

Out[25]:

	abt_id	buy_id
0	38477	10011646
1	38475	10140760
2	33053	10221960
3	27248	10246269
4	25262	10315184
5	36260	10316920
6	35810	10326220
7	32034	10333368
8	38473	10333846
9	23686	10333848

In [42]:
```
abt_df[abt_df['id'] == 35810]
```

Out[42]:

	id	name	description	price	source	producer
778	35810	Canon KP-36IP Color Ink & Paper Set - 7737A001	Canon KP-36IP Color Ink & Paper Set - 7737A001/ 36 Sheets Of 4' x 6' Photo Paper/ Ink Cartridge For Compatible Canon Dye Sublimation Printer	12.00	abt	None

In [41]:
```
buy_df[buy_df['id'] == 10326220]
```

Out[41]:

	id	name	description	producer	price	source
6	10326220	Canon KP 36IP Print Cartridge / Paper Kit - 7737A001	36 Page 4' x 6'	Canon	9.99	buy

In [ ]:
```

```