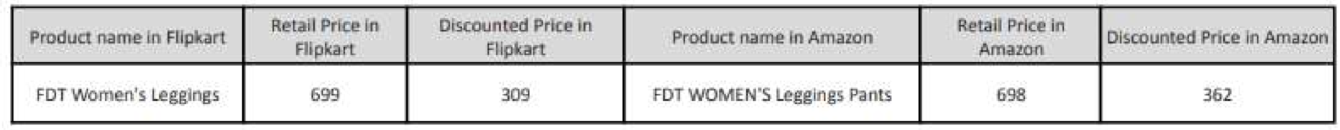
Assignment Report and Approach

Assignment: Match similar products from the Flipkart dataset with the Amazon dataset.

Description: Retrieving the Product prices from flipkart and amazon based on the user input.



Product name in Flipkart	Retail Price in Flipkart	Discounted Price in Flipkart	Product name in Amazon	Retail Price in Amazon	Discounted Price in Amazon
FDT Women's Leggings	699	309	FDT WOMEN'S Leggings Pants	698	362

https://colab.research.google.com/drive/1_Olh5GjvklWxasFGRJKQ6lMDkezOpfl0?usp=sharing

Approach:

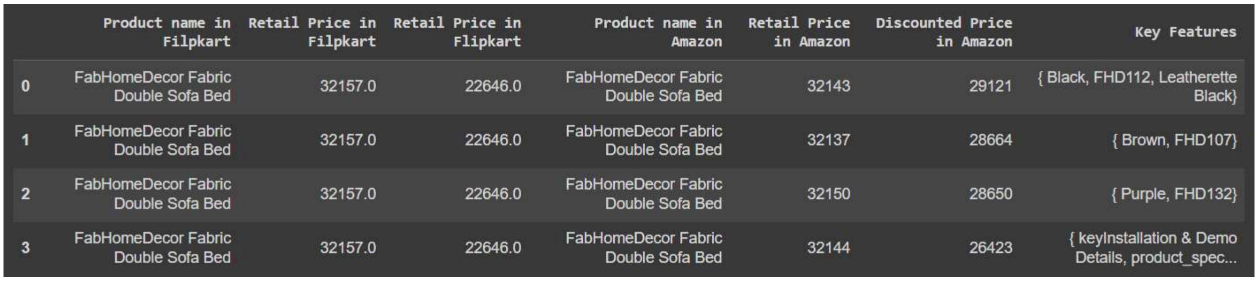
The Dataset provided has 15 features with it.
In which every product has an uniq_id and PID (product information document) which will be unique to each product , sometimes the name of the product can be changed but the uniq_id and PID remain the same .
Our **approach is based on the PID number of the product.**
Once the user provides the input , we pull the pid number of the product from flipkart dataset then , look for the exact pid number in the Amazon dataset. If found, the table consisting of price comparison will be displayed.

- **Multiple results:**

  Once the user provides the product name , there will be a case in which multiple results can be encountered , for example if the user was searching for a sort of t-shirt . Different prices will be allocated to the same company product in which the product consists of a pack of 2 t-shirts , pack of 3 and so on . Need to look into these cases, **Key features differences** between the multiple results have to be displayed in the table to two to avoid the user getting confused.

  In order to tackle this , a feature named **product specification** has to be processed to find the difference between the product . This feature had a huge amount of noise in it. Extraction of features has to be carried out removing the noise. All the null values in the product specification were changed to the status of "no info" in order to avoid a rise of errors later on.

  Sample output:
  The product FabHomeDecor Fabric Double Sofa Bed consists of multiple results with the same name , but once we look deep dive into the product specification the color of the sofa varies in between them. Color playing as a key feature had an impact on the prices of sofa too.
  Similar name on both datasets for all the results but key features bring out the differences.



	Product name in Filpkart	Retail Price in Filpkart	Retail Price in Flipkart	Product name in Amazon	Retail Price in Amazon	Discounted Price in Amazon	Key Features
0	FabHomeDecor Fabric Double Sofa Bed	32157.0	22646.0	FabHomeDecor Fabric Double Sofa Bed	32143	29121	{ Black, FHD112, Leatherette Black}
1	FabHomeDecor Fabric Double Sofa Bed	32157.0	22646.0	FabHomeDecor Fabric Double Sofa Bed	32137	28664	{ Brown, FHD107}
2	FabHomeDecor Fabric Double Sofa Bed	32157.0	22646.0	FabHomeDecor Fabric Double Sofa Bed	32150	28650	{ Purple, FHD132}
3	FabHomeDecor Fabric Double Sofa Bed	32157.0	22646.0	FabHomeDecor Fabric Double Sofa Bed	32144	26423	{ keyInstallation & Demo Details, product_spec...

Explanation:

The code was divided into four sections for proper understanding

- **Importing libraries and Datasets:**

  Loading of datasets to the environment and importing the pandas and numpy libraries were performed over here.Proper upload of dataset has to be verified once.

- **Observations / EDA:**

  Strength of the dataset, columns name , datatype of each features , looking for present of null values And sometimes even the prices of the product were in negative which is not possible i have replaced The negative values in the prices as “0” and null values in the product description as “no info” status.

```
amazon["product_specifications"].fillna("No info", inplace = True)
filpkart["product_specifications"].fillna("No info", inplace = True)
```

  Later some features were dropped which includes {"product_url","product_category_tree","image","is_FK_Advantage_product","description", "product_rating","overall_rating","brand". **[using user defined pre_processing func]**

  Final features were :
  uniq_id, pid, retail and discount price, product_specification and timestamp of crawl.

- **Approach:**

  This was to play the key part right from the input to the fetching the details of the product.

  >> INPUT will received from the user, later based on the product name corresponding PID of the product will be fetched from flipkart in case of multiple products available all the PID will be noted.

  >> Generated two list in the name of amazon_pid_list and filpkart_pid_list which consist of all the product pid.

  >> **Searching_filpkart**

The func searching_filpkart will look generate a dictionary and return to the variable called filpkart_results,
The dictionary consist of
{index: [name, id, PID, retail_price,
discount, product_sepecifications]

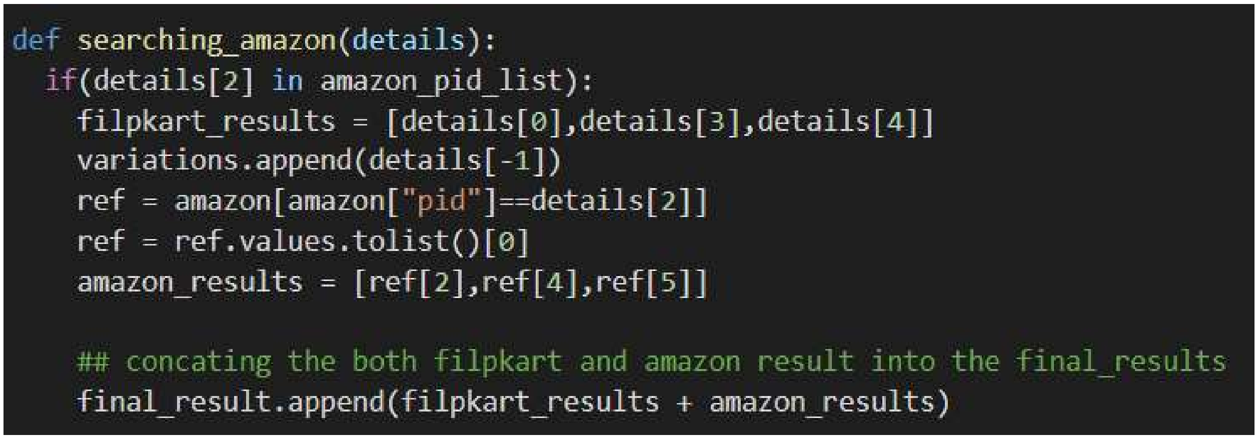
With index as key and features in a list.

```
def searching_filpkart(product):
  index = 0
  result_index={}
  for i in filpkart.product_name==product:
    if(i==True):
      result_index[index] = [filpkart.product_name[index],
    index+=1

  return result_index
```

Output:
Filpkart_results a dictionary data type with features listed in values.

>> Searching_amazon and concating the both the results



```
def searching_amazon(details):
  if(details[2] in amazon_pid_list):
    filpkart_results = [details[0],details[3],details[4]]
    variations.append(details[-1])
    ref = amazon[amazon["pid"]==details[2]]
    ref = ref.values.tolist()[0]
    amazon_results = [ref[2],ref[4],ref[5]]

    ## concating the both filpkart and amazon result into the final_results
    final_result.append(filpkart_results + amazon_results)
```

Based on the filpkart_results now we are passing the PID of the results into the func searching_amazon, where PID will be searching over the amazon_pid_list generated before.
Once found, we will consider the Name, retail price and discounted price from both the results and concatenate them to the final_result.

In case of multiple products , we are simultaneously generating the Variations list in which product_specifications will be appended for each iterations.
The mentioned three blocks in the code were completely used to reduce as much as noise in the product_specification and bring out the key features from it.

Example:

Right from here which was the initial input_specifications
{"product_specification"=>[{"key"=>"Number of Contents in Sales Package", "value"=>"Pack of 3"}, {"key"=>"Fabric", "value"=>"Cotton Lycra"}, {"key"=>"Type", "value"=>"Cycling Shorts"}, {"key"=>"Pattern", "value"=>"Solid"}, {"key"=>"Ideal For", "value"=>"Women's"}, {"value"=>"Gentle Machine Wash in Lukewarm Water, Do Not Bleach"}, {"key"=>"Style Code", "value"=>"ALTHT_3P_21"}, {"value"=>"3 shorts"}]}

Once noise was reduced our final output will be
{ Pack of 3, 3 shorts, ALTHT_3P_21}

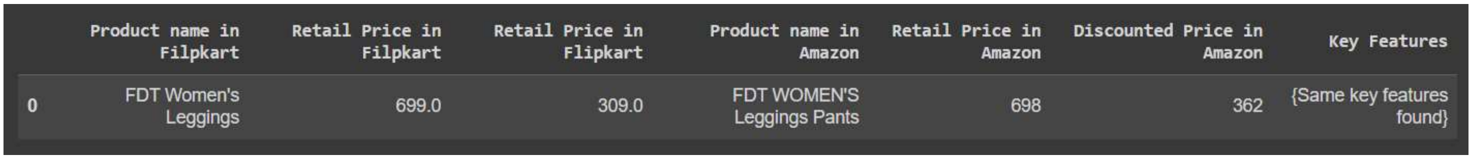
Which sort of indicating the key features were the product consisted of 3 shorts and model number over.

>>**Final Result**
In the section final result we will be converting all the outputs we have generated which include both the price comparison and the varying parameters into the data frame format and display the results.

My humble request is for one on one call or an online meeting so I could explain my approach and code, any changes and requirements can be discussed as well. Open for receiving suggestions sir.

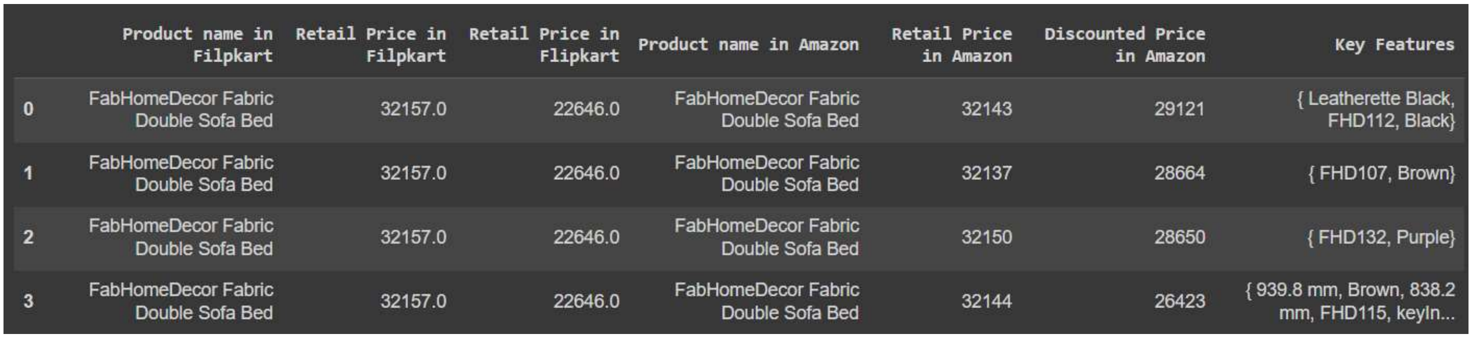
SAMPLE OUTPUTS:

Product: FDT Women's Leggings from Flipkart



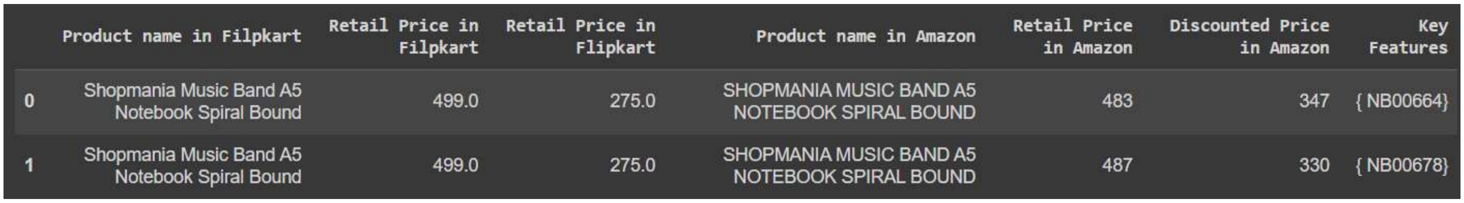
	Product name in Filpkart	Retail Price in Filpkart	Retail Price in Flipkart	Product name in Amazon	Retail Price in Amazon	Discounted Price in Amazon	Key Features
0	FDT Women's Leggings	699.0	309.0	FDT WOMEN'S Leggings Pants	698	362	{Same key features found}

Product: FabHomeDecor Fabric Double Sofa Bed



	Product name in Filpkart	Retail Price in Filpkart	Retail Price in Flipkart	Product name in Amazon	Retail Price in Amazon	Discounted Price in Amazon	Key Features
0	FabHomeDecor Fabric Double Sofa Bed	32157.0	22646.0	FabHomeDecor Fabric Double Sofa Bed	32143	29121	{ Leatherette Black, FHD112, Black}
1	FabHomeDecor Fabric Double Sofa Bed	32157.0	22646.0	FabHomeDecor Fabric Double Sofa Bed	32137	28664	{ FHD107, Brown}
2	FabHomeDecor Fabric Double Sofa Bed	32157.0	22646.0	FabHomeDecor Fabric Double Sofa Bed	32150	28650	{ FHD132, Purple}
3	FabHomeDecor Fabric Double Sofa Bed	32157.0	22646.0	FabHomeDecor Fabric Double Sofa Bed	32144	26423	{ 939.8 mm, Brown, 838.2 mm, FHD115, keyIn...

Product: Shopmania Music Band A5 Notebook Spiral Bound



	Product name in Filpkart	Retail Price in Filpkart	Retail Price in Flipkart	Product name in Amazon	Retail Price in Amazon	Discounted Price in Amazon	Key Features
0	Shopmania Music Band A5 Notebook Spiral Bound	499.0	275.0	SHOPMANIA MUSIC BAND A5 NOTEBOOK SPIRAL BOUND	483	347	{ NB00664}
1	Shopmania Music Band A5 Notebook Spiral Bound	499.0	275.0	SHOPMANIA MUSIC BAND A5 NOTEBOOK SPIRAL BOUND	487	330	{ NB00678}

In the final results, a table with the price comparison will be shown , in case of multiple products all the key features such as quantity, color variation, model number and so on were depicted .