

TEAM ID: PNT2022TMID14136

PROJECT NAME: DemandEst - AI powered Food Demand Forecaster

Team Leader

The screenshot displays a Jupyter Notebook environment with the following content:

Dropping Columns

Let's drop columns "center_id" and "meal_id" as they are not required for the further process. Display the changes of trainfinal table using head().

```
In [110]: trainfinal = trainfinal.drop(['center_id', 'meal_id'], axis=1)
trainfinal.head()
```

```
Out[110]:
```

	id	week	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders	category	cuisine	city_code	region_code	center_type
0	1379560	1	136.83	152.29	0	0	177	Beverages	Thai	647	56	TYPE_C
1	1018704	2	135.83	152.29	0	0	323	Beverages	Thai	647	56	TYPE_C
2	1196273	3	132.92	133.92	0	0	96	Beverages	Thai	647	56	TYPE_C
3	1116527	4	135.86	134.86	0	0	163	Beverages	Thai	647	56	TYPE_C
4	1343872	5	146.50	147.50	0	0	215	Beverages	Thai	647	56	TYPE_C

Display the list of columns present in trainfinal table and store it in variable "cols"

```
In [111]: cols = trainfinal.columns.tolist()
print(cols)
```

```
['id', 'week', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders', 'category', 'cuisine', 'city_code', 'region_code', 'center_type', 'op_area']
```

Rearrange the columns by slicing the columns of "cols" and print "cols"

```
In [112]: cols = cols[:2] + cols[9:] + cols[7:9] + cols[2:7]
print(cols)
```

```
['id', 'week', 'city_code', 'region_code', 'center_type', 'op_area', 'category', 'cuisine', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders']
```

Store the changes of columns in trainfinal and display the datatypes of trainfinal using trainfinal.dtypes. Here, we can see that, we not only have numerical data but we also have object data.

```
In [113]: trainfinal = trainfinal[cols]
trainfinal.head()
```

```
Out[113]:
```

	id	week	city_code	region_code	center_type	op_area	category	cuisine	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders
0	1379560	1	647	56	TYPE_C	2.0	Beverages	Thai	136.83	152.29	0	0	0
1	1018704	2	647	56	TYPE_C	2.0	Beverages	Thai	135.83	152.29	0	0	0
2	1196273	3	647	56	TYPE_C	2.0	Beverages	Thai	132.92	133.92	0	0	0
3	1116527	4	647	56	TYPE_C	2.0	Beverages	Thai	135.86	134.86	0	0	0
4	1343872	5	647	56	TYPE_C	2.0	Beverages	Thai	146.50	147.50	0	0	0

```
In [114]: trainfinal.dtypes
```

```
Out[114]: id                int64
week                int64
```

The screenshot shows a Jupyter Notebook with the following code and output:

```

In [113]: trainfinal = trainfinal[0:500]
          trainfinal.head()

Out[113]:
```

	id	week	city_code	region_code	center_type	op_area	category	cuisine	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders
0	1379560	1	647	56	TYPE_C	2.0	Beverages	Thai	136.83	152.29	0	0	
1	1018704	2	647	56	TYPE_C	2.0	Beverages	Thai	135.83	152.29	0	0	
2	1196273	3	647	56	TYPE_C	2.0	Beverages	Thai	132.92	133.92	0	0	
3	1116527	4	647	56	TYPE_C	2.0	Beverages	Thai	135.86	134.86	0	0	
4	1343872	5	647	56	TYPE_C	2.0	Beverages	Thai	146.50	147.50	0	0	

```

In [114]: trainfinal.dtypes

Out[114]:
```

id	int64
week	int64
city_code	int64
region_code	int64
center_type	object
op_area	float64
category	object
cuisine	object
checkout_price	float64
base_price	float64
emailer_for_promotion	int64
homepage_featured	int64
num_orders	int64
dtype:	object

Team Member 1

The screenshot shows a Jupyter Notebook with the following code and output:

Dropping Columns

Let's drop columns "center_id" and "meal_id" as they are not required for the further process. Display the changes of trainfinal table using head().

```

In [110]: trainfinal = trainfinal.drop(['center_id', 'meal_id'], axis=1)
          trainfinal.head()

Out[110]:
```

	id	week	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders	category	cuisine	city_code	region_code	center_type
0	1379560	1	136.83	152.29	0	0	177	Beverages	Thai	647	56	TYPE_C
1	1018704	2	135.83	152.29	0	0	323	Beverages	Thai	647	56	TYPE_C
2	1196273	3	132.92	133.92	0	0	96	Beverages	Thai	647	56	TYPE_C
3	1116527	4	135.86	134.86	0	0	163	Beverages	Thai	647	56	TYPE_C
4	1343872	5	146.50	147.50	0	0	215	Beverages	Thai	647	56	TYPE_C

Display the list of columns present in trainfinal table and store it in variable "cols"

```

In [111]: cols = trainfinal.columns.tolist()
          print(cols)

['id', 'week', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders', 'category', 'cuisine', 'city_code', 'region_code', 'center_type', 'op_area']

Rearrange the columns by slicing the columns of "cols" and print "cols"
```

jupyter Code «rosaved)



Logout

File Edit View Insert Cell Kernel Widgets Help

Not tested

Python 3 (ipykernel)

Reanange<Reco<mnns bysicng lfe<co<mnns o{"col?anQpnni<cols

In [112]: cols = cols[:2] + cols[9:] + cols[7:9] + cols[2:7]

Store the changes of columns in trainfinal and display the datatypes of trainfinal using trainfinal.dtypes. Here, we can see that, we not only have numerical but we also have object

Out[113]:

	id	veer	ig	code	region	code	ezar	ape	op	area	ae	org	Dunne	zaectou<	pros	base	price	ema	er	Cor	promo	oz	aomep	ge	emurea	n
0	1379560	1	647	ss	TYPE_c	20	Beverages	Thai	136.93	152.29	0	0														
1	1018704	2	56	TYPE_C	20	Beverages	Thai	135.83	152.29	0	0															
2	1196273	3	ss	TYPE_c	20	eeverages	Thai	132.92	133.92	0	0															
4	1343872	5	647	56	TYPE_C	20	Beverages	Thai	146.50	147.50	0	0														

['id', 'veer', 'ig', 'code', 'region', 'code', 'ezar', 'ape', 'op', 'area', 'ae', 'org', 'Dunne', 'zaectou<', 'pros', 'base', 'price', 'ema', 'er', 'Cor', 'promo', 'oz', 'aomep', 'ge', 'emurea', 'n']

'nd4
's

M Hom<Pag-Sevior<mai<a'M Cod Ju/rMoibook \$

fl O loahou889Vnoeobook/DowToads BSS-Challenge-8925FoodDema<dFoezanmg (or-FoodDeI y-Company-using IBM-Coud manSBSP/>Ctallenge-... Q

Yolube Ma

jupyter Code «rosaved)



Logout

File Edit View Insert Cell Kernel Widgets Help

Not tested

Python 3 (ipykernel)

Out[113]:

	id	veer	ig	code	region	code	ezar	ape	op	area	ae	org	Dunne	zaectou<	pros	base	price	ema	er	for	promou<	oz	aomep	ge	emures	ni
0	1379560	1	647	ss	TYPE_c	20	Beverages	Thai	136.93	152.29	0	0														
1	1018704	2	56	TYPE_C	20	Beverages	Thai	135.83	152.29	0	0															
2	1196273	3	ss	TYPE_c	20	eeverages	Thai	132.92	133.92	0	0															
4	1343872	5	647	56	TYPE_C	20	Beverages	Thai	146.50	147.50	0	0														

Team Member 2

IBM Home Page - Select or create a notebook Code - Jupyter Notebook

localhost:8891/notebooks/Downloads/SBSPS-Challenge-8325-Food-Demand-Forecasting-for-Food-Delivery-Company-using-IBM-Cloud-main/SBSPS-Challenge-...

Gmail YouTube Maps

jupyter Code (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Connected Not Trusted Python 3 (ipykernel)

Dropping Columns

Let's drop columns "center_id" and "meal_id" as they are not required for the further process. Display the changes of trainfinal table using head().

```
In [110]: trainfinal = trainfinal.drop(['center_id', 'meal_id'], axis=1)
trainfinal.head()
```

```
Out[110]:
```

	id	week	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders	category	cuisine	city_code	region_code	center_type
0	1379560	1	136.83	152.29	0	0	177	Beverages	Thai	647	56	TYPE_C
1	1018704	2	135.83	152.29	0	0	323	Beverages	Thai	647	56	TYPE_C
2	1196273	3	132.92	133.92	0	0	96	Beverages	Thai	647	56	TYPE_C
3	1116527	4	135.86	134.86	0	0	163	Beverages	Thai	647	56	TYPE_C
4	1343872	5	146.50	147.50	0	0	215	Beverages	Thai	647	56	TYPE_C

Display the list of columns present in trainfinal table and store it in variable "cols"

```
In [111]: cols = trainfinal.columns.tolist()
print(cols)
```

```
['id', 'week', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders', 'category', 'cuisine', 'city_code', 'region_code', 'center_type', 'op_area']
```

Rearrange the columns by slicing the columns of "cols" and print "cols"

IBM Home Page - Select or create a notebook Code - Jupyter Notebook

localhost:8891/notebooks/Downloads/SBSPS-Challenge-8325-Food-Demand-Forecasting-for-Food-Delivery-Company-using-IBM-Cloud-main/SBSPS-Challenge-...

Gmail YouTube Maps

jupyter Code (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Connected Not Trusted Python 3 (ipykernel)

Rearrange the columns by slicing the columns of "cols" and print "cols"

```
In [112]: cols = cols[:2] + cols[9:] + cols[7:9] + cols[2:7]
print(cols)
```

```
['id', 'week', 'city_code', 'region_code', 'center_type', 'op_area', 'category', 'cuisine', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders']
```

Store the changes of columns in trainfinal and display the datatypes of trainfinal using trainfinal.dtypes. Here, we can see that, we not only have numerical data but we also have object data.

```
In [113]: trainfinal = trainfinal[cols]
trainfinal.head()
```

```
Out[113]:
```

	id	week	city_code	region_code	center_type	op_area	category	cuisine	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders
0	1379560	1	647	56	TYPE_C	2.0	Beverages	Thai	136.83	152.29	0	0	0
1	1018704	2	647	56	TYPE_C	2.0	Beverages	Thai	135.83	152.29	0	0	0
2	1196273	3	647	56	TYPE_C	2.0	Beverages	Thai	132.92	133.92	0	0	0
3	1116527	4	647	56	TYPE_C	2.0	Beverages	Thai	135.86	134.86	0	0	0
4	1343872	5	647	56	TYPE_C	2.0	Beverages	Thai	146.50	147.50	0	0	0

```
In [114]: trainfinal.dtypes
```

```
Out[114]: id          int64
week          int64
```

The screenshot shows a Jupyter Notebook with the following content:

```

In [113]: trainfinal = trainfinal[0:500]
          trainfinal.head()

Out[113]:
```

	id	week	city_code	region_code	center_type	op_area	category	cuisine	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders
0	1379560	1	647	56	TYPE_C	2.0	Beverages	Thai	136.83	152.29	0	0	
1	1018704	2	647	56	TYPE_C	2.0	Beverages	Thai	135.83	152.29	0	0	
2	1196273	3	647	56	TYPE_C	2.0	Beverages	Thai	132.92	133.92	0	0	
3	1116527	4	647	56	TYPE_C	2.0	Beverages	Thai	135.86	134.86	0	0	
4	1343872	5	647	56	TYPE_C	2.0	Beverages	Thai	146.50	147.50	0	0	

```

In [114]: trainfinal.dtypes

Out[114]: id                int64
week                int64
city_code           int64
region_code         int64
center_type         object
op_area            float64
category            object
cuisine             object
checkout_price      float64
base_price          float64
emailer_for_promotion int64
homepage_featured   int64
num_orders          int64
dtype: object

```

Team Member 3

The screenshot shows a Jupyter Notebook with the following content:

Dropping Columns

Let's drop columns "center_id" and "meal_id" as they are not required for the further process. Display the changes of trainfinal table using head().

```

In [110]: trainfinal = trainfinal.drop(['center_id', 'meal_id'], axis=1)
          trainfinal.head()

Out[110]:
```

	id	week	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders	category	cuisine	city_code	region_code	center_type
0	1379560	1	136.83	152.29	0	0	177	Beverages	Thai	647	56	TYPE_C
1	1018704	2	135.83	152.29	0	0	323	Beverages	Thai	647	56	TYPE_C
2	1196273	3	132.92	133.92	0	0	96	Beverages	Thai	647	56	TYPE_C
3	1116527	4	135.86	134.86	0	0	163	Beverages	Thai	647	56	TYPE_C
4	1343872	5	146.50	147.50	0	0	215	Beverages	Thai	647	56	TYPE_C

Display the list of columns present in trainfinal table and store it in variable "cols"

```

In [111]: cols = trainfinal.columns.tolist()
          print(cols)

['id', 'week', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orders', 'category', 'cuisine', 'city_code', 'region_code', 'center_type', 'op_area']

```

Rearrange the columns by slicing the columns of "cols" and print "cols"

Jupyter Code (saved)



Logout

File Edit View Insert Cell Kernel Widgets Help

Not tested

Python 3 (ipykernel)

```
Reorder columns by sorting the columns of "col" and print cols
```

```
In [112]: cols = cols[:2] + cols[9:] + cols[7:9] + cols[2:7]
```

Store the changes of columns in trainfinal and display the datatypes of trainfinal using trainfinal.dtypes. Here, we can see that, we not only have numerical but we also have object

```
Out[113]:
```

	0	137956	0	647	:m	neE_c	2.0	Be•eages	Ta:	136.03	B3	162.29		0	0
	2	1196273	3	647	ā	TTPE_c	ao	Beveages	Ta	132.92	133.92		0	0	0
	4	1343872	5	b47	:é	TTPE_c	ao	Beeages	Ta	146.50	147.50		0	0	0

```
In [ ]: trainfinal.dtypes
```

```
id      int64
veer    int64
ig      int64
code    object
region  object
code    object
ezar    object
ape     object
op      object
area    object
zaevoc  object
zuisne  object
checkout object
price   object
base    object
price   object
emaiter object
for     object
promouoz object
omep    object
ge      object
emurea  object
ni      object
```

fl O loahou889Vnoeobok/DowToads BSS-Challenge-8925FoodDema&dFoezanmg (or-FoodDeIi y-Company-using IBM-Coud manSBSP/ >Ctalenge-... Q .

Yolube Ma

Jupyter Code (saved)



Logout

File Edit View Insert Cell Kernel Widgets Help

Not tested

Python 3 (ipykernel)

```
Out[113]:
```

0	1379560	1	647	ss	TYPE_c	2.0	everages	Thai b47	136.03	152.29	0	0
1	1018704	2		56	TYPE_C	20	Beverages	Thai b47	135.83	152.29	0	0
2	1196273	3		ss	TYPE_c	2.0	everages	Thai	132.92	133.92	0	0
4	1343872	5	647	56	TYPE_C	20	Beverages	Thai	146.50	147.50	0	0