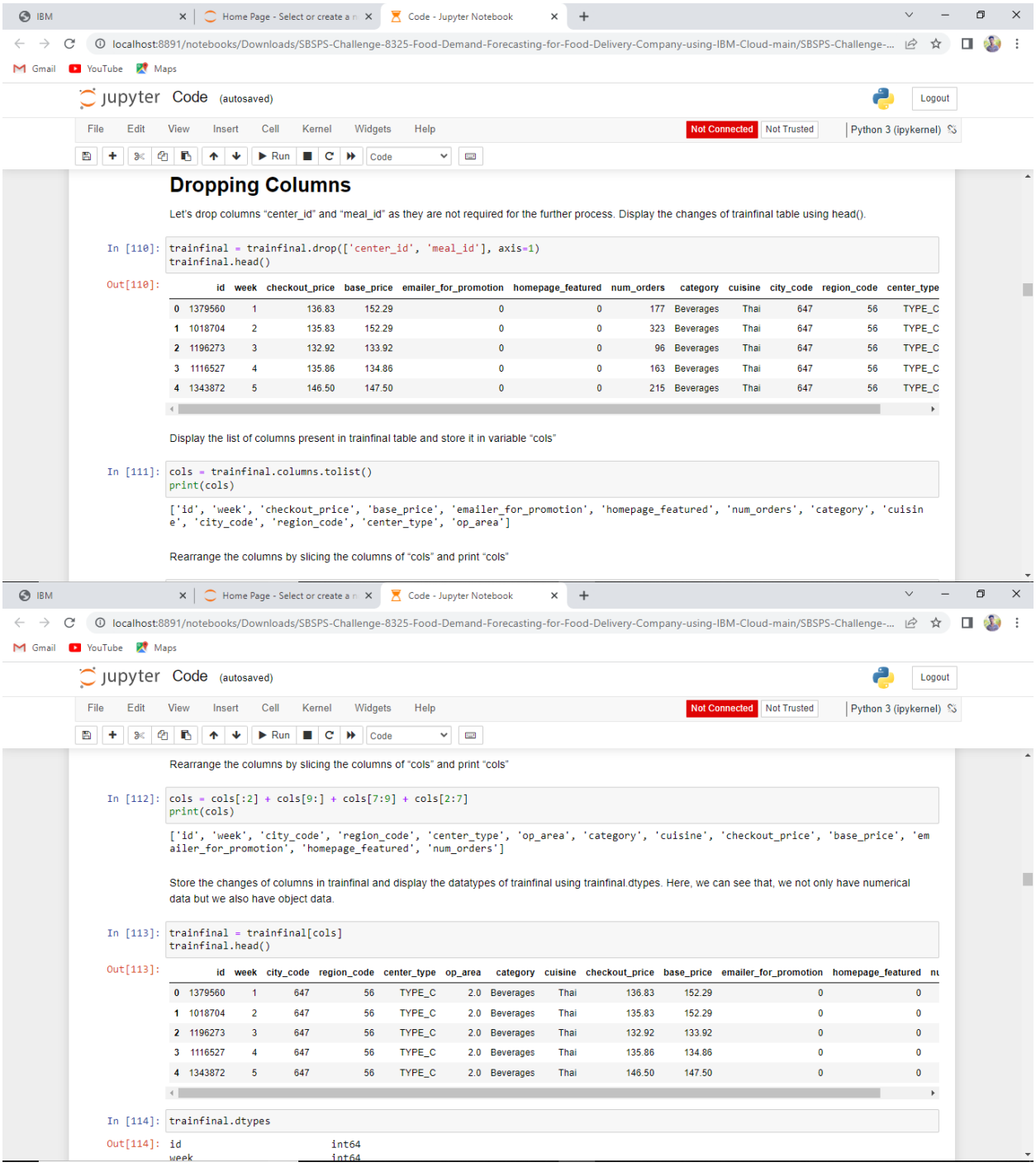
TEAM ID: PNT2022TMID14136

PROJECT NAME: DemandEst - AI powered Food Demand Forecaster

Team Leader



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', 'cuisin
e', 'city_code', 'region_code', 'center_type', 'op_area']
```

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

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', 'em
ailer_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.

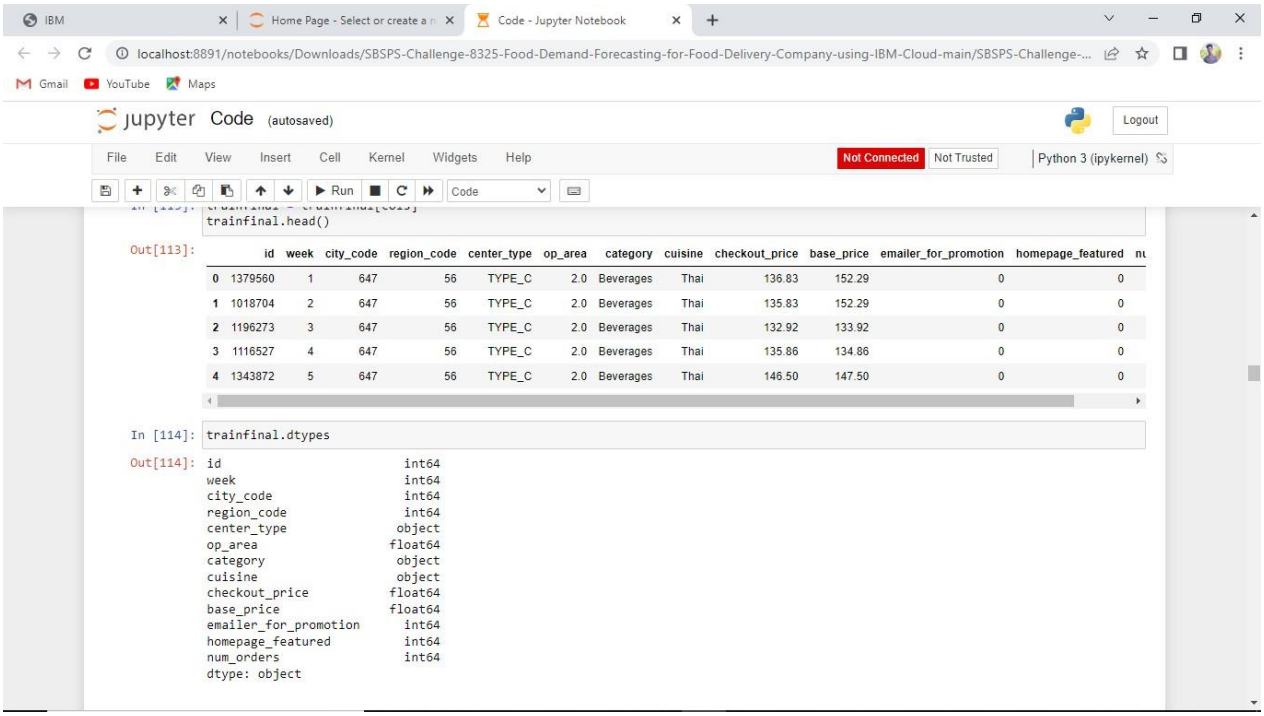
```
In [113]: trainfinal = trainfinal[cols]
          trainfinal.head()
```

Out[113]:

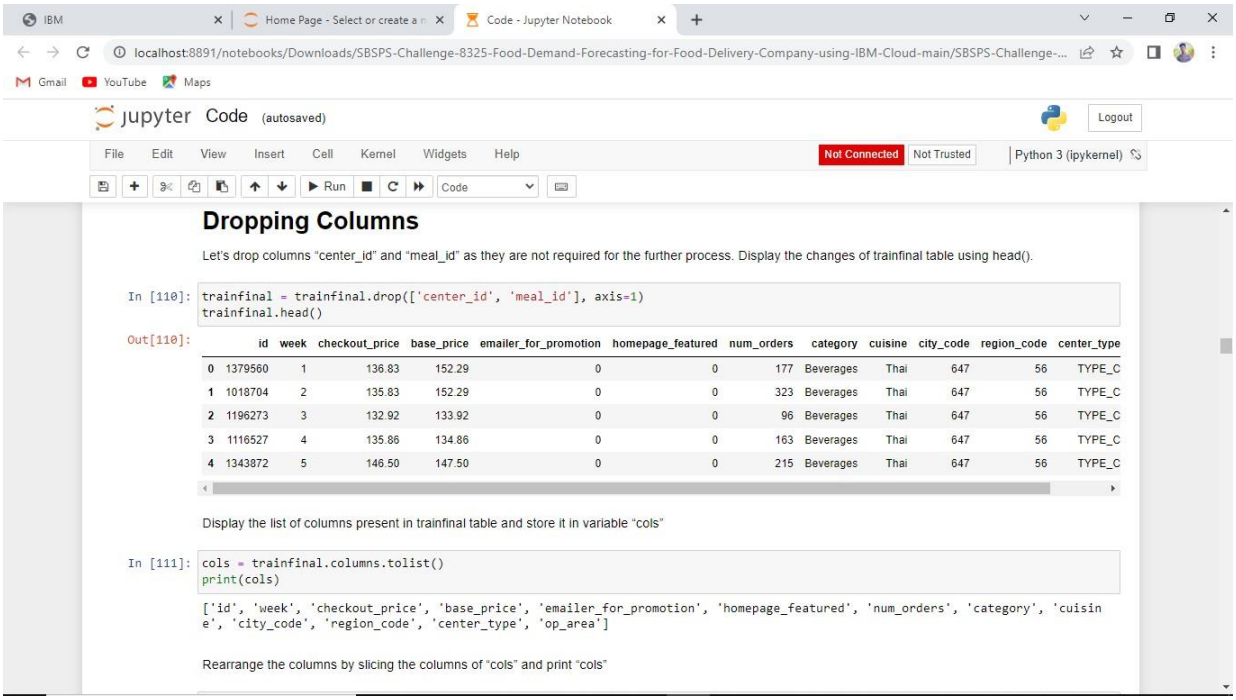
	id	week	city_code	region_code	center_type	op_area	category	cuisine	checkout_price	base_price	emailer_for_promotion	homepage_featured	nu
0	1379560	1	647	56	TYPE_C	2.0	Beverages	Thai	136.83	152.29	0	0	
1	1018704	2	647	56	TYPE_C	2.0	Beverages	Thai	135.83	152.29	0	0	
2	1196273	3	647	56	TYPE_C	2.0	Beverages	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
```



Team Member 1



jupyter Code «rosaved)

Logout

Fe EU Ve'r lseñ Cel Kernel Widgels Help Nat ted Python 3 (ipykernel)

Reanange‹ñeco«mns byslcng lñeco«mns 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]:

	io	veer	:g	‹ooe	regon	oae	‹ezmr g	op	anea	:=e	org	Dunne	zaectou‹	pros	base	prose	ema"er	Cor	promo"oz	aomep*ge	mmurea	n
0	137956	0		647		:m	neE_c	?o	Be•eages			Ta:		136	B3	162	29		0		0	
2	1196273		3	647		â	TTPE_c		ao Beveages			Ta		132.92		133.92			0		0	
4	1343872		5	b47		:é	TTPE c		ao Beeages			Ta		146.50		147.50			0		0	

I [a'nf'naI d ypes

'd 'ntd4
week 's

M Hom‹ Pag - Sev ior‹mai‹a' M Cod Ju /rMoibook $ '

fl O loahou889Vnoebook/DowToads BSS-Challenge-8925FoodDema«dFoezanmg(or-FoodDeIi y-Company-using IBM-Coud manSBSP/ ›Ctalenge-... Q

Yo lube Ma

jupyter Code «rosaved)

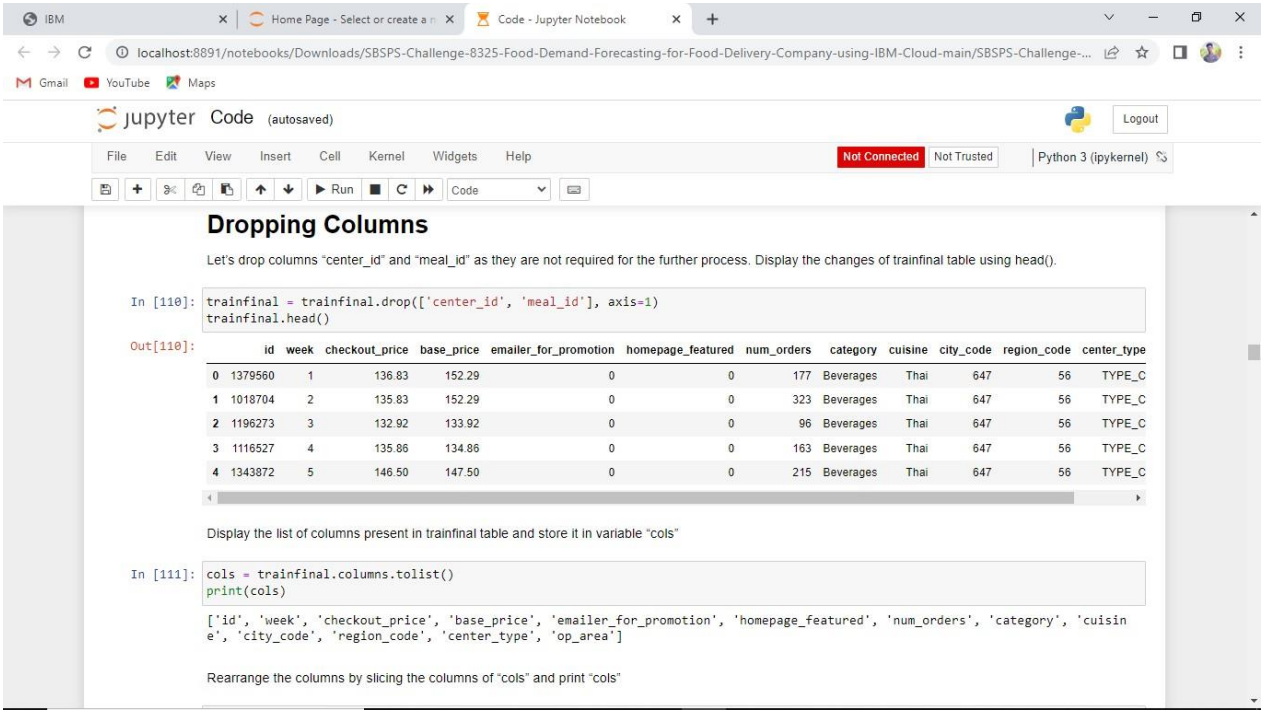
Logout

File Edu View Insert Cell Kernel Widgels Help Not ted Python 3 (ipykernel)

Out[113]:

	id	veer	‹ig	code	region	‹ode	‹ezar	ape	op	area	zaevo‹v	zuisne	‹	eckout	price	base	price	emai+er	for	promouoz	omep*ge	emmrea	ni
0	1379560		1	647		ss	TYPE_c			z.a	eeverages	Thai	b47		136.83		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.a	eeverages	Thai			132.92		133.92		0		0	
4	1343872		5	647		56	TYPE C			2 0	Beverages	Thai			146.50		147.50		0		0	

Team Member 2



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]:

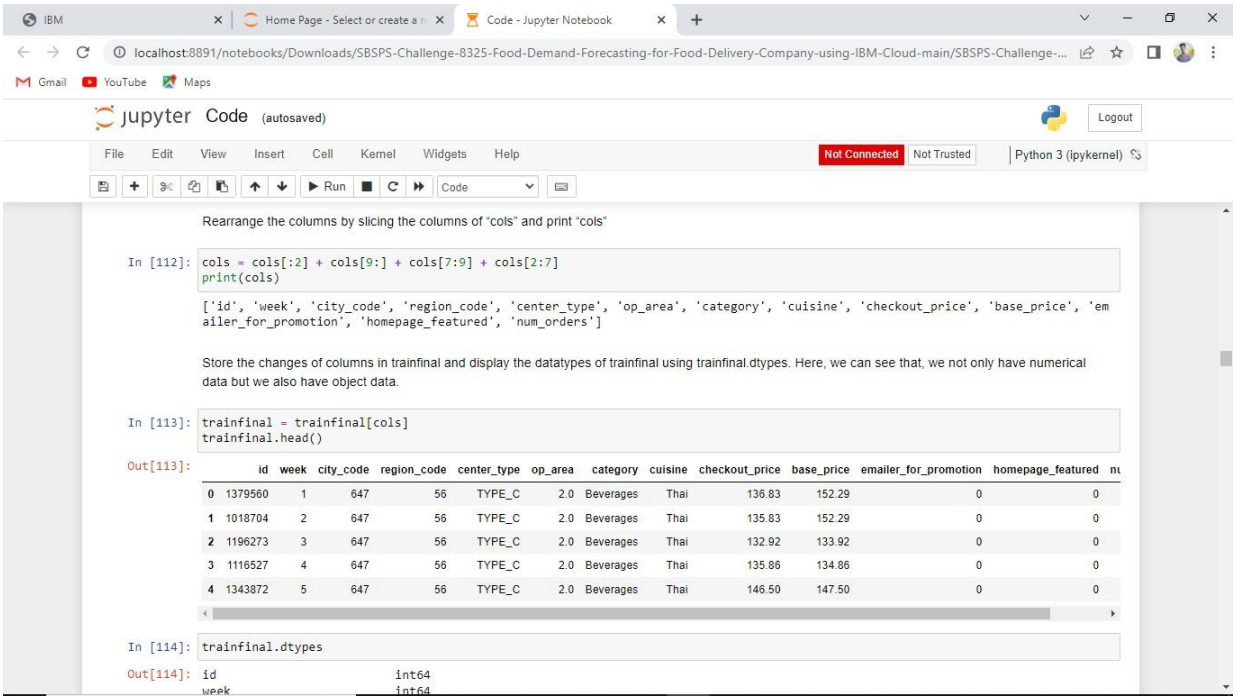
	id	week	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders	category	cuisine	city_code	region_code	center_type
0	1379560	1	136.83	152.29	0	0	177	Beverages	Thai	647	56	TYPE_C
1	1018704	2	135.83	152.29	0	0	323	Beverages	Thai	647	56	TYPE_C
2	1196273	3	132.92	133.92	0	0	96	Beverages	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', 'cuisin
e', 'city_code', 'region_code', 'center_type', 'op_area']
```

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



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', 'em
ailer_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.

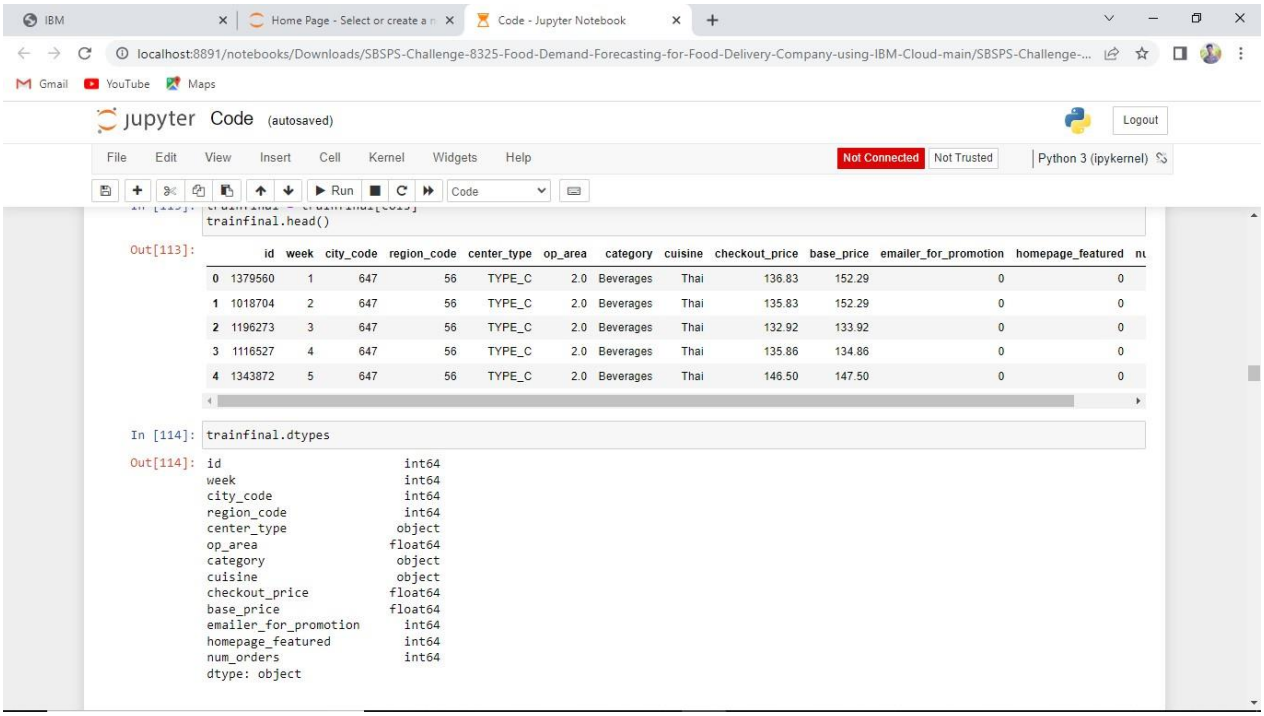
```
In [113]: trainfinal = trainfinal[cols]
          trainfinal.head()
```

Out[113]:

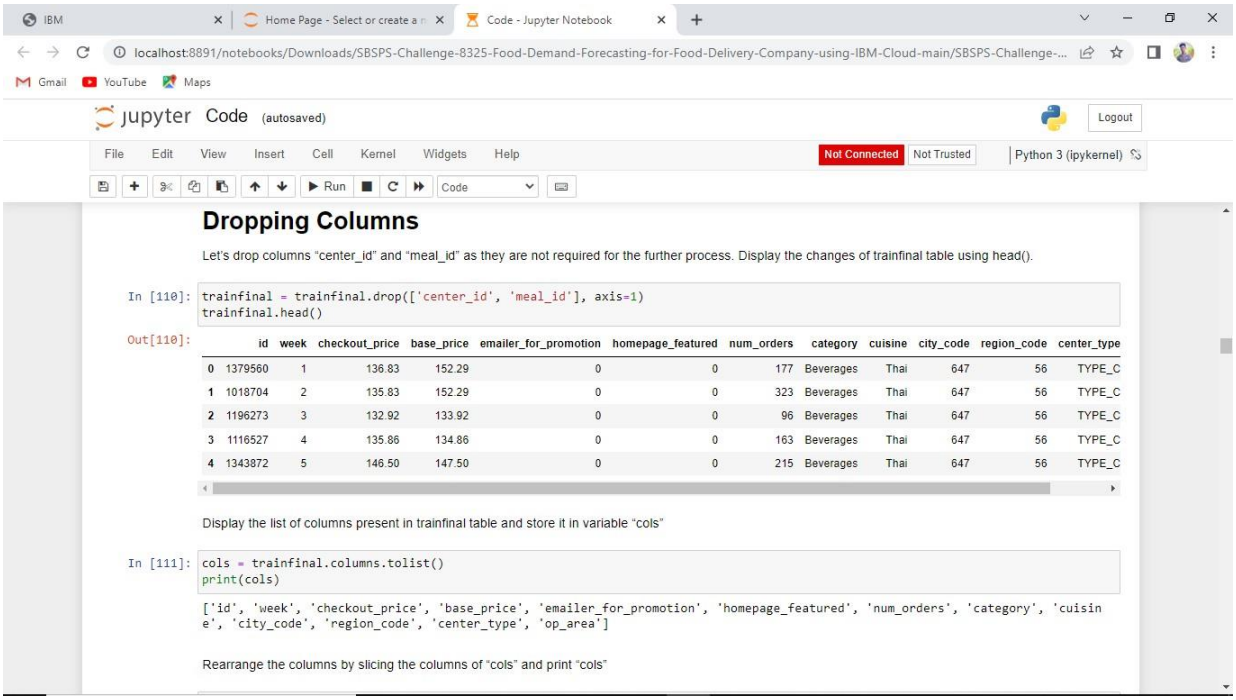
	id	week	city_code	region_code	center_type	op_area	category	cuisine	checkout_price	base_price	emailer_for_promotion	homepage_featured	nu
0	1379560	1	647	56	TYPE_C	2.0	Beverages	Thai	136.83	152.29	0	0	
1	1018704	2	647	56	TYPE_C	2.0	Beverages	Thai	135.83	152.29	0	0	
2	1196273	3	647	56	TYPE_C	2.0	Beverages	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
```



Team Member 3



jupyter Code «rosaved)

Logout

Fe EU Ve'r lseñ Cel Kernel Widgels Help Nat ted Python 3 (ipykernel)

Reanange‹ñeco«mns byslcng lñeco«mns 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]:

	io	veer	:g	‹ooe	regon	oae	‹ezmr g	op	anea	:=e	org	Dunne	zaectou‹	pros	base	prose	ema"er	Cor	promo"oz	aomep*ge	mmurea	n
0	137956	0		647		:m	neE_c		?o	Be•eages		Ta:		136	B3	162	29		0		0	
2	1196273		3	647		â	TTPE_c		ao	Beveages		Ta		132.92		133.92			0		0	
4	1343872		5	b47		:é	TTPE c		ao	Beeages		Ta		146.50		147.50			0		0	

I [a'nf'naI d ypes

'd 'ntd4
week 's

M Hom‹ Pag - Sev ior‹mai‹a' M Cod Ju /rMoibook $ '

fl O loahou889Vnoebook/DowToads BSS-Challenge-8925FoodDema«dFoezanmg(or-FoodDeIi y-Company-using IBM-Coud manSBSP/ ›Ctalenge-... Q

Yo lube Ma

jupyter Code «rosaved)

Logout

File Edu View Insert Cell Kernel Widgels Help Not ted Python 3 (ipykernel)

Out[113]:

	id	veer	‹ig	code	region	‹ode	‹ezar	ape	op	area	zaevo‹v	zuisne	‹	eckout	price	base	price	emai+er	for	promouoz	omep*ge	emmrea	ni
0	1379560		1	647		ss	TYPE_c			z.a	eeverages	Thai	b47		136.83		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.a	eeverages	Thai			132.92		133.92		0		0	
4	1343872		5	647		56	TYPE C			2 0	Beverages	Thai			146.50		147.50		0		0	