

Extract, Transform & Load on AWS platform using Python with AWS Data Wrangler

Data contains list of restaurants and their menus in USA partnered with Uber Eats.

Dataset has 2 csv files-

- 1) restaurants.csv
- 2) restaurants-menus.csv

Following task will be performed on the data:

Part A - AWS services

- 1) Create bucket in Amazon S3 and create 2 folders - *raw_data* & *cleaned_data*. Upload the files in *raw_data*.
- 2) Create dataset in Amazon Glue.
- 3) Create notebook instance in Amazon Sagemaker.
- 4) Using IAM, add policies in IAM role created in Sagemaker notebook that allows to access S3 bucket, Glue database and Athena.

Part B - Using python on Jupyter platform

- 1) Open Jupyter platform from Sagemaker notebook.
- 2) Install AWS Data Wrangler service. Fetch the raw data.
- 3) Using Pandas and NumPy in python, perform data cleaning operation.
- 4) Store the cleaned data in the bucket-folder *cleaned files* with the help of created database in Amazon Glue.
- 5) To find out insights from restaurant database, write query using query function from AWS Athena in AWS Sagemaker.

Install AWS Data Wrangler

```
In [1]: ! pip install awswrangler
```

Looking in indexes: <https://pypi.org/simple>, <https://pip.repos.neuron.amazonaws.com>

Collecting awswrangler
 Downloading awswrangler-2.16.1-py3-none-any.whl (248 kB)

 248.6/248.6 KB 4.9 MB/s eta 0:00:00a 0:00:01

Collecting numpy<2.0.0,>=1.21.0
 Downloading numpy-1.23.3-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (17.1 MB)

 17.1/17.1 MB 29.9 MB/s eta 0:00:0000:0100:01

Requirement already satisfied: pandas<2.0.0,>=1.2.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from awswrangler) (1.3.4)

Collecting requests-aws4auth<2.0.0,>=1.1.1
 Downloading requests_aws4auth-1.1.2-py2.py3-none-any.whl (24 kB)

Requirement already satisfied: boto3<2.0.0,>=1.20.17 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from awswrangler) (1.24.72)

Collecting gremlinpython<4.0.0,>=3.5.2
 Downloading gremlinpython-3.6.1-py2.py3-none-any.whl (73 kB)

 73.3/73.3 KB 18.4 MB/s eta 0:00:00

Collecting backoff<3.0.0,>=1.11.1
 Downloading backoff-2.1.2-py3-none-any.whl (14 kB)

Collecting redshift-connector<2.1.0,>=2.0.889
 Downloading redshift_connector-2.0.908-py3-none-any.whl (112 kB)

 112.0/112.0 KB 26.9 MB/s eta 0:00:00

Collecting pymysql<2.0.0,>=1.0.0
 Downloading PyMySQL-1.0.2-py3-none-any.whl (43 kB)

 43.8/43.8 KB 11.3 MB/s eta 0:00:00

Collecting progressbar2<5.0.0,>=4.0.0
 Downloading progressbar2-4.0.0-py2.py3-none-any.whl (26 kB)

Collecting jsonpath-ng<2.0.0,>=1.5.3
 Downloading jsonpath_ng-1.5.3-py3-none-any.whl (29 kB)

Collecting opensearch-py<2.0.0,>=1.0.0
 Downloading opensearch_py-1.1.0-py2.py3-none-any.whl (207 kB)

 207.5/207.5 KB 40.7 MB/s eta 0:00:00

Requirement already satisfied: pyarrow<7.1.0,>=2.0.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from awswrangler) (7.0.0)

Requirement already satisfied: botocore<2.0.0,>=1.23.17 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from awswrangler) (1.24.19)

Requirement already satisfied: openpyxl<3.1.0,>=3.0.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from awswrangler) (3.0.9)

Collecting pg8000<2.0.0,>=1.20.0
 Downloading pg8000-1.29.1-py3-none-any.whl (34 kB)

Collecting botocore<2.0.0,>=1.23.17
 Downloading botocore-1.27.75-py3-none-any.whl (9.1 MB)

 9.1/9.1 MB 33.3 MB/s eta 0:00:000:00:0100:01

Requirement already satisfied: jmespath<2.0.0,>=0.7.1 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from boto3<2.0.0,>=1.20.17->awswrangler) (0.10.0)

Requirement already satisfied: s3transfer<0.7.0,>=0.6.0 in /home/ec2-user/ana

conda3/envs/python3/lib/python3.8/site-packages (from boto3<2.0.0,>=1.20.17->aws wrangler) (0.6.0)
Requirement already satisfied: urllib3<1.27,>=1.25.4 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from botocore<2.0.0,>=1.23.17->aws wrangler) (1.26.8)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from botocore<2.0.0,>=1.23.17->aws wrangler) (2.8.2)
Collecting isodate<1.0.0,>=0.6.0
 Downloading isodate-0.6.1-py2.py3-none-any.whl (41 kB)

 41.7/41.7 KB 1.0 MB/s eta 0:00:00
Collecting aenum<4.0.0,>=1.4.5
 Downloading aenum-3.1.11-py3-none-any.whl (131 kB)

 131.5/131.5 KB 30.3 MB/s eta 0:00:00
Requirement already satisfied: aiohttp<=3.8.1,>=3.8.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from gremlinpython<4.0.0,>=3.5.2->aws wrangler) (3.8.1)
Requirement already satisfied: nest-asyncio in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from gremlinpython<4.0.0,>=3.5.2->aws wrangler) (1.5.1)
Requirement already satisfied: decorator in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from jsonpath-ng<2.0.0,>=1.5.3->aws wrangler) (5.1.0)
Requirement already satisfied: ply in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from jsonpath-ng<2.0.0,>=1.5.3->aws wrangler) (3.11)
Requirement already satisfied: six in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from jsonpath-ng<2.0.0,>=1.5.3->aws wrangler) (1.16.0)
Requirement already satisfied: et-xmlfile in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from openpyxl<3.1.0,>=3.0.0->aws wrangler) (1.0.1)
Requirement already satisfied: certifi in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from opensearch-py<2.0.0,>=1.0.0->aws wrangler) (2021.10.8)
Requirement already satisfied: pytz>=2017.3 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from pandas<2.0.0,>=1.2.0->aws wrangler) (2021.3)
Collecting scramp>=1.4.1
 Downloading scramp-1.4.1-py3-none-any.whl (8.5 kB)
Collecting python-utils>=3.0.0
 Downloading python_utils-3.3.3-py2.py3-none-any.whl (23 kB)
Requirement already satisfied: requests<2.28.1,>=2.23.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from redshift-connector<2.1.0,>=2.0.889->aws wrangler) (2.26.0)
Requirement already satisfied: beautifulsoup4<5.0.0,>=4.7.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from redshift-connector<2.1.0,>=2.0.889->aws wrangler) (4.10.0)
Requirement already satisfied: packaging in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from redshift-connector<2.1.0,>=2.0.889->aws wrangler) (21.3)
Requirement already satisfied: lxml>=4.6.5 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from redshift-connector<2.1.0,>=2.0.889->aws wrangler) (4.8.0)

Requirement already satisfied: frozenlist>=1.1.1 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from aiohttp<=3.8.1,>=3.8.0->gremlinpython<4.0.0,>=3.5.2->awswrangler) (1.2.0)

Requirement already satisfied: multidict<7.0,>=4.5 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from aiohttp<=3.8.1,>=3.8.0->gremlinpython<4.0.0,>=3.5.2->awswrangler) (5.2.0)

Requirement already satisfied: attrs>=17.3.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from aiohttp<=3.8.1,>=3.8.0->gremlinpython<4.0.0,>=3.5.2->awswrangler) (21.2.0)

Requirement already satisfied: aiosignal>=1.1.2 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from aiohttp<=3.8.1,>=3.8.0->gremlinpython<4.0.0,>=3.5.2->awswrangler) (1.2.0)

Requirement already satisfied: async-timeout<5.0,>=4.0.0a3 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from aiohttp<=3.8.1,>=3.8.0->gremlinpython<4.0.0,>=3.5.2->awswrangler) (4.0.1)

Requirement already satisfied: charset-normalizer<3.0,>=2.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from aiohttp<=3.8.1,>=3.8.0->gremlinpython<4.0.0,>=3.5.2->awswrangler) (2.0.8)

Requirement already satisfied: yarll<2.0,>=1.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from aiohttp<=3.8.1,>=3.8.0->gremlinpython<4.0.0,>=3.5.2->awswrangler) (1.7.2)

Requirement already satisfied: soupsieve>1.2 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from beautifulsoup4<5.0.0,>=4.7.0->redshift-connector<2.1.0,>=2.0.889->awswrangler) (2.3)

Requirement already satisfied: idna<4,>=2.5 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from requests<2.28.1,>=2.23.0->redshift-connector<2.1.0,>=2.0.889->awswrangler) (3.1)

Requirement already satisfied: asn1crypto>=1.4.0 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from scramp>=1.4.1->pg8000<2.0.0,>=1.20.0->awswrangler) (1.4.0)

Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from packaging->redshift-connector<2.1.0,>=2.0.889->awswrangler) (3.0.6)

Requirement already satisfied: typing-extensions>=3.6.5 in /home/ec2-user/anaconda3/envs/python3/lib/python3.8/site-packages (from async-timeout<5.0,>=4.0.0a3->aiohttp<=3.8.1,>=3.8.0->gremlinpython<4.0.0,>=3.5.2->awswrangler) (4.0.0)

Installing collected packages: aenum, scramp, python-utils, pymysql, opensearch-py, numpy, jsonpath-ng, isodate, backoff, requests-aws4auth, progressbar2, pg8000, botocore, gremlinpython, redshift-connector, awswrangler

Attempting uninstall: numpy

Found existing installation: numpy 1.20.3

Uninstalling numpy-1.20.3:

Successfully uninstalled numpy-1.20.3

Attempting uninstall: botocore

Found existing installation: botocore 1.24.19

Uninstalling botocore-1.24.19:

Successfully uninstalled botocore-1.24.19

ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency conflicts.

daal4py 2021.3.0 requires daal==2021.2.3, which is not installed.

sagemaker-pyspark 1.4.5 requires pyspark==3.3.0, but you have pyspark 3.0.0 which is incompatible.

numba 0.54.1 requires numpy<1.21,>=1.17, but you have numpy 1.23.3 which is incompatible.

```
awscli 1.25.73 requires botocore==1.27.72, but you have botocore 1.27.75 which is incompatible.
aiobotocore 2.0.1 requires botocore<1.22.9,>=1.22.8, but you have botocore 1.27.75 which is incompatible.
Successfully installed aenum-3.1.11 awscli-2.16.1 backoff-2.1.2 botocore-1.27.75 gremlinpython-3.6.1 isodate-0.6.1 jsonpath-ng-1.5.3 numpy-1.23.3 opensearch-py-1.1.0 pg8000-1.29.1 progressbar2-4.0.0 pymysql-1.0.2 python-utils-3.3.3 redshift-connector-2.0.908 requests-aws4auth-1.1.2 scrap-1.4.1
WARNING: You are using pip version 22.0.4; however, version 22.2.2 is available.
You should consider upgrading via the '/home/ec2-user/anaconda3/envs/python3/bin/python -m pip install --upgrade pip' command.
```

```
In [3]: !pip install zipfile36
```

```
Looking in indexes: https://pypi.org/simple, https://pip.repos.neuron.amazonaws.com
Collecting zipfile36
  Downloading zipfile36-0.1.3-py3-none-any.whl (20 kB)
Installing collected packages: zipfile36
Successfully installed zipfile36-0.1.3
WARNING: You are using pip version 22.0.4; however, version 22.2.2 is available.
You should consider upgrading via the '/home/ec2-user/anaconda3/envs/python3/bin/python -m pip install --upgrade pip' command.
```

Import Libraries

```
In [4]: import awscli as wr
import zipfile
```

Read Data

```
In [14]: r_menu = wr.s3.read_csv(path="s3://project-2-etl/raw_data/restaurant-menus.zip",compression="zip")
r_list = wr.s3.read_csv(path="s3://project-2-etl/raw_data/restaurants.csv")
```

```
In [15]: r_list.head(1)
```

Out[15]:

	id	position	name	score	ratings	category	price_range	full_address	zip_code	
0	1	19	PJ Fresh (224 Daniel Payne Drive)	NaN	NaN	Burgers, American, Sandwiches	\$	224 Daniel Payne Drive, Birmingham, AL, 35207	35207	33.56236

```
In [16]: r_menu.head(1)
```

```
Out[16]:
```

	restaurant_id	category	name	description	price
0	1	Extra Large Pizza	Extra Large Meat Lovers	Whole pie.	15.99 USD

Delete unnecessary columns

```
In [17]: r_list.drop(columns=['position','score','ratings'],inplace=True)
```

```
In [18]: r_list.head(1)
```

```
Out[18]:
```

	id	name	category	price_range	full_address	zip_code	lat	lng
0	1	PJ Fresh (224 Daniel Payne Drive)	Burgers, American, Sandwiches	\$	224 Daniel Payne Drive, Birmingham, AL, 35207	35207	33.562365	-86.830703

```
In [19]: r_menu.drop(columns=['description'],inplace=True)
```

```
In [20]: r_menu.head(1)
```

```
Out[20]:
```

	restaurant_id	category	name	price
0	1	Extra Large Pizza	Extra Large Meat Lovers	15.99 USD

Renaming columns

```
In [21]: r_list.rename(columns={'id':'restaurant_id','name':'restaurant_name','full_address':'address'},inplace=True)
```

```
In [22]: r_list.head(1)
```

```
Out[22]:
```

	restaurant_id	restaurant_name	category	price_range	address	zip_code	lat	lng
0	1	PJ Fresh (224 Daniel Payne Drive)	Burgers, American, Sandwiches	\$	224 Daniel Payne Drive, Birmingham, AL, 35207	35207	33.562365	-86.830703



```
In [23]: r_menu.rename(columns={'category':'menu','name':'dish_name','price':'price_USD'},inplace=True)
```

```
In [24]: r_menu.head(1)
```

```
Out[24]:
```

	restaurant_id	menu	dish_name	price_USD
0	1	Extra Large Pizza	Extra Large Meat Lovers	15.99 USD

Dropping Nan values

```
In [25]: r_list.isna().value_counts()
```

```
Out[25]: restaurant_id  restaurant_name  category  price_range  address  zip_code  lat
False                  False            False    False        False    False    Fal
se False    33399                                True        False    False    Fal
se False    6527                                False       True    True    Fal
se False    172                                True        True    True    Fal
se False    106                                True        True    True    Fal
se False    10              True    False    False    False    Fal
se False    9              True    False    False    True    Fal
se False    9                                False    True    True    Fal
se False    4                                False    False    False    Fal
dtype: int64
```

Dropping only those rows whose address and zip code is not available

```
In [26]: r_list.dropna(inplace=True)
```

```
In [27]: r_list.isna().value_counts()
```

```
Out[27]: restaurant_id  restaurant_name  category  price_range  address  zip_code  lat
False                  False            False    False        False    False    Fal
se False    33399                                False    True    True    Fal
dtype: int64
```

```
In [28]: r_menu.isna().value_counts()
```

```
Out[28]: restaurant_id  menu  dish_name  price_USD
False                  False  False      False      3375211
dtype: int64
```

Cleaning Individual Columns and Duplicate values

- Restaurant List

Check column : *restaurant_name* with *address,zip_code,lat,lng*

```
In [29]: r_list.duplicated(subset=['restaurant_name','address','zip_code','lat','lng'])  
.value_counts()
```

```
Out[29]: False    33382  
        True      17  
        dtype: int64
```

```
In [30]: r_list.drop_duplicates(subset=['restaurant_name','address','zip_code','lat','lng'],keep='last',inplace=True)
```

```
In [31]: r_list.duplicated(subset=['restaurant_name','address','zip_code','lat','lng'])  
.value_counts()
```

```
Out[31]: False    33382  
        dtype: int64
```

Check column : *zip_code* : Considering 5 digit zipcode, the 1st 5 digits were considered as zipcode; for all other cases rows were deleted

```
In [32]: zip = lambda x: x[:5]  
  
list = []  
  
for a in map(zip,r_list['zip_code']):  
    try:  
        var = int(a)  
        list.append(var)  
    except ValueError:  
        var = ''  
        list.append(var)  
r_list['zip_code']=list
```

```
In [33]: r_list = r_list[r_list['zip_code']!='']
r_list['zip_code'] = r_list['zip_code'].astype(int)
r_list.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 33377 entries, 0 to 40226
Data columns (total 8 columns):
#   Column          Non-Null Count  Dtype
---  -
0   restaurant_id    33377 non-null  int64
1   restaurant_name  33377 non-null  object
2   category         33377 non-null  object
3   price_range      33377 non-null  object
4   address          33377 non-null  object
5   zip_code         33377 non-null  int64
6   lat              33377 non-null  float64
7   lng              33377 non-null  float64
dtypes: float64(2), int64(2), object(4)
memory usage: 2.3+ MB
```

- Restaurant Menu

Check column : *restaurant_name*

```
In [34]: r_menu.duplicated().value_counts()
```

```
Out[34]: False    3345541
         True      29670
         dtype: int64
```

```
In [35]: r_menu.drop_duplicates(keep='last',inplace=True)
```

```
In [36]: r_menu.duplicated().value_counts()
```

```
Out[36]: False    3345541
         dtype: int64
```

Check column : *price* : Column will be converted into float type after removing USD.

```
In [37]: r_menu['price_USD'] = r_menu['price_USD'].str.extract(r'(^\\d*\\.\\d*)',expand=False).astype(float)
```

```
In [49]: r_menu.info()
r_menu.head(3)
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3345541 entries, 0 to 3375210
Data columns (total 4 columns):
#   Column          Dtype
---  -
0   restaurant_id   int64
1   menu            object
2   dish_name       object
3   price_usd       float64
dtypes: float64(1), int64(1), object(2)
memory usage: 127.6+ MB
```

Out[49]:

	restaurant_id	menu	dish_name	price_usd
0	1	Extra Large Pizza	Extra Large Meat Lovers	15.99
1	1	Extra Large Pizza	Extra Large Supreme	15.99
2	1	Extra Large Pizza	Extra Large Pepperoni	14.99

Storing the cleaned files in S3 bucket having cleaned_data folder

To utilise space appropriately, *r_menu* and *r_list* dataframe was stored in parquet format.

```
In [42]: wr.s3.to_parquet(df=r_list,path="s3://project-2-etl/cleaned_data/restaurant_li
st/",dataset=True,database="project-2-etl-database",table="restaurant_list")
```

```
Out[42]: {'paths': ['s3://project-2-etl/cleaned_data/restaurant_list/2f5217fa3adb4b36a
e75245b03abb972.snappy.parquet'],
'partitions_values': {}}
```

```
In [50]: wr.s3.to_parquet(df=r_menu,path="s3://project-2-etl/cleaned_data/restaurants_m
enu/",dataset=True,database="project-2-etl-database",table="restaurants_menu")
```

```
Out[50]: {'paths': ['s3://project-2-etl/cleaned_data/restaurants_menu/4d2a2bb40c924cc7
97e836ec6b8d7991.snappy.parquet'],
'partitions_values': {}}
```

Query the required results

The query can be performed using AWS Athena also. But query is performed from this notebook by integrating Athena with the help of AWS Data Wrangler.

1) Query to find restaurants starting with A/

```
In [82]: sql_query_1 = """
SELECT restaurant_name,price_range,address
FROM restaurant_list
WHERE restaurant_name LIKE 'A1%'
"""
```

```
In [83]: r_list_query = wr.athena.read_sql_query(sql_query_1,database="project-2-etl-da
tabase")
```

```
In [84]: r_list_query
```

Out[84]:

	restaurant_name	price_range	address
0	Allstar Diner	\$	5704 Birmingport Rd, Sylvan Springs, AL, 35118
1	All Original Pizzeria	\$\$	215, Helena, AL, 35080
2	Aloha Hawaiian Grill	\$\$	25771 Perdido Beach Boulevard, Orange Beach, A...
3	Albertsons Express (105 Buffalo Way)	\$	105 Buffalo Way, Jackson, WY, 83001
4	Albertsons Express (5800 Yellowstone Rd)	\$	5800 Yellowstone Rd, Cheyenne, WY, 82009
...
145	All About Breakfast (5610 North Interstate Hig...	\$\$	5610 North Interstate Highway 35, Austin, TX, ...
146	Aleida's Restaurant	\$	2011 little elm trl #106 cedar park tx78613, C...
147	All About Breakfast (200 Buttercup Creek Boule...	\$\$	200 Buttercup Creek Boulevard, Cedar Park, TX,...
148	All Star Burger (Bee Cave)	\$	12921 Hill Country Blvd, Bee Cave, TX, 78738
149	Almarah Mediterranean Cuisine (Ranch Rd)	\$	12129 Ranch Rd 620 N, Austin, TX, 78750

150 rows × 3 columns

2) Query to find Restaurants starting with *R* with *pizza* in the menu having *price* between *10 & 40 USD*

```
In [85]: sql_query_2 = """
SELECT
    restaurant_list.restaurant_name
    ,restaurants_menu.restaurant_id
    ,restaurants_menu.menu
    ,restaurants_menu.dish_name
    ,restaurants_menu.price_USD
FROM restaurant_list
INNER JOIN restaurants_menu
ON restaurant_list.restaurant_id=restaurants_menu.restaurant_id
WHERE (price_USD BETWEEN 10 AND 40) AND menu LIKE '%pizza%' AND restaurant_na
me LIKE 'R%'
"""
```

```
In [86]: r_comb_query = wr.athena.read_sql_query(sql_query_2,database="project-2-etl-da
tabase")
```

```
In [87]: r_comb_query
```

Out[87]:

	restaurant_name	restaurant_id	menu	dish_name	price_usd
0	Romio's Pizza & Pasta (Redmond)	9343	Italian specialty pizzas	Zorba Specialty Pizza - 18" - X-Large	33.19
1	Romio's Pizza & Pasta (Redmond)	9343	Italian specialty pizzas	Zorba Specialty Pizza - 15" - Large	27.85
2	Romio's Pizza & Pasta (Redmond)	9343	Italian specialty pizzas	Zorba Specialty Pizza - 12" - Medium	22.49
3	Romio's Pizza & Pasta (Redmond)	9343	Italian specialty pizzas	Zorba Specialty Pizza - 10" - Small	17.15
4	Romio's Pizza & Pasta (Redmond)	9343	Italian specialty pizzas	Zeus Specialty Pizza - 18" - X-Large	33.19
...
81	Rocco's	15345	Specialty Pizzas, 20" or Half of a 20" pizza	Cousin Margaret (Margherita)	31.00
82	Rocco's	15345	Specialty Pizzas, 20" or Half of a 20" pizza	Chilango	37.00
83	Rocco's	15345	Specialty Pizzas, 20" or Half of a 20" pizza	Chicken Pesto	39.00
84	Rocco's	15345	Specialty Pizzas, 20" or Half of a 20" pizza	BBQ Pizza	39.00
85	Rocco's	15345	Specialty Pizzas, 20" or Half of a 20" pizza	Banh Mi Pizza	39.00

86 rows × 5 columns