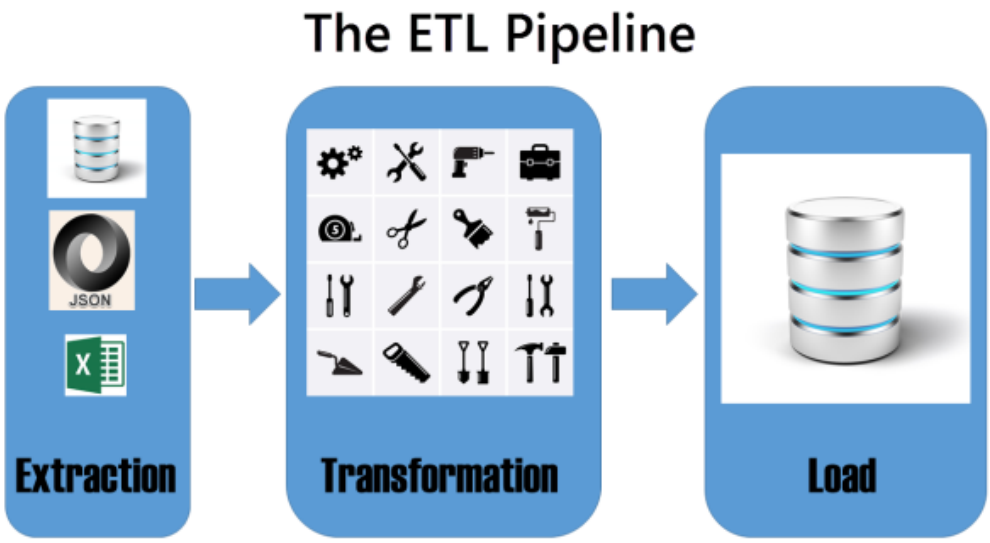
In [133]:
```
# Development of an ETL pipeline using pandas

import pandas as pd     #for Dataframe + transformation
```

In [134]:
```
from IPython.display import Image

# get the image
Image(url="https://i1.wp.com/www.timmitchell.net/wp-content/uploads/2017/01/et
```

Out[134]:



In [135]:
```
import mysql.connector

'''Programming languages like Python need a special driver before they can spe
MySQL Connector/Python enables Python programs to access MySQL databases, usin
dat is compliant wif the Python Database API. It is written in pure Python.
This module does not come built-in with Python. To install it type the below c

pip install mysql-connector-python'''
```

Out[135]:
```
'Programming languages like Python need a special driver before they can spea
k to a database\nMySQL Connector/Python enables Python programs to access MyS
QL databases, using an API\ndat is compliant wif the Python Database API. It
is written in pure Python.\nThis module does not come built-in with Python. T
o install it type the below command in the terminal.\n\npip install mysql-con
nector-python'
```

In [136]:
```
mydb = mysql.connector.connect(
  host="localhost",
  user="root",
  password="abhayjr11"
)

print(mydb)
```

```
<mysql.connector.connection_cext.CMySQLConnection object at 0x000001B7CF4AEE90>
```

Extract

In [93]:
```
# Read Data from CSV file(source)
data = pd.read_csv('C:\\work\\ETL\\books.csv')
```

In [94]:
```
# show the metadata (info about dataset)
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 700 entries, 0 to 699
Data columns (total 7 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   Name         700 non-null    object 
 1   Author       700 non-null    object 
 2   User Rating  700 non-null    float64
 3   Reviews      700 non-null    int64  
 4   Price        700 non-null    int64  
 5   Year         700 non-null    int64  
 6   Genre        700 non-null    object 
dtypes: float64(1), int64(3), object(3)
memory usage: 38.4+ KB
```

In [95]:
```
# analysis the data
data.head()
```

Out[95]:

	Name	Author	User Rating	Reviews	Price	Year	Genre
0	Act Like a Lady. Think Like a Man: What Men Re...	Steve Harvey	4.6	5013	17	2009	Non Fiction
1	Arguing with Idiots: How to Stop Small Minds a...	Glenn Beck	4.6	798	5	2009	Non Fiction
2	Breaking Dawn (The Twilight Saga. Book 4)	Stephenie Meyer	4.6	9769	13	2009	Fiction
3	Crazy Love: Overwhelmed by a Relentless God	Francis Chan	4.7	1542	14	2009	Non Fiction
4	Dead And Gone: A Sookie Stackhouse Novel (Sook...	Charlaine Harris	4.6	1541	4	2009	Fiction

transform

In [96]:
```
# find the null fields
data.isnull().sum()

# output must be zero for all fields then only we can proceed to insert the da
# if there is/are null value then we can remove it or can give random value(ma

''' all the values the non zero data is healty and clean'''
```

Out[96]:
```
' all the values the non zero data is healty and clean'
```

In [97]:
```
# Drop duplicates records
data.drop_duplicates(subset=None, keep='first', inplace=False, ignore_index=Fa
```

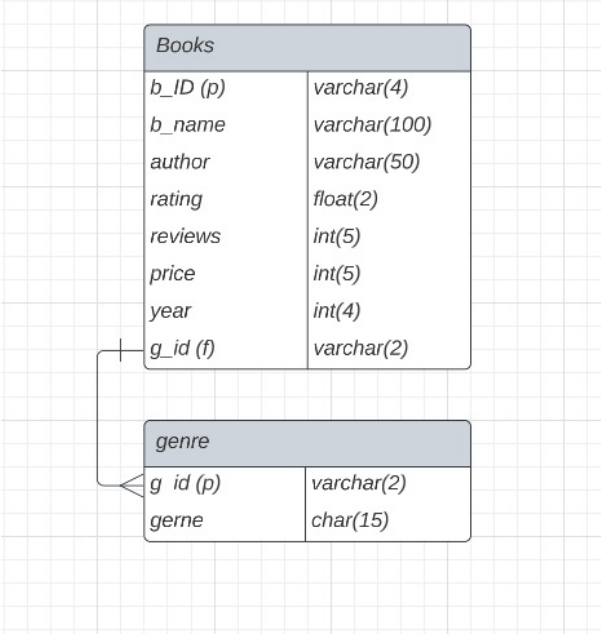
Out[97]:

	Name	Author	User Rating	Reviews	Price	Year	Genre
0	Act Like a Lady. Think Like a Man: What Men Re...	Steve Harvey	4.6	5013	17	2009	Non Fiction
1	Arguing with Idiots: How to Stop Small Minds a...	Glenn Beck	4.6	798	5	2009	Non Fiction
2	Breaking Dawn (The Twilight Saga. Book 4)	Stephenie Meyer	4.6	9769	13	2009	Fiction
3	Crazy Love: Overwhelmed by a Relentless God	Francis Chan	4.7	1542	14	2009	Non Fiction
4	Dead And Gone: A Sookie Stackhouse Novel (Sook...	Charlaine Harris	4.6	1541	4	2009	Fiction
...	...	...	...	...	...	...	...
695	The Wonderful Things You Will Be	Emily Winfield Martin	4.9	20920	9	2022	Fiction
696	Ugly Love: A Novel	Colleen Hoover	4.7	33929	10	2022	Fiction
697	Verity	Colleen Hoover	4.6	71826	11	2022	Fiction
698	What to Expect When You're Expecting	Heidi Murkoff	4.8	27052	13	2022	Non Fiction
699	Where the Crawdads Sing	Delia Owens	4.8	208917	10	2022	Fiction

700 rows × 7 columns

In [98]:
```
# Designing and creating database & tables
# get the image
Image(filename='C:\work\ETL\dataetl.jpg',width=300, height=400)
```

Out[98]:



In [99]:
```
# create a cursor and Print the databases (list of databses)
mycursor = mydb.cursor()

# Teh Cursor of the mysql-connector Python module is used to execute statement

mycursor.execute("show databases")
myresult = mycursor.fetchall()

for x in myresult:
    print(x)
```

```
('information_schema',)
('mysql',)
('performance_schema',)
('sakila',)
('sys',)
('topbooks',)
('world',)
```

In [16]:
```
# create a database

mycursor.execute("CREATE DATABASE topbooks")
```

#cross check werther database is created or not :-

mycursor = mydb.cursor() mycursor.execute("show databases") myresult = mycursor.fetchall()

for x in myresult: print(x)

In [155]:
```
#select database to work in
mydb = mysql.connector.connect(
  host="localhost",
  user="root",
  password="abhayjr11",
    database="topbooks"
)

mycursor = mydb.cursor()
```

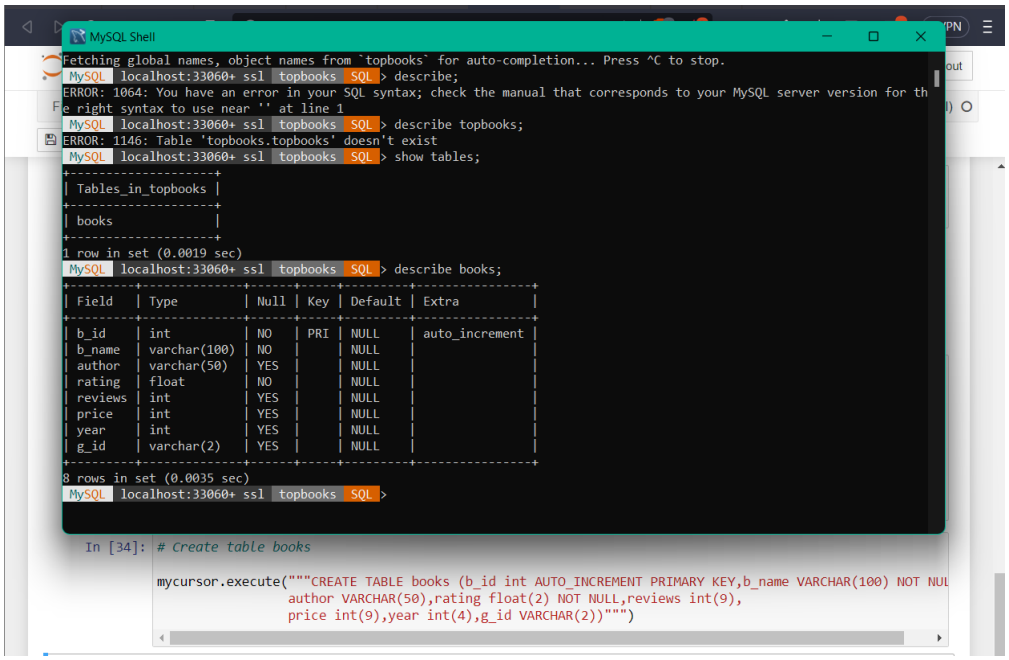
In [43]:
```
# Create table genre

mycursor.execute("""CREATE TABLE genre (gr_id int PRIMARY KEY, gerne VARCHAR(2
```

In [50]:
```
# Create table books

mycursor.execute("""CREATE TABLE books (b_id int AUTO_INCREMENT PRIMARY KEY,b_
                 author VARCHAR(50),rating float(2) NOT NULL,reviews int(9),
                 price int(9),year int(4), g_id int(2))""")

#forign key added manually
```

Load

In [101]:
```
#table is created
Image(filename='C:\work\ETL\Scre.png',width=500, height=400)
```

Out[101]:



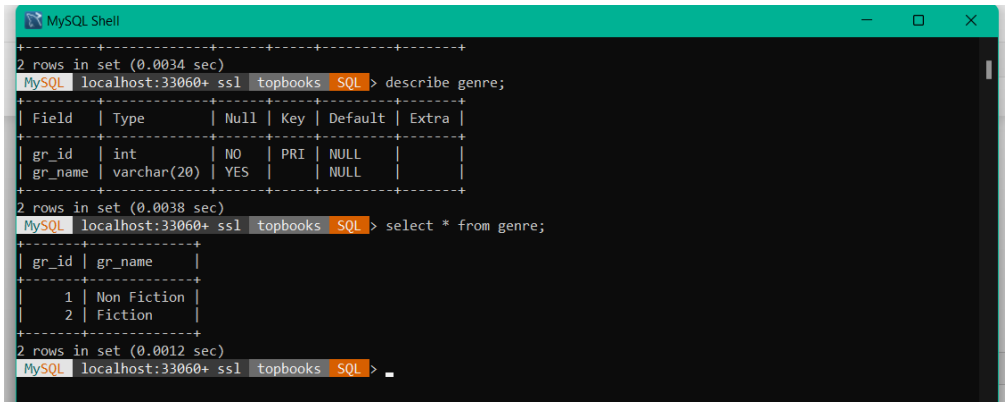
In [102]:
```
data['Genre'].unique()
```

Out[102]:
```
array(['Non Fiction', 'Fiction'], dtype=object)
```

In [141]:
```
# inster data into table genre
d1=tuple([1,'Non Fiction'])
d2=tuple([2,'Fiction'])
sql = "INSERT INTO topbooks.genre( gr_id, gr_name) VALUES(%s,%s)"
mycursor.execute(sql, d1)
mycursor.execute(sql, d2)
mydb.commit()
```

In [142]:
```
Image(filename='C:\work\ETL\Scr.png',width=500, height=400)
```

Out[142]:

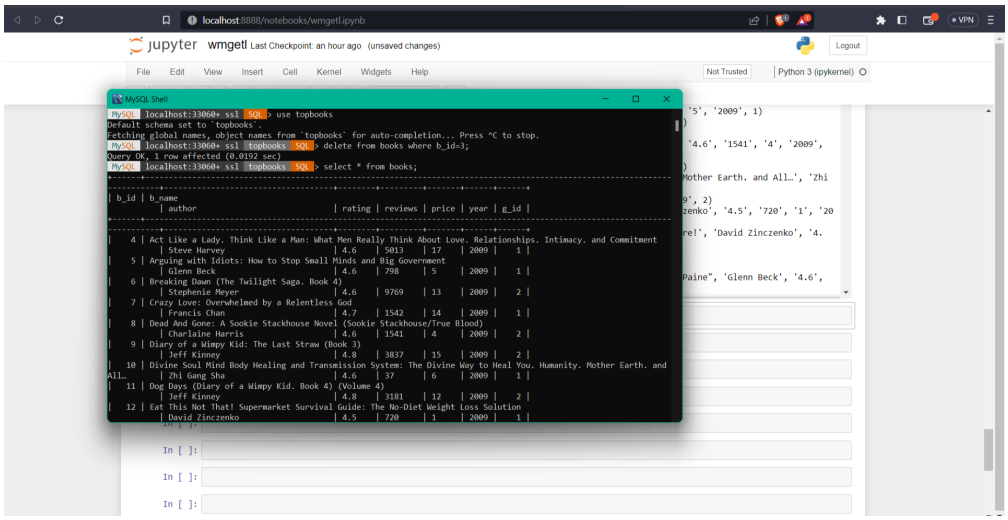


In [156]:
```
#inserting data into table group
with open('C:\\work\\ETL\\books.csv','r',encoding="utf-8",errors="ignore") as
    i=0
    data=file.readlines()
    for row in data:
        list=row.split(",")
        #print(list[6])
        if list[6]=='Genre\n':
            continue
        if list[6]=='Fiction\n':
            gid=2
        else:
            gid=1
        top=tuple([list[0],list[1],list[2],list[3],list[4],list[5],gid])
        i+=1
        print(top)
        sql = "INSERT INTO topbooks.books( b_name,author,rating,reviews,price,
        mycursor.execute(sql, top)
        mydb.commit()
```

```
Heal You. Humanity. Mother Earth. and All…', 'Zhi Gang Sha', '4.6', '37',
'6', '2009', 1)
('Dog Days (Diary of a Wimpy Kid. Book 4) (Volume 4)', 'Jeff Kinney', '4.
8', '3181', '12', '2009', 2)
('Eat This Not That! Supermarket Survival Guide: The No-Diet Weight Loss S
olution', 'David Zinczenko', '4.5', '720', '1', '2009', 1)
('Eat This. Not That! Thousands of Simple Food Swaps that Can Save You 10.
20. 30 Pounds--or More!', 'David Zinczenko', '4.3', '956', '14', '2009',
1)
('Eclipse (Twilight Sagas)', 'Stephenie Meyer', '4.7', '5505', '7', '200
9', 2)
('Eclipse (Twilight)', 'Stephenie Meyer', '4.7', '5505', '18', '2009', 2)
("Glenn Beck's Common Sense: The Case Against an Out-of-Control Governmen
t. Inspired by Thomas Paine", 'Glenn Beck', '4.6', '1365', '11', '2009',
1)
('Going Rogue: An American Life', 'Sarah Palin', '4.6', '1636', '6', '200
9', 1)
("Good to Great: Why Some Companies Make the Leap and Others Don't", 'Jim
Collins', '4.5', '3457', '14', '2009', 1)
('Have a Little Faith: A True Story', 'Mitch Albom', '4.8', '1930', '4',
```

In [157]:
```
# records instered into books table
Image(filename='C:\work\ETL\Screen.png',width=500, height=400)
```

Out[157]:



Done

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: