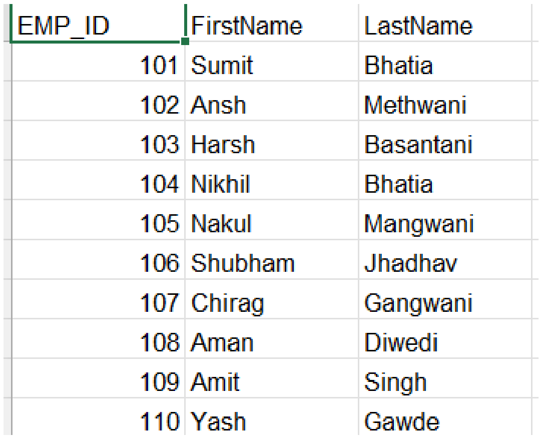
PRACTICAL NO. 3

Aim: Execute a code for implementing ETL in Python. (PETL - Python’s ETL Library)

Part A: Excel to SQL

Step 1: Create an excel sheet and name it as Employee with respective columns



EMP_ID	FirstName	LastName
101	Sumit	Bhatia
102	Ansh	Methwani
103	Harsh	Basantani
104	Nikhil	Bhatia
105	Nakul	Mangwani
106	Shubham	Jhadhav
107	Chirag	Gangwani
108	Aman	Diwedi
109	Amit	Singh
110	Yash	Gawde

Step 2: Write a python script to load this excel and send it to the SQL Server in the SSM Studio

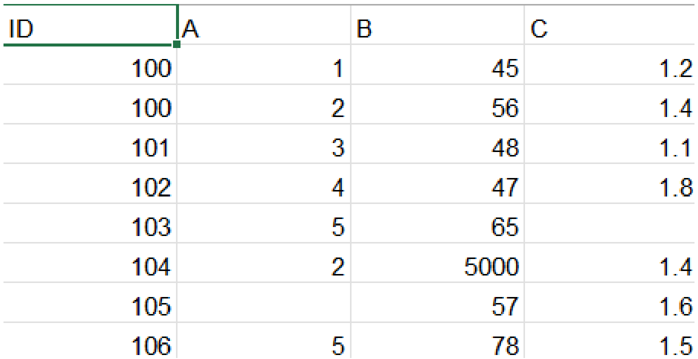
Code :-

```
import pandas as pd
import sqlalchemy as sa
import pyodbc
print(pyodbc.drivers())

#extracting data from excel
data=pd.read_excel("D:\\TYCS\\DWDM\\employee.xlsx")
print(data)


#transforming data into new clm
data['full name']=data["first name"]+' '+data["last name"]
print(data)

#loading data in sql engine
engine=sa.create_engine('mssql+pyodbc://ASUS-27/DWDM?driver=ODBC Driver 17 for SQL Server')
data.to_sql(name='emp',con=engine,index=False,if_exists='fail')
```

Step 3: Check the SSM Studio’s SQL Server for the Employee table with the new column “Full Name”

Part B: Excel to Excel

Step 1: Create new excel file with missing values and inconsistent data and name it as Sample



ID	A	B	C
100	1	45	1.2
100	2	56	1.4
101	3	48	1.1
102	4	47	1.8
103	5	65	
104	2	5000	1.4
105		57	1.6
106	5	78	1.5

Step 2: Write the python script to perform ETL and save transformed data in new excel file

code:-

```
import pandas as pd
df =
pd.read_excel("D:\\TYCS\\DWDM\\sa
mple.xlsx")
print("original dataset")
print(df)

def fill_missing_values(df):
  for col in
df.select_dtypes(include=["int","float"]
).columns:
    val = df[col].mean()
    df.fillna({col:val},inplace=True)
  return df

def drop_duplication(df,column_list):
  df =
df.drop_duplicates(subset=column_li
st)
  return df

def remove_outliners(df,column_list):
  for col in column_list:
    avg=df[col].mean()
    std=df[col].std()
    low=avg-2*std
    high=avg+2*std

df=df[df[col].between(low,high,inclusi
ve="both")]
  return df
```

```
def_processed =
(df.pipe(fill_missing_values)

.pipe(drop_duplication,"id")

.pipe(remove_outliners,["sem1","sem2","sem2"])
)

def_processed.to_excel("dwdh2.xlsx")
```

Step 3: Check the new processed data in newly form excel file