Name: Shivprasad CP

Roll No: 9696

Batch: A

Experiment 2

**Implement SQL queries for OLAP operations:**
**Part 1**

```
create table dim_time(time_key serial primary key, days date, weeks int, months int, quarter int,
years int);
select * from dim_time;

insert into dim_time(days, weeks, months, quarter, years)
values  ('2016-03-08', 33, 8, 3, 2016),
('2018-06-27', 27, 26, 6, 2018),
('2019-04-30', 30, 18, 4, 2019),
('2015-06-01', 1, 22, 6, 2015),
('2015-12-06', 6, 49, 12, 2015),
('2012-12-21', 21, 51, 12, 2012);

create table dim_location(location_key serial primary key, street varchar, city varchar, states varchar,
country varchar);
select * from dim_location;

insert into dim_location(street, city, states, country)
values ('gully chowl', 'Navi Mumbai', 'Maharashtra', 'India'),
('Film city road', 'Mumbai', 'Maharashtra', 'India'),
('Sector 24', 'Noida', 'U.P', 'India'),
('Aul market road', 'Patamundi', 'Orrisa', 'India'),
('Shivaji Chowk', 'Pune', 'Maharashtra', 'India');

insert into dim_location(street, city, states, country)
values ('New market', 'Noida', 'U.P', 'India');

create table dim_treatment(treatment_key serial primary key, treatment_name varchar, costs int,
medicine_info varchar);
select * from dim_treatment;

insert into dim_treatment(treatment_name, costs, medicine_info)
values('Root Canal', 2000, 'Yes'),
('Braces', 2500, 'No'),
('Teeth whitening', 3000, 'Yes'),
('Root Canal', 2500, 'Yes'),
('Wisdom extract', 4000, 'No'),
('Root Canal', 3000, 'Yes');
```

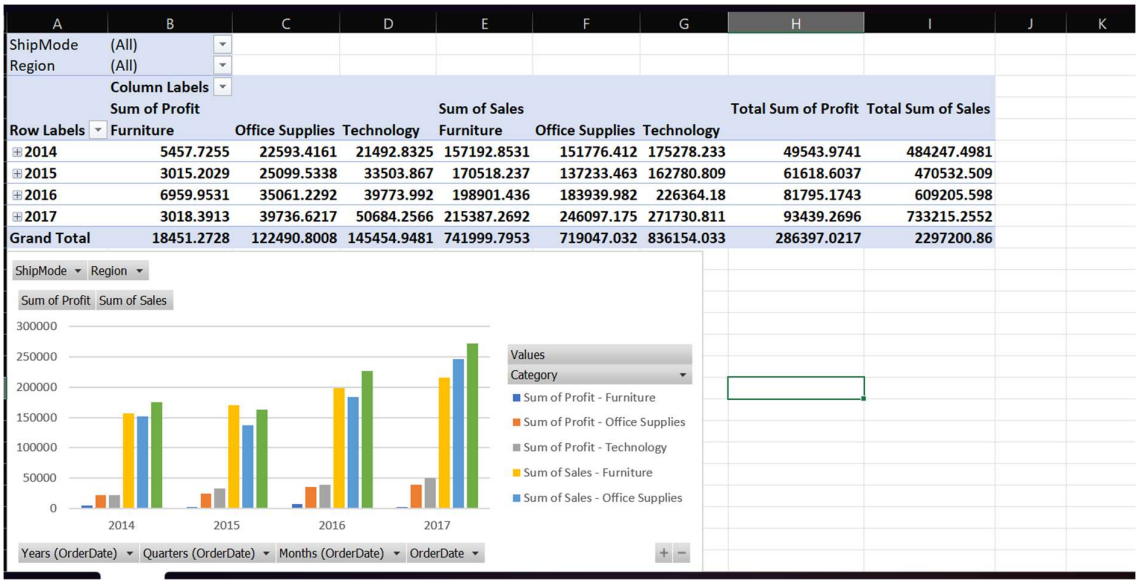
```
create table dim_diagonse(diagonse_key serial primary key, diagonse_info varchar, Doctor_key int);
select * from dim_diagonse;

insert into dim_diagonse(diagonse_info, Doctor_key)
values('Root Canal', 20),
('Braces', 21),
('Teeth whitening', 40),
('Root Canal', 72),
('Wisdom extract', 34),
('Root Canal', 40);

create table fact_revenue(time_key int references dim_time(time_key),
                location_key int references dim_location(location_key),
                treatment_key int references dim_treatment(treatment_key),
                diagonse_key int references dim_diagonse(diagonse_key),
                net_revenue decimal(19,4), total_diagonse int,
                primary key(time_key, location_key, treatment_key, diagonse_key)
               );

insert into fact_revenue values(1,1,1,1,20000.50, 10),
(2,2,2,2,10000.50, 5),
(3,3,3,3,15000.50, 8),
(4,4,4,4,22000.50, 12),
(5,5,5,5,24000.50, 15),
(6,6,6,6,18000.50, 9);

select * from fact_revenue;

drop table fact_revenue;

-- Roll Up Operations
select city, states, sum(net_revenue) from dim_location inner join
fact_revenue on dim_location.location_key = fact_revenue.location_key
group by rollup(states, city) order by states,city;

--Cube operations

select years, quarter, total_diagonse, sum(net_revenue) from fact_revenue natural inner join
dim_time
group by cube(years, quarter, total_diagonse);

--Slice Operations
select states, city, sum(net_revenue) from dim_location inner join fact_revenue on
fact_revenue.location_key = dim_location.location_key where states = 'Maharashtra' group by states,
city;

--Dice Operations
select states, city, sum(net_revenue) from dim_location inner join fact_revenue on
fact_revenue.location_key = dim_location.location_key where states = 'Maharashtra' and city =
'Mumbai'
group by states, city;
```

Part 2:

Perform data analysis with visualization for the following:

a)To view monthly, quarterly, yearly profit, sales of each category, region wise

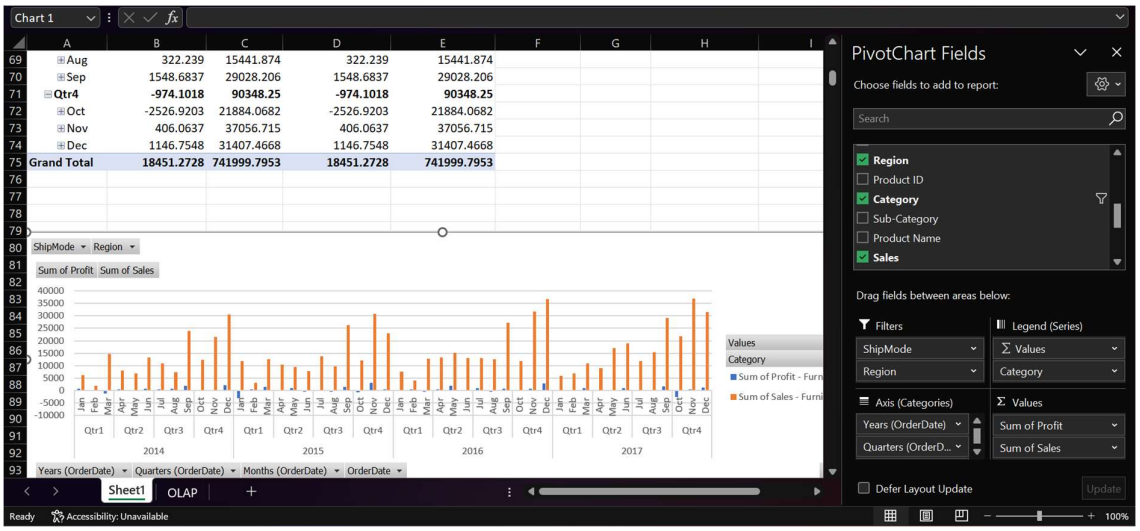
b) Comparison of sales and profit on various years.



| ShipMode | (All) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Region | (All) | | | | | | | | |
| | Column Labels | | | | | | | | |
| | Sum of Profit | | | Sum of Sales | | | Total Sum of Profit | Total Sum of Sales | |
| Row Labels | Furniture | Office Supplies | Technology | Furniture | Office Supplies | Technology | | | |
| 2014 | 5457.7255 | 22593.4161 | 21492.8325 | 157192.8531 | 151776.412 | 175278.233 | 49543.9741 | 484247.4981 | |
| 2015 | 3015.2029 | 25099.5338 | 33503.867 | 170518.237 | 137233.463 | 162780.809 | 61618.6037 | 470532.509 | |
| 2016 | 6959.9531 | 35061.2292 | 39773.992 | 198901.436 | 183939.982 | 226364.18 | 81795.1743 | 609205.598 | |
| 2017 | 3018.3913 | 39736.6217 | 50684.2566 | 215387.2692 | 246097.175 | 271730.811 | 93439.2696 | 733215.2552 | |
| Grand Total | 18451.2728 | 122490.8008 | 145454.9481 | 741999.7953 | 719047.032 | 836154.033 | 286397.0217 | 2297200.86 | |

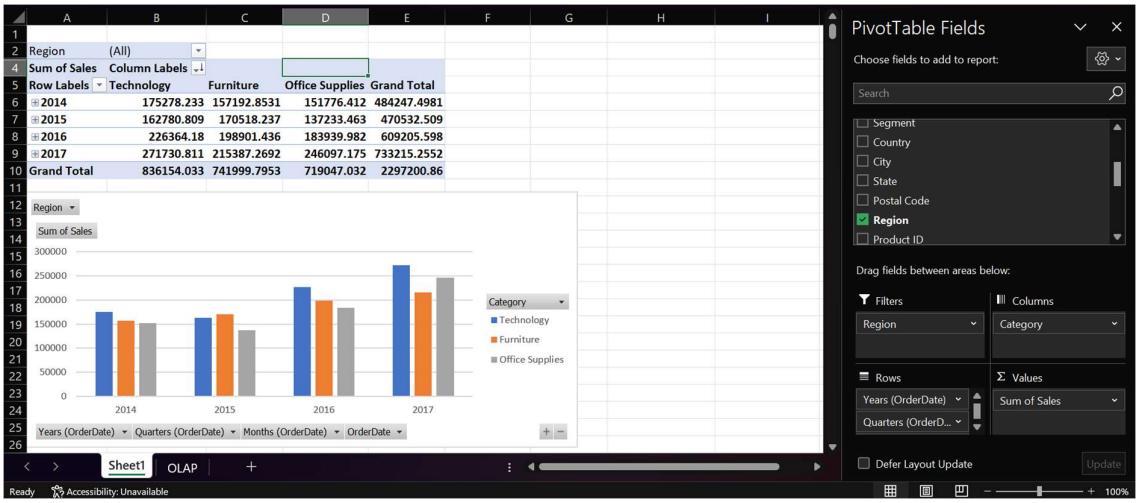


c) Comparison of sales in various months for product category =furniture.

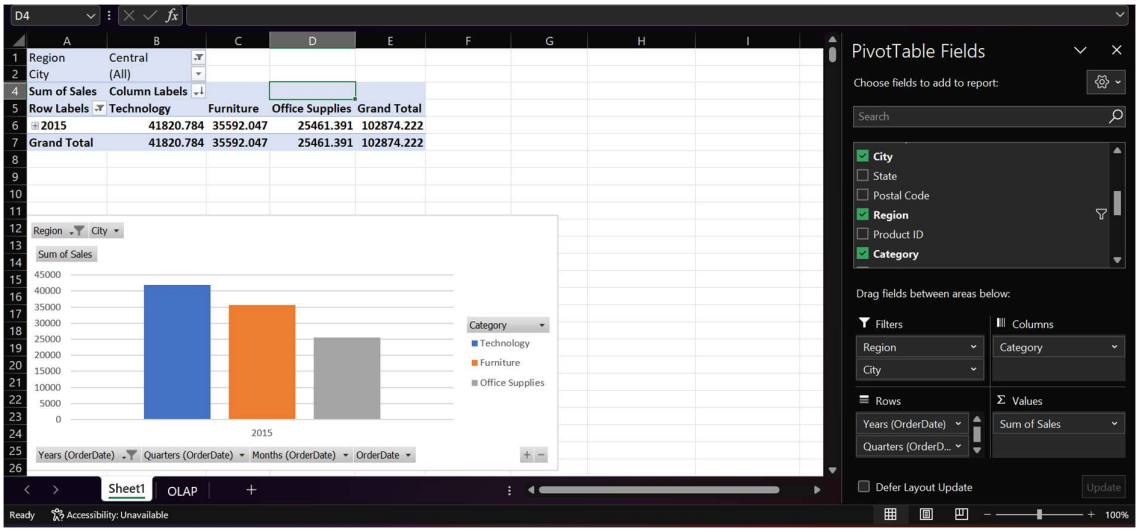


| Aug | 322.239 | 15441.874 | 322.239 | 15441.874 |
|---|---|---|---|---|
| Sep | 1548.6837 | 29028.206 | 1548.6837 | 29028.206 |
| Qtr4 | -974.1018 | 90348.25 | -974.1018 | 90348.25 |
| Oct | -2526.9203 | 21884.0682 | -2526.9203 | 21884.0682 |
| Nov | 406.0637 | 37056.715 | 406.0637 | 37056.715 |
| Dec | 1146.7548 | 31407.4668 | 1146.7548 | 31407.4668 |
| Grand Total | 18451.2728 | 741999.7953 | 18451.2728 | 741999.7953 |

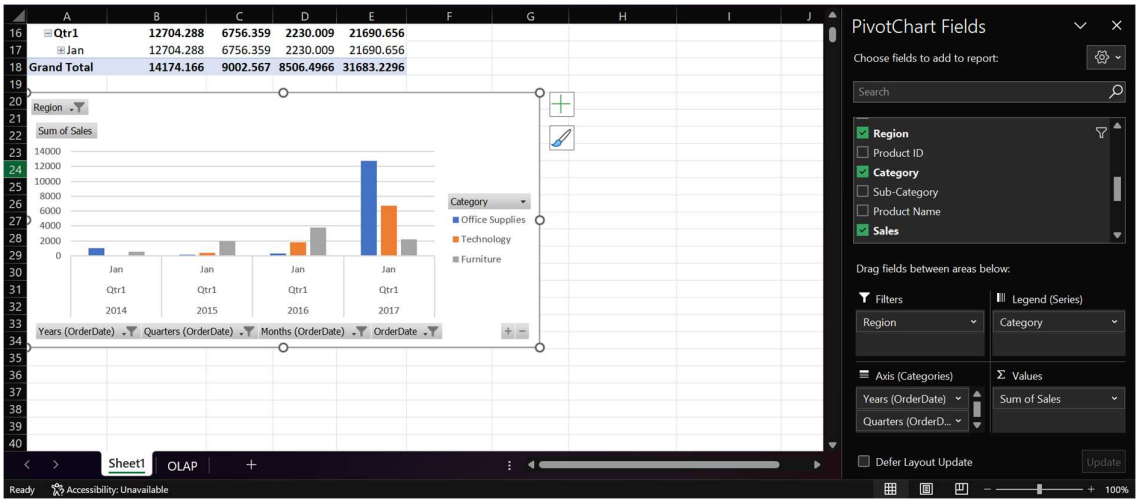
d) Need to know which product has more demand on which location?



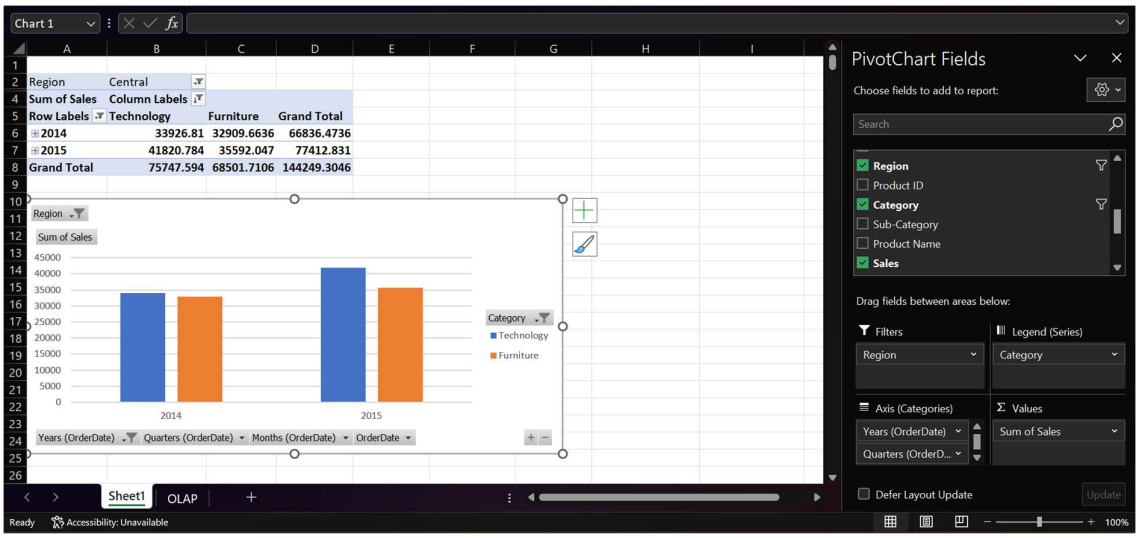

e) What is sale of product category wise, city wise for year=2015 ?




f) What is sales of product categories of Jan month of each year?




g) What is trend of sales on year 2014 and 2015 for product category furniture and technology?




Part 3:

List down the data Extraction Transformation and Loading processes applicable for your DW system.

Extract:

- Data Source Identification: Different sources of dental clinical data, which could include electronic health records, appointment systems, billing systems, patient information, etc.
- Data Extraction: Extract relevant data from the identified sources using appropriate methods, such as database queries, flat file exports, etc.
- Data Profiling: Analyze and profile the extracted data to understand its quality, structure, and potential issues.

Transform:

- Data Cleansing: Cleaning the extracted data by identifying missing values, inconsistencies, errors, and duplicates.

- Data Transformation: Convert data into a common format and standardize units, terminologies, and codes to ensure consistency.
- Data Integration: Integrate data from various sources into a unified format, considering data types, relationships, and hierarchies.
- Data Validation: Validate the transformed data to ensure that it meets the defined quality standards and business rules.

Loading:

- Staging: Store the cleaned and transformed data in a staging area, separate from the data warehouse, to facilitate further validation.
- Data Warehouse Loading: Load the validated and transformed data into the data warehouse.