3. **Design a complete set of SQL queries to satisfy transaction requirements identified in the previous stages, using the relational schema and views defined in tasks 2 and 3 above.**

- Select Queries - Done for Guest views:

Virtual Table 1 -

Viewing the cost and usage amount that uses every energy source by month

- SELECT DATE_PART('month', StartDate), SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate)
  GROUP BY DATE_PART('month', StartDate);

Viewing the cost and usage amount that uses more than 1 and less than the total amount of energy sources by month

- SELECT DATE_PART('month', StartDate), SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type in ('<Energy Source>', '<Energy Source>')
  GROUP BY DATE_PART('month', StartDate);

Viewing the cost and usage amount that uses 1 energy source by month

- SELECT DATE_PART('month', StartDate), SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type = <Energy Source>
  GROUP BY DATE_PART('month', StartDate);

Viewing the cost and usage amount that uses every energy source by year

- SELECT DATE_PART('year', StartDate), SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)

FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate)
GROUP BY DATE_PART('year', StartDate);

Viewing the cost and usage amount that uses more than 1 and less than the total amount of energy sources by year

- SELECT DATE_PART('year', StartDate), SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type in ('<Energy Source>', '<Energy Source>')
  GROUP BY DATE_PART('year', StartDate);

Viewing the cost and usage amount that uses 1 energy source by year

- SELECT DATE_PART('year', StartDate), SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type = <Energy Source>
  GROUP BY DATE_PART('year', StartDate);

Virtual Table 2-

Viewing the cost and usage amount that uses every energy source in a 15 minute time period

- SELECT DATE_PART('hour', DATE_INTERVAL), Cost, UsageAmount, UsageAmount/cast(Cost as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate)
  GROUP BY DATE_PART('hour', DATE_INTERVAL);

Viewing the cost and usage amount that uses more than 1 and less than the total amount of energy sources in a 15 minute time period

- SELECT DATE_PART('hour', DATE_INTERVAL), Cost, UsageAmount, UsageAmount/cast(Cost as float)

FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type in ('<Energy Source>', '<Energy Source>')
GROUP BY DATE_PART('hour', DATE_INTERVAL);

Viewing the cost and usage amount that uses 1 energy source in a 15 minute time period

- SELECT DATE_PART('hour', DATE_INTERVAL, Cost, UsageAmount, UsageAmount/cast(Cost as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type = <Energy Source>
  GROUP BY DATE_PART('hour', DATE_INTERVAL);

Virtual Table 3-

Viewing the cost amount that uses every energy source, with user being able to input year or month

- SELECT DATE_PART(<userDateType>, StartDate), Meter_Type, SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate)
  GROUP BY DATE_PART(<userDateType>, StartDate);

Viewing the cost amount that uses more than 1 and less than the total amount of energy sources with user being able to input year or month

- SELECT DATE_PART(<userDateType>, StartDate), Meter_Type, SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type in ('<Energy Source>', '<Energy Source>')
  GROUP BY DATE_PART(<userDateType>, StartDate);

Viewing the cost amount that uses 1 energy source with user being able to input year or month

- SELECT DATE_PART(<userDateType>, StartDate), Meter_Type, SUM(Cost), UsageAmount, UsageAmount/cast(SUM(Cost) as float)
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type = <Energy Source>
  GROUP BY DATE_PART(<userDateType>, StartDate);

Virtual Table 4 -

Viewing the cost amount that uses every energy source, with user being able to input year, specifying by month

- SELECT DATE_PART('month', StartDate), Meter_Type, SUM(UsageAmount),
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate)
  GROUP BY DATE_PART('month', StartDate);

Viewing the cost amount that uses more than 1 and less than the total amount of energy sources with user being able to input year, specifying by month

- SELECT DATE_PART('month', StartDate), Meter_Type, SUM(UsageAmount),
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE <userYear> >= Date_part('year', StartDate) AND <userYear> <= Date_part('year', EndDate) AND Meter_Type in ('<Energy Source>', '<Energy Source>')
  GROUP BY DATE_PART('month', StartDate);

Viewing the cost amount that uses 1 energy source, with user being able to input year, specifying by month

- SELECT DATE_PART('month', StartDate), Meter_Type, SUM(UsageAmount),
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE StartDate >= <Starting Date> AND 'year <= <Ending Date> AND Meter_Type = <Energy Source>
  GROUP BY DATE_PART('month', StartDate);

Virtual Table 5 -

Viewing the cost amount that uses every energy source, with user being able to input year, specifying by season

- SELECT DATE_PART(‘year’, StartDate), Meter_Type, SUM(UsageAmount),
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE StartDate >= <Starting Date> AND ‘year <= <Ending Date> AND Type_Of_Season = <SeasonInput>
  GROUP BY DATE_PART(year’, StartDate), Type_Of_Season;

Viewing the cost amount that uses more than 1 and less than the total amount of energy sources with user being able to input year, specifying by season

- SELECT DATE_PART(‘year’, StartDate), Meter_Type, SUM(UsageAmount),
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE StartDate >= <Starting Date> AND ‘year <= <Ending Date> AND Type_Of_Season = <SeasonInput> AND Meter_Type in (‘<Energy Source>’, ‘<Energy Source>’)
  GROUP BY DATE_PART(year’, StartDate), Type_Of_Season;

Viewing the cost amount that uses 1 energy source, with user being able to input year, specifying by season

- SELECT DATE_PART(‘year’, StartDate), Meter_Type, SUM(UsageAmount),
  FROM (ENERGY_SOURCE NATURAL JOIN MAPS_TO) NATURAL JOIN DATE_INTERVAL
  WHERE StartDate >= <Starting Date> AND ‘year <= <Ending Date> AND Type_Of_Season = <SeasonInput> AND Meter_Type = <Energy Source>
  GROUP BY DATE_PART(year’, StartDate), Type_Of_Season;

- ***Insert Queries - Meant for Admins***:

INSERT INTO DATE_INTERVAL
VALUES(<MeterConsumptionID>, <StartDate>, <EndDate>, <TypeOfSeason>, <Time>);

INSERT INTO BUILDING
VALUES(<Portfolio_Manager_ID>, <Name>, <Construction_Status>, <Gross_Floor_Area>, <YearBuilt>);

INSERT INTO BUILDING_TYPE
VALUES(<Name>, <Property_Type>);

INSERT INTO DATE_INTERVAL
VALUES (<MeterConsumptionID>, <StartDate>, <EndDate>, <TypeOfSeason>, <Time>);

INSERT INTO FUEL_OIL
VALUES (<PortfolioManagerMeterID>. <Units>, <Meter_Type>);

INSERT INTO NATURAL_GAS
VALUES (<PortfolioManagerMeterID>. <Units>, <Meter_Type>);

INSERT INTO ELECTRIC_GRID
VALUES (<PortfolioManagerMeterID>. <Units>, <Meter_Type>);

INSERT INTO OTHER_SOURCE
VALUES (<PortfolioManagerMeterID>. <Units>, <Meter_Type>);

INSERT INTO ENERGY_SOURCE
VALUES(<PortfolioManagerMeterID>, <MeterName>, <Meter_Type>);

INSERT INTO ENERGY_SOURCE_COST
VALUES(<PortfolioManagerMeterID>, <Cost>, <UsageAmount>);

- ***Update Queries - Meant for Admins***:

Below are examples of update queries that the admin could take for each table. Overall, the update queries would follow the format of UPDATE … SET…. WHERE;

UPDATE DATE_INTERVAL
SET StartDate = '09-10-2001', End Date = '10-10-2001'
WHERE MeterConsumptionID = 000045;

UPDATE BUILDING
SET Name = 'Eickhoff Food Hall'
WHERE Name = 'Eickhoff';

UPDATE BUILDING_TYPE
SET Property_Type = 'Dormitory'
WHERE Property_Type = 'ResidenceHall/Dormitory';

UPDATE FUEL_OIL
SET Units = 'gallons'
WHERE Units = 'cups';

```
UPDATE NATURAL_GAS
SET Units = 'cubic feet'
WHERE Units = 'tablespoons';

UPDATE ELECTRIC_GRID
SET Units = 'watts'
WHERE Units = 'kWh';

UPDATE OTHER_SOURCE
SET Units = 'kBtu'
WHERE Units = 'cups';

UPDATE ENERGY_SOURCE
SET Meter_Type = 'EL5'
WHERE Meter_Type = 'EL6';

UPDATE ENERGY_SOURCE_COST
SET Cost = 0
WHERE UsageAmount = 0;
```

- ***Delete Queries - Meant for Admins:***

```
DELETE FROM DATE_INTERVAL
WHERE MeterConsumptionID = <MeterConsumptionID>;

DELETE FROM BUILDING
WHERE Portfolio_Manager_ID = <Portfolio_Manager_ID>;

DELETE FROM FUEL_OIL
WHERE PortfolioManagerMeterID = <PortfolioManagerMeterID>;

DELETE FROM NATURAL_GAS
WHERE PortfolioManagerMeterID = <PortfolioManagerMeterID>;

DELETE FROM ELECTRIC_GRID
WHERE PortfolioManagerMeterID = <PortfolioManagerMeterID>;

DELETE FROM OTHER_SOURCE
WHERE PortfolioManagerMeterID = <PortfolioManagerMeterID>;
```

```
DELETE FROM ENERGY_SOURCE
WHERE PortfolioManagerMeterID = <PortfolioManagerMeterID>;
```