Project Report

1. **Define the Information Needs to Fetch:**
    - Clearly understand and define the specific information or data you need from the database. This could involve understanding the business requirements or user queries.
2. **Collect the Tables Associated with the Task:**
    - Identify the tables in the database that contain the relevant data. These tables should be related to the information you're trying to retrieve.
3. **Find the Relative Columns and Map Them:**
    - Determine which columns in the identified tables contain the required information. Map these columns to their corresponding attributes or fields in the task.
4. **Find the Connection Between Tables for Joining Operations:**
    - Identify the relationships between the tables. This involves finding the common columns that can be used to join the tables together. This is crucial for constructing a meaningful query.
5. **Consider the Conditions Given in the Task:**
    - Take into account any specific conditions or criteria mentioned in the task that should be applied to filter the data. This could involve WHERE clauses or other conditional statements.
6. **Formulate the SQL Query:**
    - Based on the above information, construct a SQL query that combines the relevant tables, selects the necessary columns, and applies any conditions specified. This may involve INNER JOIN, LEFT JOIN, WHERE clauses, etc.
7. **Test the Resulting Table:**
    - Before finalizing the query, run it on a test environment to ensure that the results match the expected outcome. This helps identify any issues or errors in the query.
8. **End:**
    - If the test is successful, you can consider the process complete. If not, review the query, identify and fix any issues, and retest until the desired results are achieved.

Here's a simple example SQL query structure:

```
SELECT
    t1.column1,
```

```
    t2.column2,
    ...
FROM
    table1 t1
JOIN
    table2 t2 ON t1.common_column = t2.common_column
WHERE
    t1.condition_column = 'some_value'
    AND t2.another_condition_column > 100;
```

Refer the Excel sheet for the approach:

Need to find Account Number, Customer Name, Customer Type, Who initiated the request, Request date, Request Type.

```
SELECT
    A.AccountNumber,
    A.CustomerName,
    A.CustomerType,
    R.Initiator AS WhoInitiatedRequest,
    R.RequestDate,
    R.RequestType
FROM
    Accounts A
JOIN
    Requests R ON A.AccountNumber = R.AccountNumber;
```

table 1	[CUSTFORM].[CustomerForm]
table 2	[CUSTREQUESTFLOW].[ApprovedCustomer_SystemID]
table 3	[ADM].[Employee]
table 4	[CUSTFORM].[CustomerType]

Joining approach

Inner Join

table 1	table 2
CDM_ID	CDM_ID

table 1	table 3
CustomerSubmittedEmail_ByEmployeeID	Employee_ID

table 1	table 4
CustomerType_ID	CustomerType_ID

```
SELECT
    CF.CDM_ID,
    CF.CustomerSubmittedEmail_ByEmployeeID,
    E.Employee_ID,
    CT.CustomerType_ID
FROM
    [CUSTFORM].[CustomerForm] CF
INNER JOIN
    [CUSTREQUESTFLOW].[ApprovedCustomer_SystemID] AC
ON
    CF.CDM_ID = AC.CDM_ID
INNER JOIN
    [ADM].[Employee] E
ON
    CF.CustomerSubmittedEmail_ByEmployeeID = E.Employee_ID
INNER JOIN
    [CUSTFORM].[CustomerType] CT
ON
    CF.CustomerType_ID = CT.CustomerType_ID;
```

Condition: Where CSI_System_Name = ORCL, Show CSI_System_ID

```
SELECT
```

```
    CF.CDM_ID,
    CF.CustomerSubmittedEmail_ByEmployeeID,
    E.Employee_ID,
    CT.CustomerType_ID,
    AC.CSI_System_ID
FROM
    [CUSTFORM].[CustomerForm] CF
INNER JOIN
    [CUSTREQUESTFLOW].[ApprovedCustomer_SystemID] AC
ON
    CF.CDM_ID = AC.CDM_ID
INNER JOIN
    [ADM].[Employee] E
ON
    CF.CustomerSubmittedEmail_ByEmployeeID = E.Employee_ID
INNER JOIN
    [CUSTFORM].[CustomerType] CT
ON
    CF.CustomerType_ID = CT.CustomerType_ID
WHERE
    AC.CSI_System_Name = 'ORCL';
```

Results:

- The report includes customer details along with the associated approved customer system information, focusing on records where the system name is 'ORCL'.