- List all books by a specific author: select * from Item where Author = "Brené Brown";
- Find books by publication year: select * from Item where Pub_Year = 2023;
- Check membership status: select * from Library_member where Account_status = "Active";
- Fine calculation: select * from Check_out_transaction where Fine_amount > 0;
- Book availability: select * from Copies_of_item where Availability = "Available";
- Frequent borrowers (members that have borrowed more than once):
  select Name from Library_member join Check_out_transaction on Check_out_transaction.Member_ID = Library_member.Member_ID group by Library_member.Member_ID, Name having count(Check_out_transaction.Member_ID) > 1;
- Books due soon (books that are still need to be return prior to due date):
  select Title, Copies_of_item.Item_ID from Item join Copies_of_item on Copies_of_item.ISBN = Item.ISBN join Check_out_transaction on Check_out_transaction.Item_ID = Copies_of_item.Item_ID where(Check_out_transaction.Fine_amount = 0.00 and Check_out_transaction.Return_date is NULL);
- Members with overdue books (members that have not returned a book and have a fine now):
  select Name from Library_member join Check_out_transaction on Check_out_transaction.Member_ID = Library_member.Member_ID where(Check_out_transaction.Fine_amount > 0.00 and Check_out_transaction.Return_date is NULL);
- Average borrowing time:
  select avg(DATEDIFF(Return_date, Check_out_date)) from Check_out_transaction where Return_date is not NULL;
- Most popular author in the last month(database built with the transaction just in a month):
  select Item.Author, count(*) as checkouts from Item join Copies_of_item ON Copies_of_item.ISBN = Item.ISBN JOIN Check_out_transaction ON Check_out_transaction.Item_ID = Copies_of_item.Item_ID group by Item.Author order by checkouts desc limit 1;
- Monthly fees report: SELECT DATE_FORMAT(Return_date, '%Y-%m') AS Report_Month, SUM(Fine_amount) AS Monthly_Total_Fines FROM Check_out_transaction WHERE Fine_amount > 0 AND Return_date IS NOT NULL GROUP BY Report_Month ORDER BY Report_Month;
- Exceeded borrowing limits: select * from Library_member where Account_status = "Suspended";
- Frequent borrowed items by client type:

  ```
  WITH BorrowCounts AS (
     SELECT lm.Membership_type, i.Item_Type, COUNT(*) AS Borrow_Count FROM
  Library_member lm JOIN Check_out_transaction cot ON lm.Member_ID = cot.Member_ID
  JOIN Copies_of_item coi ON cot.Item_ID = coi.Item_ID JOIN Item i ON coi.ISBN = i.ISBN
  GROUP BY lm.Membership_type, i.Item_Type
  ),
  MaxCounts AS (
     SELECT Membership_type, MAX(Borrow_Count) AS Max_Borrow_Count FROM
  BorrowCounts GROUP BY Membership_type
  )
  SELECT bc.Membership_type, bc.Item_Type, bc.Borrow_Count FROM BorrowCounts bc
  JOIN MaxCounts mc ON bc.Membership_type = mc.Membership_type
  ```

```
AND bc.Borrow_Count = mc.Max_Borrow_Count;
```

- Never late returns:
  ```
  select Name from Library_member join Check_out_transaction on
  Check_out_transaction.Member_ID = Library_member.Member_ID where
  DATEDIFF(Due_date, Return_date) > 0;
  ```
- Average loan duration:
  ```
  select avg(DATEDIFF(Return_date, Check_out_date)) from Check_out_transaction where
  Return_date is not NULL;
  ```
- Monthly summary report:
  ```
  SELECT DATE_FORMAT(Check_out_date, '%Y-%m') AS Report_Month,
  COUNT(Transaction_ID) AS Books_Checked_Out, SUM(Fine_amount) AS Monthly_Fines
  FROM Check_out_transaction GROUP BY Report_Month ORDER BY Report_Month;
  ```
- Statistics breakdown:
  ```
  Select
          (select count(Transaction_ID) from Check_out_transaction) as Total_Checkouts,
          (select sum (Fine_amount) from Check_out_transaction) as Total_Fines_Collected,
          (select count(Member_ID) from Library_member where Account_status = “Active” as
          Active_members;
  ```

- Client borrowing report :
  ```
  select Name, Item.Title, Check_out_transaction.Due_date,
  Check_out_transaction.Return_date from Library_member join Check_out_transaction on
  Check_out_transaction.Member_ID = Library_member.Member_ID join Copies_of_item on
  Copies_of_item.Item_ID = Check_out_transaction.Item_ID join Item on
  Copies_of_item.ISBN = Item.ISBN
  ```
- Item availability and history:
  ```
  select Title from Item join Copies_of_item on Copies_of_item.ISBN = Item.ISBN where
  Availability = "available";
  ```
- Overdue items report:
  ```
  select Title from Item join Copies_of_item on Copies_of_item.ISBN = Item.ISBN join
  Check_out_transaction on Check_out_transaction.Item_ID = Copies_of_item.Item_ID
  where(Check_out_transaction.Fine_amount > 0.00 and Check_out_transaction.Return_date
  is NULL);
  ```
- Revenue summary:
  ```
  SELECT COALESCE(SUM(Fine_amount), 0) AS Total_Fines_Last_Month FROM
  Check_out_transaction WHERE Return_date >= DATE_FORMAT(CURDATE() - INTERVAL
  1 MONTH, "%Y-%m-01") AND Return_date < DATE_FORMAT(CURDATE(), "%Y-%m-01");
  ```

Connection Guide:

- MySQL Client
  - If you have the MySQL client installed you can type the following command into your command prompt or terminal.
  - `mysql -h Bstoney.co -P 3306 -u externaluser -p`
  - MySQL will prompt you for the password which is "EECS465"

- Database Managers
    - Click add new connection and add this information in:
        - Domain name: Bstoney.co
        - Port: 3306
        - User: externaluser
        - Password: EECS465