NOTE: for the insert, delete, and update queries, we use pointers, for example, “$customerID”, to point to GUI user input so that they can be run directly in the xampp admin SQL box.

```
//Insert Operation
INSERT INTO customerCheckinOut VALUES
('$customerID','$deskNum','$dates','$paymentType','$customerName');

//Delete Operation
DELETE FROM customerCheckinOut WHERE customerID = '$customerID';

//Update Operation
UPDATE customerCheckinOut
SET
deskNum = '$deskNum',
dates = '$dates',
paymentType = '$paymentType',
customerName = '$customerName'
WHERE customerID = '$customerID';

//Selection
SELECT customerID, deskNum, dates, paymentType, customerName FROM
customerCheckinOut WHERE paymentType ='cash';
SELECT customerID, deskNum, dates, paymentType, customerName FROM
customerCheckinOut WHERE dates LIKE '%Aug%' or 'aug' or'AUG';

//Projection
SELECT customerID,paymentType FROM customerCheckinOut;
SELECT customerID,paymentType FROM customerCheckinOut;
SELECT customerID,dates FROM customerCheckinOut;

//Join Query
SELECT FrontDesk.Building,FrontDesk.deskNum, buildingFrontdesk.Floor
FROM FrontDesk
INNER JOIN buildingFrontdesk ON FrontDesk.Building = buildingFrontdesk.Building;

//Aggregation query
SELECT * FROM hotelRoom
where Price = (SELECT MAX(Price) FROM hotelRoom);

//Nested Aggregation with group-by
SELECT Building, COUNT(*) from FrontDesk, customerCheckinOut
WHERE FrontDesk.deskNum = customerCheckinOut.deskNum
GROUP BY Building
```

```
HAVING COUNT(*) >= ALL (SELECT COUNT(*) from FrontDesk, customerCheckinOut
WHERE FrontDesk.deskNum = customerCheckinOut.deskNum
GROUP BY Building);

//Division query
SELECT customerID, customerName from customerCheckinOut
WHERE customerID =
(SELECT DISTINCT customerID FROM visit as sx
WHERE NOT EXISTS (
(SELECT Name FROM amenities as p )
EXCEPT
(SELECT amenityName FROM visit as sp WHERE sp.customerID = sx.customerID ) ));
```