

Set 1

1. a. Write a query to find the names and ratings of sailor whose rating is better than some sailor called Horatio using ANY operator.

```
SELECT name, rating
FROM sailors
WHERE rating > ANY (
    SELECT rating
    FROM sailors
    WHERE name = 'Horatio'
);
```

-
- b. write a query to find the boat ID's of the boats other than those which are reserved by sailor with sid 31 using NOT IN operator.

```
SELECT boatID
FROM boats
WHERE boatID NOT IN (
    SELECT bid
    FROM reserves
    WHERE sid = 31
);
```

2. a. Write a query to find the sailor id's of sailors with age over 20 who have not reserved boat 103.

```
SELECT sid
FROM sailors
WHERE age > 20
AND sid NOT IN (
    SELECT sid
    FROM reserves
    WHERE bid = 103
);
```

);

b. Write a query to display the all sailors info and who have reserved any boat. (use left outer join)

```
SELECT s.sid, s.name, s.age, r.bid
FROM sailors s
LEFT OUTER JOIN reserves r ON s.sid = r.sid;
```

3. a. Write a query to find the sailors whose sid is 22,31,64,or 74 and name starts with H using intersect.

```
SELECT sid, name
FROM sailors
WHERE sid = 22 AND name LIKE 'H%'
INTERSECT
SELECT sid, name
FROM sailors
WHERE sid = 31 AND name LIKE 'H%'
INTERSECT
SELECT sid, name
FROM sailors
WHERE sid = 64 AND name LIKE 'H%'
INTERSECT
SELECT sid, name
FROM sailors
WHERE sid = 74 AND name LIKE 'H%';
```

b. Write a query to find the names of sailors whose age is greater than the sailors who have reserved boat 101 or 102.

```
SELECT name
FROM sailors
WHERE age > (
    SELECT MAX(age)
    FROM sailors
    WHERE sid IN (
        SELECT sid
        FROM reserves
        WHERE bid IN (101, 102)
    )
);
```

4. Create a view std_courses_view which shows the students sid, name and the course id's registered.
5. Create a trigger to validate the age at the time before inserting a new record into Sailors table and if invalid age(age<0 or age >100) found then it stores 0 instead.
6. Create a cursor which updates a table by increasing the salary of each employee by 1500. After the update, the SQL%ROWCOUNT attribute is used to find out how many rows were affected by the operation.
7. Explain syntaxes and example queries of DDL and DML commands.
8. Draw ER Diagram for the following specifications: The ATM system is specialized software that aids in managing a bank account or holder's funds simply. Users can check account balances, make cash withdrawals or deposits, print a record of account activities or transactions. Customers can utilize the ATM system to conduct self-service transactions such as deposits, cash withdrawals, bill payments, and account transfers

Set2

1. a. Write a query to find the names of sailors who have reserved either boat 103 or 104 using IN operator.

```
SELECT name
FROM sailors
WHERE sid IN (
    SELECT sid
    FROM reserves
    WHERE bid IN (103, 104)
);
```

-
- b. Write a query to find the sailors whose sid is 22,32,64,or 85 and name starts with A using intersect.

```
SELECT sid, name
FROM sailors
WHERE sid = 22 AND name LIKE 'A%'
INTERSECT
SELECT sid, name
FROM sailors
WHERE sid = 32 AND name LIKE 'A%'
INTERSECT
SELECT sid, name
FROM sailors
WHERE sid = 64 AND name LIKE 'A%'
INTERSECT
SELECT sid, name
FROM sailors
WHERE sid = 85 AND name LIKE 'A%';
```

-
-
2. a. Write a query to find the names of sailors who have reserved all Interlake boats.

```
SELECT name
FROM sailors s
WHERE NOT EXISTS (
    SELECT bid
    FROM boats b
    WHERE b.bname LIKE 'Interlake%'
);
```

```

AND NOT EXISTS (
  SELECT 1
  FROM reserves r
  WHERE r.sid = s.sid
  AND r.bid = b.bid
)
);

```

b. Write a query to find the ids and names of sailors who have reserved any boat using simple join operation.

```

SELECT s.sid, s.name
FROM sailors s
JOIN reserves r ON s.sid = r.sid;

```

3. a. Write a query to find the ids, names of sailors who have reserved a red boat or a green boat using Union operator.

```

SELECT s.sid, s.name
FROM sailors s
JOIN reserves r ON s.sid = r.sid
JOIN boats b ON r.bid = b.bid
WHERE b.color = 'red'

```

UNION

```

SELECT s.sid, s.name
FROM sailors s
JOIN reserves r ON s.sid = r.sid
JOIN boats b ON r.bid = b.bid
WHERE b.color = 'green';

```

b. Write a query to find the names of the students whose percentage is greater than the students who registered for java or C.

```

SELECT s.name

```

```

FROM students s
WHERE s.percentage > (
    SELECT MAX(percentage)
    FROM students
    WHERE name IN (
        SELECT name
        FROM registrations
        WHERE course IN ('Java', 'C')
    )
);

```

4. Create a view student_view which shows the students sid whose percentage is greater than 60.
5. Create a procedure to display the n number of records from students table.
6. Write a PL-SQL Program to create a trigger to backup a row which is delete from the one table and inserted in the another table.
7. What are aggregate functions, explain each of them with an example query of your own?
8. Identify Entities and attributes for the Mobile Billing System which deals with maintenance of Customer details, billing details, Plan details, cities of service details and draw E-R diagram.

Set3

1. a. Write a query to find the name and the age of the youngest sailor

```

SELECT name, age
FROM sailors
ORDER BY age
LIMIT 1;

```

- b. Find the names of sailors who have reserved either of boats 101,103,104 using IN operator.

```

SELECT name
FROM sailors
WHERE sid IN (
    SELECT sid

```

```
FROM reserves
WHERE bid IN (101, 103, 104)
);
```

2. a. Find the names of sailors who have reserved both a red and a green boat using INTERSECT operator.

```
SELECT name
FROM sailors
WHERE sid IN (
    SELECT sid
    FROM reserves
    WHERE bid IN (
        SELECT bid
        FROM boats
        WHERE color = 'red'
    )
)

INTERSECT

SELECT sid
FROM reserves
WHERE bid IN (
    SELECT bid
    FROM boats
    WHERE color = 'green'
)
);
```

- b. Write a query to find sid of sailors whose age is greater than 20 or who have reserved boat 103 using union operator.

```
SELECT sid
FROM sailors
WHERE age > 20

UNION
```

```
SELECT sid
FROM reserves
WHERE bid = 103;
```

3. a. Write a query to find the names of the students who secured O grade and registered atleast one course.

```
SELECT s.name
FROM students s
JOIN grades g ON s.student_id = g.student_id
WHERE g.grade = 'O'
AND s.student_id IN (
    SELECT student_id
    FROM registrations
);
```

- b. Write a query to find names of students who secured percentage greater than the students who registered for java.

```
SELECT s.name
FROM students s
WHERE s.percentage > (
    SELECT MAX(percentage)
    FROM students
    WHERE name IN (
        SELECT name
        FROM registrations
        WHERE course = 'Java'
    )
);
```

4. Create a view `Sailors_reservation_view` which shows the Sailors name who reserved Interlake.

5. Create a trigger to store the sid of a student into the table deleted_students before deleting any record from students table.
6. Create a cursor which updates a table by increasing the salary of each employee by 1500. After the update, the SQL%ROWCOUNT attribute is used to find out how many rows were affected by the operation.
7. Explain syntaxes and example queries of DCL and TCL commands.
8. Identify Entities and attributes for University Management System in which details of institutions, faculty details, course details, branch details, college details are maintained and draw E-R diagram.

Set4

1. a. Write a query to find the name and the age of the eldest sailor.

```
SELECT name, age
FROM sailors
ORDER BY age DESC
LIMIT 1;
```

- b. Write a query to find the names of Sailors who have reserved the boats reserved by Luber using IN operator

```
SELECT name
FROM sailors
WHERE sid IN (
    SELECT sid
    FROM reserves
    WHERE bid IN (
        SELECT bid
        FROM reserves
        WHERE sid = (
            SELECT sid
            FROM sailors
            WHERE name = 'Luber'
        )
    )
);
```

2. a. Write a query to find the names of sailors who have reserved Clipper and Marine boats using INTERSECT operator.

```
SELECT name
FROM sailors
WHERE sid IN (
    SELECT sid
    FROM reserves
    WHERE bid IN (
        SELECT bid
        FROM boats
        WHERE bname = 'Clipper'
    )
)

INTERSECT

SELECT sid
FROM reserves
WHERE bid IN (
    SELECT bid
    FROM boats
    WHERE bname = 'Marine'
)
);
```

- b. Write a query to display all sid's who have reservations and all boats info. (use right outer join)

```
SELECT r.sid, b.*
FROM reserves r
RIGHT OUTER JOIN boats b ON r.bid = b.bid;
```

3. a. Write a query to find the sids of sailors who have reserved a red boat or a green boat.

```
SELECT DISTINCT r.sid
FROM reserves r
```

```
JOIN boats b ON r.bid = b.bid
```

```
WHERE b.color = 'red' OR b.color = 'green';
```

b. Write a query to find the names of sailors whose age is greater than the maximum age of the sailors who reserved boat 101 or 102.

```
SELECT name
```

```
FROM sailors
```

```
WHERE age > (
```

```
    SELECT MAX(age)
```

```
    FROM sailors
```

```
    WHERE sid IN (
```

```
        SELECT sid
```

```
        FROM reserves
```

```
        WHERE bid IN (101, 102)
```

```
    )
```

```
);
```

4. Create a view student_view which shows students with distinction.(i.e., grade O)

```
CREATE VIEW student_view AS
```

```
SELECT name
```

```
FROM students
```

```
WHERE grade = 'O';
```

5. Create a procedure that prints the max percentage obtained by a student.
6. Write a PL-SQL Program to Print the salary changes when the salary is changed.
7. Write a query to add a new column: Saddress to a Sailors table using alter command.
8. A hospital with a set of patients and a set of medical doctors. Associate with each patient a log of the various tests and examinations conducted. Identify the tables, attributes and their relationships between them, represent them using a E-R diagram.

Set5

1. Find the ids and names of sailors who have reserved two different boats on the same day.

```
SELECT s.sid, s.name
FROM sailors s
JOIN reserves r1 ON s.sid = r1.sid
JOIN reserves r2 ON s.sid = r2.sid AND r1.bid <> r2.bid
WHERE r1.res_date = r2.res_date;
```

2. a. Write a query to find the names of sailors who have reserved Clipper and Marine boats using INTERSECT operator.

```
SELECT name
FROM sailors
WHERE sid IN (
    SELECT sid
    FROM reserves
    WHERE bid IN (
        SELECT bid
        FROM boats
        WHERE bname = 'Clipper'
    )
)

INTERSECT

SELECT sid
FROM reserves
WHERE bid IN (
    SELECT bid
    FROM boats
    WHERE bname = 'Marine'
)
);
```

b. Write a query to find sid of sailors whose age is greater than 20 or who have reserved boat 103 using union operator

```
SELECT sid
FROM sailors
WHERE age > 20
```

UNION

```
SELECT sid
FROM reserves
WHERE bid = 103;
```

3. a. Write a query to find the sids of sailors who have reserved a red boat or a green boat

```
SELECT DISTINCT r.sid
FROM reserves r
JOIN boats b ON r.bid = b.bid
WHERE b.color IN ('red', 'green');
```

b. Find the sids of sailors with age over 20 who have not reserved a red boat

```
SELECT s.sid
FROM sailors s
LEFT JOIN reserves r ON s.sid = r.sid
LEFT JOIN boats b ON r.bid = b.bid AND b.color = 'red'
WHERE s.age > 20 AND b.bid IS NULL;
```

4. Create a view student_view which shows students with distinction. (i.e., grade O).

5. Create a procedure that prints the greatest of 3 numbers.

6. Write a trigger to do the following:

if the ticket is booked in advance of more than 60 days, reject it.

I.e, date of journey must not greater than 60 days from reservation date

7. Write a Query to display grade and percentage of students whose percentage is greater than the average percentage of all students.

8.

Consider a university database for the scheduling of classrooms for final exams. This database could be modeled as the single entity set *exam*, with attributes *course-name*, *section-number*, *room-number*, and *time*. Alternatively, one or more additional entity sets could be defined, along with relationship sets to replace some of the attributes of the *exam* entity set, as

a. course with attributes *name*, *department*, and *dc-number*

b. section with attributes *section-number* and *enrollment*, and dependent as a weak entity set on *course*

c. room with attributes *sr-number*, *capacity*, and *building*

Show an E-R diagram illustrating the use of all three additional entity sets listed.

Set6

1. a. Find the names of sailors who have reserved both a red and a green boat using INTERSECT operator.

```
SELECT name
FROM sailors
WHERE sid IN (
    SELECT sid
    FROM reserves
    WHERE bid IN (
        SELECT bid
        FROM boats
        WHERE color = 'red'
    )
)
```

INTERSECT

```
SELECT sid
FROM reserves
WHERE bid IN (
    SELECT bid
```

```
        FROM boats
        WHERE color = 'green'
    )
);
```

b. Write a query to display the all sailors info and who have reserved any boat. (use left outer join)

```
SELECT s.sid, s.name, s.age, r.bid
FROM sailors s
LEFT OUTER JOIN reserves r ON s.sid = r.sid;
```

2. a. Write a query to find the ids, names of sailors who have reserved a red boat or a green boat using Union operator.

```
SELECT s.sid, s.name
FROM sailors s
JOIN reserves r ON s.sid = r.sid
JOIN boats b ON r.bid = b.bid
WHERE b.color = 'red'
```

UNION

```
SELECT s.sid, s.name
FROM sailors s
JOIN reserves r ON s.sid = r.sid
JOIN boats b ON r.bid = b.bid
WHERE b.color = 'green';
```

b. Write a query to find names of students who secured percentage greater than the students who registered for java

```
SELECT s.name
FROM students s
```

```

WHERE s.percentage > (
    SELECT MAX(percentage)
    FROM students
    WHERE name IN (
        SELECT name
        FROM registrations
        WHERE course = 'Java'
    )
);

```

3. Create a view `Sailors_reservation_view` which shows the Sailors name who reserved Interlake
4. Create a trigger to validate the age at the time before inserting a new record into Sailors table and if invalid age (age < 0 or age > 100) found then it stores 0 instead.
5. Create a cursor which updates a table by increasing the salary of each employee by 1500. After the update, the `SQL%ROWCOUNT` attribute is used to find out how many rows were affected by the operation.
6. Write a query to find the names of sailors who are elder than Dustin, Lubber and Rusty
7. Write a query to add a new column: Grade to a Students table using alter command and set grade according to percentage obtained
8. A university registrar's office maintains data about the following entities: (a) courses, including number, title, credits, syllabus, and prerequisites; (b) course offerings, including course number, year, semester, section number, instructor(s), timings, and classroom; (c) students, including student-id, name, and program; and (d) instructors, including identification number, name, department, and title. Further, the enrollment of students in courses and grades awarded to students in each course they are enrolled for must be appropriately modeled. Draw E-R model.

Set7

1. a. Find the names of sailors who have reserved either of boats 101,103,104 using IN operator.
b. Write a query to find the name and the age of the eldest sailor.
2. a. Write a query to find the names of sailors who have reserved all Interlake boats
b. Write a query to find the ids and names of sailors who have reserved any boat using simple join operation
3. a. Write a query to find the names of the students who secured O grade and registered atleast one course.
b. Write a query to find the names of sailors whose age is greater than the sailors who have reserved boat 101 or 102
4. Create a view student_view which shows the students sid whose percentage is greater than 60
5. Create a procedure to display the n number of records from students table.
6. CREATE OR REPLACE TRIGGER trig1 before insert on Passenger for each row to avoid duplicate insertion.
7. Given an example by using any of the table given above for Order By and Group By clauses.
8. Identify Entities and Attributes for the given specification: A manufacturing company produces products. The following product information is stored: product name, product ID and quantity on hand. These products are made up of many components. Each component can be supplied by one or more suppliers. The following component information is kept: component ID, name, description, suppliers who supply them, and products in which they are used

Set8

1. a. Write a query to find the names of Sailors who have reserved the boats reserved by Luber using IN operator.
SELECT DISTINCT s.sailor_name
FROM Sailors s
WHERE s.sailor_id IN (
SELECT r.sailor_id
FROM Reservations r
JOIN Sailors s2 ON r.sailor_id = s2.sailor_id
WHERE s2.sailor_name = 'Luber'
);

b. Write a query to find the sailors whose sid is 22,32,64,or 85 and name starts with A using intersect

```
SELECT sailor_id, sailor_name
FROM Sailors
WHERE sailor_id IN (22, 32, 64, 85)
INTERSECT
SELECT sailor_id, sailor_name
FROM Sailors
WHERE sailor_name LIKE 'A%';
```

2. a. Write a query to find the sailor id's of sailors with age over 20 who have not reserved boat 103.

```
SELECT sailor_id
FROM Sailors
WHERE age > 20
AND sailor_id NOT IN (
  SELECT sailor_id
  FROM Reservations
  WHERE boat_id = 103
);
```

B Write a query to print names of the students whose percentage is within 60 and 75.

```
SELECT student_name
FROM Students
WHERE percentage >= 60 AND percentage <= 75;
```

3. a. Write a query to find the names of the students who secured O grade and registered atleast one course

b. Write a query to find the names of sailors whose age is greater than the maximum age of the sailors who reserved boat 101 or 102

```
SELECT s.sailor_name
FROM Sailors s
WHERE s.age > (
    SELECT MAX(s2.age)
    FROM Sailors s2
    JOIN Reservations r ON s2.sailor_id = r.sailor_id
    WHERE r.boat_id IN (101, 102)
);
```

4. Create a view std_courses_view which shows the students sid, name and the course id's registered
5. Create a trigger to store the sid of a student into the table deleted_students before deleting any record from students table.
6. Create a cursor which updates a table by increasing the salary of each employee by 1500. After the update, the SQL%ROWCOUNT attribute is used to find out how many rows were affected by the operation.
7. Explain DML commands with example for each by using above tables?
8. Draw ER diagram for the Online Shopping System where the order details, billing details, customer details, product details, cities of service details are maintained