

```

CREATE DATABASE HRC3;
USE HRC3;

CREATE TABLE STUDENT (
    RollNo INT PRIMARY KEY,
    StudentName VARCHAR(100) NOT NULL,
    Age INT
);

CREATE TABLE COURSE (
    CID INT PRIMARY KEY,
    CourseName VARCHAR(100) NOT NULL,
    CourseType VARCHAR(20) NOT NULL,
    Course_Incharge VARCHAR(100) NOT NULL,
    TotalSeats INT DEFAULT 50
);

CREATE TABLE ADMISSION (
    SID INT PRIMARY KEY AUTO_INCREMENT,
    RollNo INT,
    CID INT,
    AdmissionDate DATE,
    EnrollmentFeesPaid VARCHAR(3) DEFAULT 'No',
    FOREIGN KEY (RollNo) REFERENCES STUDENT(RollNo),
    FOREIGN KEY (CID) REFERENCES COURSE(CID)
);

INSERT INTO STUDENT (RollNo, StudentName, Age) VALUES
(101, 'Anil Kumar', 20),
(102, 'Asha Singh', 21),
(103, 'Rohit Gupta', 22),
(104, 'Riya Jain', 19),
(105, 'Zara Khan', 20),
(106, 'Xavier Roy', 23),
(107, 'Arjun Pandit', 21),
(108, 'Nisha Patel', 22),
(109, 'Abhay Mishra', 20);

INSERT INTO COURSE (CID, CourseName, CourseType, Course_Incharge, TotalSeats) VALUES
(1, 'Computer Science', 'Fulltime', 'Dr. R. Gupta', 60),
(2, 'Chemistry', 'Parttime', 'Dr. Anjali Verma', 50),
(3, 'Mathematics', 'Fulltime', 'Prof. S. Gupta', 55),
(4, 'Physics', 'Parttime', 'Dr. Meera Nair', 40),
(5, 'BSc(P)CS', 'Fulltime', 'Dr. Vikas Sharma', 45);

INSERT INTO ADMISSION (RollNo, CID, AdmissionDate, EnrollmentFeesPaid) VALUES
(101, 1, '2024-07-20', 'Yes'),
(102, 2, '2024-07-21', 'No'),
(103, 3, '2024-07-20', 'Yes'),
(104, 4, '2024-07-19', 'Yes'),
(105, 2, '2024-07-21', 'No'),
(106, 5, '2024-07-22', 'Yes'),
(107, 1, '2024-07-21', 'No'),
(108, 3, '2024-07-22', 'Yes'),
(109, 1, '2024-07-23', 'Yes'),
(109, 2, '2024-07-23', 'No');

-- 1. Retrieve names of students enrolled in any course
SELECT DISTINCT s.StudentName
FROM STUDENT s
JOIN ADMISSION a ON s.RollNo = a.RollNo;

-- 2. Retrieve names of students enrolled in at least one part-time course
SELECT DISTINCT s.StudentName
FROM STUDENT s
JOIN ADMISSION a ON s.RollNo = a.RollNo
JOIN COURSE c ON a.CID = c.CID
WHERE c.CourseType = 'Parttime';

-- 3. Retrieve students' names starting with letter 'A'
SELECT StudentName FROM STUDENT WHERE StudentName LIKE 'A%';

-- 4. Retrieve details of students studying in 'Computer Science' or 'Chemistry'
SELECT s.*
FROM STUDENT s
JOIN ADMISSION a ON s.RollNo = a.RollNo
JOIN COURSE c ON a.CID = c.CID

```

```

WHERE c.CourseName IN ('Computer Science', 'Chemistry');

-- 5. Retrieve students whose names start with 'X' or 'Z' and RollNo ends with '9'
SELECT StudentName
FROM STUDENT
WHERE (StudentName LIKE 'X%' OR StudentName LIKE 'Z%')
AND CAST(RollNo AS CHAR) LIKE '%9';

-- 6. Find course details with more than 2 students enrolled
SELECT c.*, COUNT(a.RollNo) AS StudentCount
FROM COURSE c
JOIN ADMISSION a ON c.CID = a.CID
GROUP BY c.CID
HAVING COUNT(a.RollNo) > 2;

-- 7. Update student name
UPDATE STUDENT SET StudentName = 'Ayush' WHERE RollNo = 101;

-- 8. Find course names with more than 5 students enrolled
SELECT c.CourseName
FROM COURSE c
JOIN ADMISSION a ON c.CID = a.CID
GROUP BY c.CID
HAVING COUNT(a.RollNo) > 5;

-- 9. Youngest student in 'BSc(P)CS'
SELECT s.StudentName
FROM STUDENT s
JOIN ADMISSION a ON s.RollNo = a.RollNo
JOIN COURSE c ON a.CID = c.CID
WHERE c.CourseName = 'BSc(P)CS'
ORDER BY s.Age ASC
LIMIT 1;

-- 10. Most popular course (maximum enrolled)
SELECT c.CourseName
FROM COURSE c
JOIN ADMISSION a ON c.CID = a.CID
GROUP BY c.CID
ORDER BY COUNT(a.RollNo) DESC
LIMIT 1;

-- 11. Two most popular part-time courses
SELECT c.CourseName
FROM COURSE c
JOIN ADMISSION a ON c.CID = a.CID
WHERE c.CourseType = 'Parttime'
GROUP BY c.CID
ORDER BY COUNT(a.RollNo) DESC
LIMIT 2;

-- 12. Students admitted to full-time courses only
SELECT DISTINCT s.StudentName
FROM STUDENT s
WHERE s.RollNo NOT IN (
    SELECT a.RollNo FROM ADMISSION a
    JOIN COURSE c ON a.CID = c.CID
    WHERE c.CourseType = 'Parttime'
)
AND s.RollNo IN (
    SELECT a.RollNo FROM ADMISSION a
    JOIN COURSE c ON a.CID = c.CID
    WHERE c.CourseType = 'Fulltime'
);

-- 13. Courses with >30 students enrolled
SELECT c.CourseName
FROM COURSE c
JOIN ADMISSION a ON c.CID = a.CID
GROUP BY c.CID
HAVING COUNT(a.RollNo) > 30;

-- 14. Students + courses with at least one enrollment
SELECT DISTINCT s.StudentName, c.CourseName
FROM STUDENT s
JOIN ADMISSION a ON s.RollNo = a.RollNo
JOIN COURSE c ON a.CID = c.CID;

```

```

-- 15. Course names where faculty incharge has 'Gupta'
SELECT CourseName
FROM COURSE
WHERE Course_Incharge LIKE '%Gupta%';

-- 16. Course names where enrolled students = 10% of total seats
SELECT c.CourseName, COUNT(a.RollNo) AS EnrolledStudents, c.TotalSeats
FROM COURSE c
LEFT JOIN ADMISSION a ON c.CID = a.CID
GROUP BY c.CID
HAVING COUNT(a.RollNo) = ROUND(0.1 * c.TotalSeats);

-- 17. Vacant seats for each course
SELECT c.CourseName, (c.TotalSeats - COUNT(a.RollNo)) AS VacantSeats
FROM COURSE c
LEFT JOIN ADMISSION a ON c.CID = a.CID
GROUP BY c.CID;

-- 18. Increment total seats by 10%
UPDATE COURSE
SET TotalSeats = ROUND(TotalSeats * 1.10);

-- 19. (Already added EnrollmentFeesPaid field)
SELECT * FROM ADMISSION;

-- 20. Update admission date by 1 year
UPDATE ADMISSION
SET AdmissionDate = DATE_ADD(AdmissionDate, INTERVAL 1 YEAR);

-- 21. Create view: course names with total enrolled students
CREATE OR REPLACE VIEW Course_Enrollment_View AS
SELECT c.CourseName, COUNT(a.RollNo) AS TotalEnrolled
FROM COURSE c
LEFT JOIN ADMISSION a ON c.CID = a.CID
GROUP BY c.CID;
SELECT * FROM Course_Enrollment_View;

-- 22. Count number of courses with more than 5 students enrolled for each course type
SELECT c.CourseType, COUNT(*) AS CoursesWithMoreThan2Students
FROM COURSE c
JOIN ADMISSION a ON c.CID = a.CID
GROUP BY c.CourseType, c.CID
HAVING COUNT(a.RollNo) > 2;

-- 23. Add column Mobile number in student table with default value '9999999999'
ALTER TABLE STUDENT ADD COLUMN MobileNumber VARCHAR(15) DEFAULT '9999999999';
SELECT RollNo, StudentName, MobileNumber FROM STUDENT;

-- 24. Find total number of students whose age > 18
SELECT COUNT(*) AS StudentsAbove18
FROM STUDENT
WHERE Age > 18;

-- 25. Find names of students born in 2001 and admitted to at least one part-time course
SELECT DISTINCT s.StudentName
FROM STUDENT s
JOIN ADMISSION a ON s.RollNo = a.RollNo
JOIN COURSE c ON a.CID = c.CID
WHERE (2020 - s.Age) = 2001
AND c.CourseType = 'Parttime';

```