|  |
| --- |
| **• Lab 1: Create a new database named school\_db and a table called students with the following columns: student\_id, student\_name, age, class, and address.**  Ans:  CREATE DATABASE student\_db;  CREATE TABLE student(  student\_id int,  student\_name text,  age int,  class text,  address text  ); |
| **• Lab 2: Insert five records into the students table and retrieve all records using the SELECT statement.**  Ans:  INSERT INTO student VALUES(1,'ram',18,' maths','ahmedabad'),(2,'syam',19,' chemisty','surat'),(3,'sita',21,' pysics','vadodra'),(4,'lakshman',20,' biology','ahmedabad'),(5,'hanuman',18,'maths','surat'); |
| **• Lab 1: Write SQL queries to retrieve specific columns (student\_name and age) from the students table.**  Ans:  SELECT student\_name,age FROM student; |
| **• Lab 2: Write SQL queries to retrieve all students whose age is greater than 10**  Ans:  SELECT age FROM student WHERE age>10; |
| **• Lab 1: Create a table teachers with the following columns: teacher\_id (Primary Key), teacher\_name (NOT NULL), subject (NOT NULL), and email (UNIQUE).**  Ans:  CREATE TABLE teacher(  t\_id int PRIMARY KEY,  t\_name text NOT null,  subject text NOT null,  email text UNIQUE  );  INSERT INTO teacher VALUES(1,'rakeshsir','java','rakesh@gmail.com'),(2,'rmeshsir','python','rmesh@gmail.com'),(3,'bhikhasir','c++','bhikha@gmail.com'),(4,'shilpamem','javascript','shilpa@gmail.com'),(5,'shilymem','php','shily@gmail.com'); |

|  |
| --- |
| **• Lab 2: Implement a FOREIGN KEY constraint to relate the teacher\_id from the teachers table with the students table.**  Ans:  CREATE TABLE student1(  student\_id int PRIMARY KEY,  student\_name text,  t\_id int,  FOREIGN KEY (t\_id) REFERENCES teacher(t\_id)); |
| **• Lab 1: Create a table courses with columns: course\_id, course\_name, and course\_credits. Set the course\_id as the primary key.**  Ans:  CREATE TABLE course(  course\_id int PRIMARY KEY,  course\_name text,  course\_credit INT); |
| **• Lab 2: Use the CREATE command to create a database university\_db.**  Ans:  CREATE DATABASE university\_db; |
| **• Lab 1: Modify the courses table by adding a column course\_duration using the ALTER command.**  Ans:  [ALTER](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/alter-table.html) [TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/alter-table.html) course ADD course\_duration int; |
| **• Lab 2: Drop the course\_credits column from the courses table.**  Ans:  [ALTER](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/alter-table.html) [TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/alter-table.html) course DROP COLUMN course\_credit; |
| **• Lab 1: Drop the teachers table from the school\_db database**  Ans:  [DROP](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/drop-table.html) [TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/drop-table.html) teacher; |
| **• Lab 2: Drop the students table from the school\_db database and verify that the table has been removed**.  Ans:  DROP TABLE student1;  [DROP](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/drop-database.html) [DATABASE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/drop-database.html) student\_db; |

|  |
| --- |
| **• Lab 1: Insert three records into the courses table using the INSERT command.**  Ans:  INSERT INTO course VALUES('java',60000),('python',70000),('react',65000); |
| **• Lab 2: Update the course duration of a specific course using the UPDATE command.**  Ans:  [UPDATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/update.html) course [SET](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/set.html) c\_duration=8 WHERE c\_name='java';  UPDATE course SET c\_duration=9 WHERE c\_name='python';  UPDATE course SET c\_duration=10 WHERE c\_name='react'; |
| **• Lab 3: Delete a course with a specific course\_id from the courses table using the DELETE command.**  Ans:  DELETE FROM course WHERE c\_name='java';  DELETE FROM course WHERE c\_name='python';  DELETE FROM course WHERE c\_name='react'; |
| **• Lab 1: Retrieve all courses from the courses table using the SELECT statement.**  Ans:  [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \* FROM course1; |
| **• Lab 2: Sort the courses based on course\_duration in descending order using ORDER BY.**  Ans:  SELECT \* FROM course1 ORDER BY c\_duration DESC; |
| **• Lab 3: Limit the results of the SELECT query to show only the top two courses using LIMIT.**  Ans:  [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \* FROM course1 c\_cost LIMIT 2; |

|  |
| --- |
| • Lab 1: Create two tables: departments and employees. Perform an INNER JOIN to display employees along with their respective departments.  Ans:  SELECT departmenr.d\_name,emp.e\_name,departmenr.d\_id FROM departmenr INNER JOIN emp ON departmenr.d\_id=emp.d\_id; |
| • Lab 2: Use a LEFT JOIN to show all departments, even those without employees.  Ans:  [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) departmenr.d\_name,emp.e\_name,departmenr.d\_id FROM departmenr [LEFT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-functions.html%23function_left) JOIN emp ON departmenr.d\_id=emp.d\_id; |
| • Lab 1: Group employees by department and count the number of employees in each department using GROUP BY.  Ans:  [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) d\_name,[COUNT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_count)(name) FROM emp1 GROUP BY d\_name; |
| • Lab 2: Use the AVG aggregate function to find the average salary of employees in each department.  Ans:  [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) [AVG](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_avg)(salary) AS avgsalary FROM emp1; |
| • Lab 1: Write a stored procedure to retrieve all employees from the employees table based on department.  Ans:  [CREATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-procedure.html) [PROCEDURE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-procedure.html) ins1(id int,name text,d\_name text)  BEGIN  [INSERT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/insert.html) INTO emp [VALUES](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/miscellaneous-functions.html%23function_values)(id,name,d\_name);  End  CALL ins1(1,'meru',’it');  CALL ins1(2,'rohan','it');  CALL ins1(3,'kishan','salse'); |
| • Lab 2: Write a stored procedure that accepts course\_id as input and returns the course details.  Ans:  Delimeter $$  Create procedure t\_13( in I int , out course\_d text)  Begin  Select coursedetail into course\_d from course\_1 where i=course\_id;  End  Call t\_13(1 ,@coursedetail);  Select @coursedetail; |
| • Lab 1: Create a view to show all employees along with their department names.  Ans:  CREATE VIEW v1 AS SELECT name,d\_name FROM emp1;  SELECT \* FROM v1; |
| • Lab 2: Modify the view to exclude employees whose salaries are below $50,000.  Ans:  CREATE VIEW v12 AS SELECT \* FROM emp1 WHERE salary>=50000;  SELECT \* FROM v12; |

|  |
| --- |
| • Lab 1: Create a trigger to automatically log changes to the employees table when a new employee is added.  Ans:  CREATE TRIGGER t1 AFTER insert ON em FOR EACH ROW BEGIN INSERT INTO emdata(id,name,salary,pro) VALUES(new.id,new.name,new.salary,'insert record'); END  INSERT INTO em VALUES(1,'ram',30000),(2,'shyam',35000);  [INSERT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/insert.html) INTO em [VALUES](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/miscellaneous-functions.html%23function_values)(3,'ganesh',50000); |
| • Lab 2: Create a trigger to update the last\_modified timestamp whenever an employee record is updated.  Ans:    CREATE TRIGGER t2 AFTER UPDATE ON em FOR EACH ROW BEGIN INSERT INTO emdata(id,name,salary,pro) VALUES (new.id,new.name,new.salary,'update record'); END;  UPDATE em SET name='suresh' WHERE id=3;  UPDATE em SET salary=65000 WHERE id=3; |