|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| item | | Tasks | | Min pts | Max pts |
| 1 | | ER Diagram  At least 7 tables **all linked**  Each table must be linked in the ERD and within the code | |  | 2 |
| 2. | | At least 2 of the tables must have a minimum of 15 records  A date datatype must be in one of the tables  In one of your numeric columns make there are some nulls | |  | 2 |
| 3. | | QUESTION: What database engine are you using and why is it important INNODB first it is the default engine for the current MYSQL version, but most importantly, it supports foreign keys which are critical to the referential integrity of data. | |  | 2 |
| 4. | | Joins   1. Join 4 tables together   use repair\_shop;  SELECT device\_name,CONCAT(customers.first\_name," ",customers.last\_name)AS Cust\_full\_name,start\_time,  employee.first\_name AS Empl\_fname,repair\_fee  FROM customers JOIN device\_info USING(customer\_id)  JOIN session\_info USING(device\_id)  JOIN employee USING (employee\_id) JOIN  repairs USING (repair\_id) JOIN parts USING (part\_id)  LIMIT 5; | |  | 3 |
| 5 | | Formatting date and math   1. Use your date column, either calculate the number of days from the date to today or if you have 2 date columns, subtract 1 date from another   SELECT device\_name,DATEDIFF(CURRENT\_DATE(),CHECK\_IN\_DATE)  AS DaysNum FROM session\_info  JOIN device\_info USING(device\_id)  GROUP BY device\_name;     1. Round numeric data to 2 decimals   SELECT ROUND(AVG(total\_due),2)FROM repairs;     1. Display your date data. Format the date to the day name, month name, date, and 4-digit year   SELECT CONCAT(dayname(check\_in\_date),”/ ”,monthname(check\_in\_date)) as DayAndMo  Year(check\_in\_date) as `year` from session\_info;     1. Display your time zone   SELECT @@system\_time\_zone AS OHIO\_TZONE; | |  | 8 |
| 6 | | Subqueries and merge statements: Write a query for each of the tasks below. In other words, do not combine two tasks in the same query   1. Create a subquery that calculates the average of the numeric column with nulls and then if the row has a null, replace the null with the average number   SELECT total\_due,  IFNULL(total\_due, avg(total\_due)) as new\_col  FROM repairs;     1. Create a subquery with where and order by clauses   SELECT part\_name FROM parts JOIN repairs USING(part\_id) WHERE repair\_fee > (  SELECT AVG(part\_price+total\_due)AS totalSpent  FROM parts JOIN repairs USING(part\_id)  WHERE part\_type LIKE 'ap%'  ORDER BY part\_name ASC);     1. Create a subquery with a extract data from one table and use it to limit data from another table (for examples look at Chapter 7’s lecture notes, page 1 the vendor ID example)      1. Use a subquery with a *not in* operator   SELECT CONCAT('$' , SUM(repair\_fee) )AS Non\_insurance\_payments  FROM repairs  WHERE repair\_id NOT IN(  SELECT repair\_id  FROM parts JOIN repairs USING(part\_id) JOIN session\_info USING(repair\_id)JOIN insurance USING(session\_id)  WHERE insurance\_type IS NOT NULL);     1. Use a subquery in a DML action   UPDATE parts  SET part\_name = 'cheapest part'  WHERE part\_id IN (SELECT part\_id FROM(SELECT part\_id FROM parts  JOIN repairs USING(part\_id)  WHERE repairs\_fee <= 80) AS newTable);  SELECT\* FROM parts     1. Create a query with a CTE (with statement)   WITH rate\_emp AS(  SELECT CONCAT(first\_name," ",last\_name) as full\_name,quoted\_minutes,total\_min\_spent  FROM session\_info JOIN employee USING(employee\_id))  SELECT full\_name  FROM rate\_emp  WHERE total\_min\_spent <= quoted\_minutes | |  | 18 |
| 7 | | Views   1. Create a simple view   -- display employees who met customer wait time  CREATE OR REPLACE VIEW best\_perfomers AS  SELECT first\_name, last\_name, position  from employee join session\_info using(employee\_id)  WHERE total\_min\_spent <= quoted\_minutes  order by quoted\_minutes ASC;  SELECT\* FROM best\_perfomers     1. Create a view with a check option   CREATE OR REPLACE VIEW simple\_view AS  SELECT part\_id, total\_due,insurance\_type  FROM parts JOIN repairs USING(part\_id) JOIN session\_info USING(repair\_id)JOIN insurance USING(session\_id)  WHERE insurance\_type IS NOT NULL  WITH CHECK OPTION;     1. Create a complex view   CREATE OR REPLACE VIEW sub\_emp AS  SELECT CONCAT(first\_name," ",last\_name) AS Full\_name, hire\_date  FROM employee  GROUP BY position  HAVING hire\_date < '2019-03-08'  ORDER BY hire\_date ASC;  select\* from sub\_emp;     1. Update a record in a simple view   set autocommit = 0;  UPDATE simple\_view  SET total\_due = total\_due + 200  WHERE insurance\_type LIKE '1%';  SELECT\* FROM simple\_view     1. Display only the to 10 records in one of your tables   use repair\_shop;  select\*from customers  order by first\_name DESC  LIMIT 10; | |  | 5 |
| 8 | | Indexes   1. Create an index (other than unique or primary key)   USE repair\_shop;  CREATE INDEX check\_date  ON session\_info(device\_id, check\_in\_date, start\_time);  SELECT device\_id,check\_in\_date,start\_time  FROM session\_info  USE INDEX (check\_date)  WHERE  device\_id IN(1,2,3,4)  ORDER by check\_in\_date ASC;     1. List all indexes using a query   SHOW INDEX FROM session\_info ; | |  | 4 |
| 9 | | QUESTION: Describe the advantages and disadvantages of indexes  The advantages include efficiency as they speed up the search process. Instead of searching the whole database, a query will go directly to the specific index.  The downside is that they may slow down insert, delete, update functions on column you have put them on.  Also, indexes can slow down the whole database if you have too many of them. It’s recommended to create just a few on columns that are highly queried on. | |  | 1 |
| 10 | | Procedures and functions   1. Create and call a stored procedure that reads data from a table   DELIMITER //  CREATE PROCEDURE AboutDevice()  BEGIN  SELECT DISTINCT CONCAT(customers.first\_name," ",customers.last\_name) AS full\_name, device\_name,check\_in\_date FROM session\_info  JOIN device\_info USING(device\_id) JOIN customers USING(customer\_id)  WHERE check\_in\_date = '2021-08-09';  END //  DELIMITER ;  CALL AboutDevice;     1. Create and call a stored procedure that has a case statement   Select employee.first\_name,quoted\_minutes,total\_min\_spent,  case WHEN total\_min\_spent < quoted\_minutes THEN  'EXCELLENT'  WHEN total\_min\_spent = quoted\_minutes THEN  'GOOD JOB'  ELSE  ' '  END as perform\_rating from session\_info join employee using(employee\_id);     1. Create and call a stored procedure that has an if statement   drop procedure if exists if\_statement;  DELIMITER //  CREATE PROCEDURE if\_statement()  BEGIN  SELECT repair\_id,total\_due,  IF (total\_due > 85,"under warranty","no warranty") as Message  FROM repairs;  END //  DELIMITER ;  call if\_statement();     1. Create and call a stored procedure that has a conditional handler for SQLWARNING   drop procedure if exists sql\_warning;  DELIMITER //  CREATE PROCEDURE sql\_warning()  BEGIN  DECLARE null\_entry TINYINT DEFAULT FALSE;  DECLARE CONTINUE HANDLER FOR 1048  SET null\_entry= TRUE;  INSERT INTO customers VALUES("16",NULL,"Jamal","jamy@yahoo.com","9345678987");  if null\_entry then select " first name value can't be null" as Warning;  end if;  END //  DELIMITER ;  call sql\_warning();     1. Create a procedure that has a transaction   drop procedure if exists sql\_transac;  DELIMITER //  CREATE PROCEDURE sql\_transac()  BEGIN  DECLARE sql\_error TINYINT DEFAULT FALSE;  DECLARE CONTINUE HANDLER FOR sqlexception  SET sql\_error= TRUE;  START TRANSACTION ;  INSERT INTO customers VALUES("16","Marlene","Jamal","jamy@yahoo.com","9345678987");  INSERT INTO customers VALUES("15","Marlene","Jamal","jamy@yahoo.com","9345678987");  if sql\_error = false  then commit;  select "row updated " as sql\_message;  else  rollback;  select " update encounted error " as sql\_message;  end if;  END //  DELIMITER ;  call sql\_transac();     1. Create and use a stored function   -- How many deviced we checked in on a certain date?  Use repair\_shop;  DROP PROCEDURE getcheckindate ;  DELIMITER //  CREATE PROCEDURE getcheckindate (IN custid DATE)  BEGIN  SELECT COUNT(device\_id)as Num\_Of\_Devices FROM session\_info  WHERE check\_in\_date = custid;  END //  DELIMITER ;  CALL getcheckindate ("2021-08-09");  SELECT\* FROM SESSION\_INFO; | |  | 18 |
| 11 | | Triggers   1. Create and use an update trigger   use repair\_shop;  ALTER TABLE device\_info  ADD COLUMN `status` varchar(20);  DROP TRIGGER IF EXISTS statuses;  DELIMITER //  CREATE TRIGGER statuses  BEFORE UPDATE  ON device\_info FOR EACH ROW  BEGIN  IF new.issue= "cracked display" THEN  SET new.`status`= "NEEDS NEW DISPLAY";  ELSE  SET new.`status`= "OTHER ISSUE";  END IF;  END//  DELIMITER ;  -- test trigger  SET AUTOCOMMIT = 0;  UPDATE device\_info  SET issue = "liquid damage"  WHERE device\_id = 3 ;  select\* from device\_info;     1. Create and use a trigger that saves data to a separate table   use repair\_shop;  DROP TABLE IF EXISTS new\_table;  CREATE TABLE new\_table AS (SELECT employee\_id,first\_name,last\_name,hire\_date , manager\_id,`position` FROM employee);  DROP TRIGGER IF EXISTS pos\_trigger;  CREATE TRIGGER pos\_trigger  AFTER UPDATE  ON employee FOR EACH ROW  INSERT INTO new\_table  VALUES (old.employee\_id,old.first\_name,old.last\_name,old.hire\_date , old.manager\_id,new.hire\_date)  ;  -- test trigger  UPDATE employee  SET hire\_date = "2022-09-08"  WHERE employee\_id = 3;  SELECT\* FROM new\_table; | |  | 6 |
| 12 | | Undo   1. Show a rollback using a savepoint   START TRANSACTION ;  SAVEPOINT before\_any\_update;  INSERT INTO customers VALUES("16","Marlene","Jamal","jamy@yahoo.com","934.567.8987");  rollback to before\_any\_update;  COMMIT;  select\* from customers; | |  | 2 |
| 13 | | Write a security plan for your database. Assume you have at least 4 employees. List all the tables and who will have access. Make use of the 4 privilege levels   1. What users would you create?   Root/admin users  Managers  Leads  Regular employees   1. What roles would you create?   If it’s a big corporate, I would create a group of IT admins and grant them roles to select, insert, delete, update, add, drop with all admin option on all databases and tables  A second tier would be a group of managers with roles to manipulate all columns is a specified tables in specified databases but without a privilege to drop columns. Just select, insert, update, or delete rows  I would also create users(leads) who have permissions over certain tables in a database and give them role to just select, update and insert  Last but not least, low level users would be granted permissions to play with a few specified columns   1. What privileges would you grant?   Refer to number 2. | |  | 6 |
| 14 | | Create a user with the password root  Grant them select, insert, update on only one database/schema  CREATE USER IF NOT EXISTS 'database\_user1'@'LOCALHOST' IDENTIFIED BY 'root';  GRANT select, insert, update  on repair\_shop.\*  TO 'database\_user1'@'LOCALHOST' ;  Display the list of users  select \* from mysql.user; | |  | 4 |
| 15 | | QUESTION: Identify each of the error codes below and explain how you will fix code   1. 1288   Trying to delete a row in an underlying table from a view. I understand if you remove WITH CHECK OPTION it will let you delete the row.   1. 1395   the number of rows that would be deleted from your view does not match the number of rows that would be deleted from the underlying tables.   1. 1451   This error means you are trying to delete a record that has a parent from another table or when a PK datatype is not same as a FK datatype. You should first delete or drop the foreign key before you perform deletion   1. 1146   Most of the time this happened to me was when I was trying to query a table that does not exist. How to correct it? By double checking for improper table names   1. List and describe an error that you got when creating your dbases and completing this exam’s tasks    * + Table, column does not exist due to improper naming      + View, procedure already exists after I forgot to first delete the existing ones      + Duplicate entry when trying to save a similar value on a unique column      + Commands out of sync error when trying to call functions out of the order      + No database was selected occurs when you run a command without using selecting a database first | |  | 5 |
| 16 | | QUESTION: What would happen if you try to delete a record that has a parent from another table?  we will get an error message based on the foreign key constraint. | |  | 2 |
| 17 | | Show a snapshot of the Schema Inspector | |  | 2 |
| 18 | | Which tables will you set transaction isolation level and WHY | |  | 2 |
| 19 | | Name the types of log files and how you access them  Binary log,slow query log,general log,error log,relay log. In mysql workbench ,you can access those by opening administration the server logs | |  | 2 |
| 20 | | Take a screenshot of the following  The Users and Privileges    Status and System Variables | |  | 2 |
| 21 | | Why create global variables  Any new connection starts with those global variable settings then you may override them later with session settings if you need to. Global variables affect the overall operation of the server while session only affect the current connection to the server. | |  | 2 |
|  |  | |