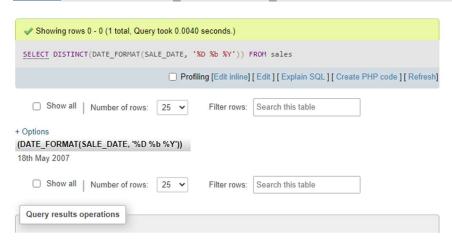
TASK 7.1

SELECT DISTINCT(SALE_DATE) FROM `sales`



Task 7.2

SELECT DISTINCT(DATE FORMAT(SALE DATE, '%D %b %Y')) FROM sales



Task 7.3

SELECT CURRENT DATE(), CURRENT TIME()



Task 7.3 (2)

SELECT EMP_LNAME, EMP_FNAME, EMP_DOB, DATE_FORMAT(EMP_DOB, '%b') AS MONTH

FROM employee WHERE month(EMP DOB) = '11'



Task 7.4 (2)

SELECT DATEDIFF ('2009-12-25', '2008-12-25')



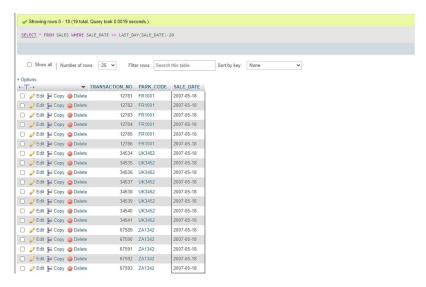
Task 7.6

SELECT EMP_LNAME, EMP_FNAME, EMP_HIRE_DATE, ADDDATE(EMP_HIRE_DATE, INTERVAL 12 MONTH) AS 'FIRST APPRAISAL' FROM EMPLOYEE



Task 7.7

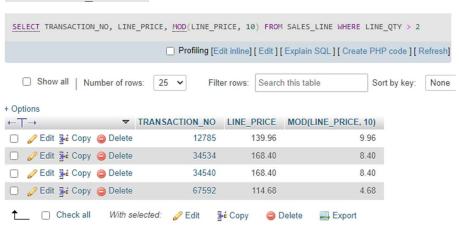
SELECT * FROM SALES WHERE SALE DATE >= LAST DAY(SALE DATE)-20



Task 7.8

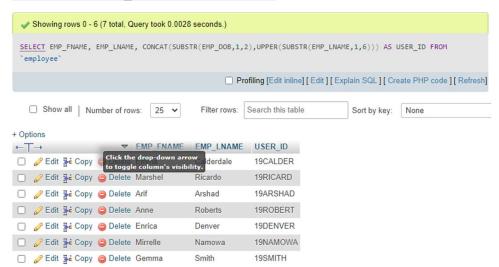
SELECT TRANSACTION_NO, LINE_PRICE, MOD (LINE_PRICE, 10) FROM SALES_LINE

WHERE LINE QTY > 2



Task 7.10

SELECT EMP_FNAME, EMP_LNAME, CONCAT(SUBSTR(EMP_DOB, 1, 2), UPPER(SUBSTR(EMP_LNAME, 1, 6))) AS USER ID FROM `employee`



EX 7.1

SELECT EMP_DOB, CONCAT(EMP_FNAME," ",EMP_LNAME) AS EMP_NAME
FROM `employee` WHERE DAYOFMONTH(EMP_DOB) = '14'



EX 7.2

SELECT CONCAT(EMP_FNAME," ",EMP_LNAME) AS EMP_NAME,
DATEDIFF('2008-11 -25', EMP DOB)/365 AS EMP APPR AGE FROM employee



EX 7.3

SELECT CONCAT (SUBSTR (EMP PHONE, 1, 3), LOWER (SUBSTR (EMP FNAME, 1, 2)))

AS EMP PASSWORD FROM employee

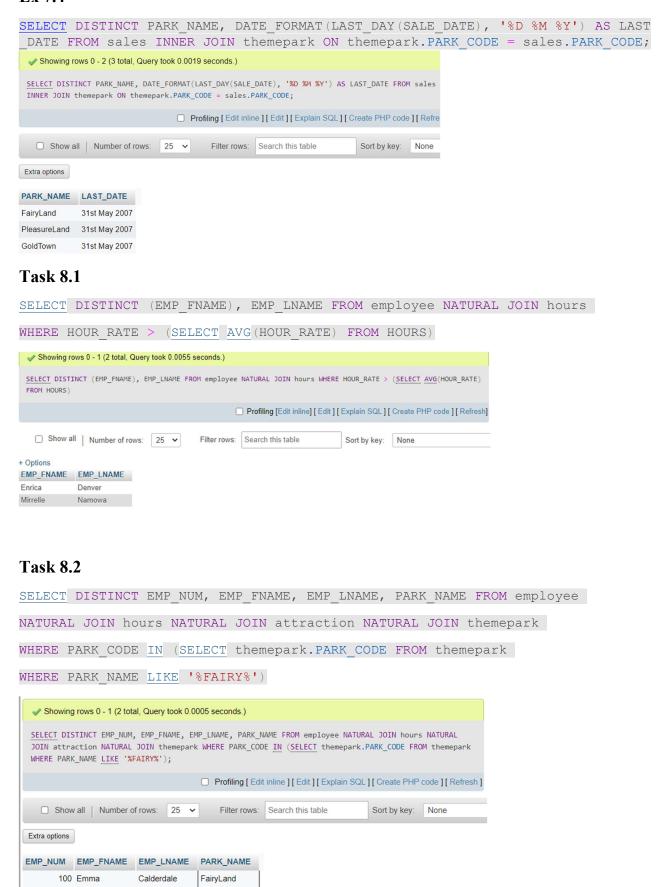


Ex 7.4

105 Mirrelle

Namowa

FairyLand



Task 8.3

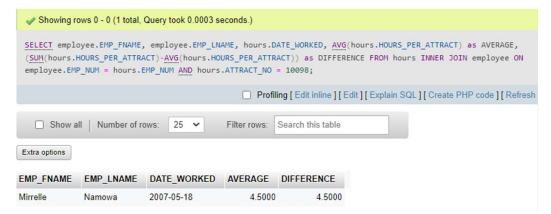
SELECT DISTINCT EMP_FNAME, EMP_LNAME, SUM(HOURS_PER_ATTRACT) AS SUM_HOURS FROM employee NATURAL JOIN hours WHERE DATE WORKED LIKE '%-05-%'

GROUP BY EMP_FNAME, EMP_LNAME, DATE_WORKED HAVING SUM_HOURS > (SELECT AVG (HOURS PER ATTRACT) FROM hours);



Task 8.4

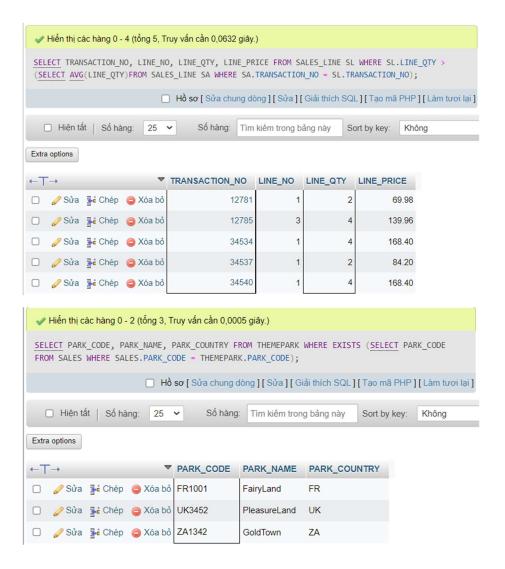
SELECT employee.EMP_FNAME, employee.EMP_LNAME, DATE_WORKED, AVG(HOURS_PER_ATT
RACT) as AVERAGE, (SUM(hours.HOURS_PER_ATTRACT) AVG(HOURS_PER_ATTRACT)) as DIFFERENCE FROM hours INNER JOIN employee ON emplo
yee.EMP NUM = hours.EMP NUM AND hours.ATTRACT NO = 10098;



Task 8.5

SELECT TRANSACTION_NO, LINE_NO, LINE_QTY, LINE_PRICE FROM SALES_LINE SL WHERE S
L.LINE_QTY > (SELECT AVG(LINE_QTY) FROM SALES_LINE SA WHERE SA.TRANSACTION_NO =
SL.TRANSACTION_NO);

SELECT PARK_CODE, PARK_NAME, PARK_COUNTRY FROM THEMEPARK WHERE EXISTS (SELECT PARK CODE FROM SALES WHERE SALES.PARK CODE = THEMEPARK.PARK CODE);



Task 8.6

SELECT PARK CODE, PARK NAME, PARK COUNTRY FROM THEMEPARK WHERE NOT EXISTS (SELECT PARK CODE FROM SALES WHERE SALES.PARK CODE = THEMEPARK.PARK CODE)
GROUP BY PARK COUNTRY;



Task 9.1

CREATE VIEW TPARKSSOLD AS SELECT *FROM THEMEPARK WHERE EXISTS (SELECT PARK_CODE
FROM SALES WHERE SALES.PARK CODE = THEMEPARK.PARK CODE);



Task 9.2

CREATE VIEW TICKET_SALES AS SELECT themepark.PARK_NAME, min(LINE_PRICE), max(LINE_PRICE), avg(LINE_PRICE) FROM sales_line NATURAL JOIN sales NATURAL JOIN themepark WHERE themepark.PARK_CODE = (SELECT sales.PARK_CODE WHERE sales_line.TRANSACTION NO = sales.TRANSACTION NO) GROUP BY themepark.PARK NAME;



Task 9.4

CREATE VIEW EMPFR AS SELECT * FROM EMPLOYEE WHERE PARK_CODE = 'FR1001' WITH CHECK OPTION;



UPDATE empfr SET PARK CODE = 'FR1001' WHERE EMP NUM = 101;

(VIEW không đổi do empfr không chứa EMP_NUM=101 \Rightarrow update thẳng table employee chứ không phải thẳng view này)

Task 9.5



Task 9.6

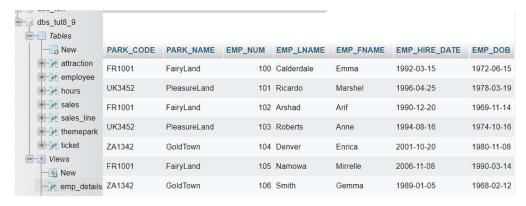
DROP VIEW empfr;

Ex9.1

CREATE VIEW EMP_DETAILS AS

SELECT themepark.PARK_CODE, themepark.PARK_NAME,EMP_NUM, EMP_LNAME, EMP_FNAME,
EMP_HIRE_DATE, EMP_DOB FROM employee INNER JOIN themepark ON themepark.PARK_C
ODE = employee.PARK CODE;

EX9.2



EX 9.3

SELECT EMP FNAME, EMP LNAME, PARK NAME FROM `emp details`;

EMP_FNAME	EMP_LNAME	PARK_NAME
Emma	Calderdale	FairyLand
Marshel	Ricardo	PleasureLand
Arif	Arshad	FairyLand
Anne	Roberts	PleasureLand
Enrica	Denver	GoldTown
Mirrelle	Namowa	FairyLand
Gemma	Smith	GoldTown

9.4.

DROP VIEW `emp details`;