Bresia Prudente

CS 480 - HW 3

3/15/2016

**1)(a)**

SELECT DISTINCT c.Customer\_ID, c.Customer\_Name

FROM Customers c

JOIN Bookings b ON c.Customer\_ID=b.Customer\_ID

JOIN Rooms r ON b.Room\_ID=r.Room\_ID

WHERE r.price=200 OR r.price=150;

**(b)**

SELECT DISTINCT c.Customer\_ID, c.Customer\_Name

FROM Customers c

JOIN Bookings b ON c.Customer\_ID=b.Customer\_ID

WHERE b.Room\_ID NOT IN (

SELECT r.Room\_ID

FROM Rooms r

WHERE r.price=100

);

**(c)**

SELECT c.Customer\_ID, c.Customer\_Name, COUNT(\*) AS totalCount

FROM Customers c

WHERE Customer\_ID IN (

SELECT b.id

FROM Bookings b

JOIN Rooms r ON b.Room\_ID=r.Room\_ID

)

GROUP BY c.Customer\_Name

HAVING COUNT(\*) >= 2;

**(d)**

SELECT c.Customer\_ID, c.Customer\_Name, COUNT(\*) AS totalCount

FROM Customers c

GROUP BY c.Customer\_Name

HAVING COUNT(\*) >= 3;

**(e)**

SELECT c.Customer\_ID, c.Customer\_Name, COUNT(\*) AS totalCount

FROM Customers c

WHERE Customer\_ID IN (

SELECT b.id

FROM Bookings b

JOIN Rooms r ON b.Room\_ID=r.Room\_ID

)

GROUP BY c.Customer\_Name

HAVING COUNT(\*) >= 2;

**2) (a)**

SELECT e.E\_ID, e.Ename

FROM Employee e

JOIN Works w ON e.E\_ID=w.E\_ID

JOIN Airplane a ON w.A\_ID=a.A\_ID

WHERE a.FlyingRange>=5000;

**(b)**

SELECT e.E\_ID, e.Ename

FROM Employee e

JOIN Works w ON e.E\_ID=w.E\_ID

JOIN Airplane a ON w.A\_ID=a.A\_ID

WHERE a.FlyingRange>=5000

HAVING COUNT(\*)>2;

**(c)**

DELETE FROM Airplane

WHERE FlyingRange>40000000;

This will also remove values associated with the key A\_ID in the Works table.

**(d)**

INSERT INTO Airplane(A\_ID, Model, FlyingRange, TotalMiles)

VALUES

(2345, 'Boeing 737', 3000, 0);

This will update the Airplane table and also update values associated with the key A\_ID in the Works table.

**(e)**

UPDATE Employee

SET Stipend=0

WHERE Ename='Manfred von Richthofen';

This will not affect any of the other tables. Only the Employee table will be affected.

**3)(a)**

Assuming that the users are connecting to the database locally:

CREATE USER Alice@localhost IDENTIFIED BY 'alice\_pass';

CREATE USER Bob@localhost IDENTIFIED BY 'bob\_pass';

CREATE USER Jane@localhost IDENTIFIED BY 'jane\_pass';

**(b)**

GRANT ALL PRIVILEGES ON DDL.\* TO Alice@localhost;

**(c)**

GRANT SELECT ON DDL.\* TO Bob@localhost;

**(d)**

GRANT SELECT, UPDATE ON DDL.\* TO Jane@localhost;

REVOKE SELECT, UPDATE ON DDL.instructor FROM Jane@localhost;

GRANT DELETE, INSERT ON DDL.instructor TO Jane@localhost;

**4)(a)**

|  |  |  |  |
| --- | --- | --- | --- |
| E\_ID | A\_ID | Ename | Stipend |
| 3215 | 4567 | Jane Zimmer | 80000 |

**(b)**

|  |
| --- |
| FlyingRange |
| 4000 |

**(c)**

|  |
| --- |
| E\_ID |
| 1234 |
| 4567 |

**(d)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| E\_ID | A\_ID | Model | Flying Range | TotalMiles |
| null | 1234 | Boeing 737 | 3000 | 30000000 |
| 3215 | 4567 | Boeing 737 | 4000 | 10000000 |
| 4312 | 7890 | Airbus A300 | 4000 | 0 |

**5)(a)**

CREATE FUNCTION get\_total\_budget(limit) RETURNS INT

BEGIN

DECLARE result INT;

SET result=0;

SELECT SUM(budget) AS budgetTotal FROM department;

IF limit > budget THEN

SET result=limit;

ELSEIF limit < budget THEN

SET result=budget;

ELSE

SET result=0;

END IF;

RETURN(result);

END

**(b)**

CREATE FUNCTION get\_total\_credits(dept\_name) RETURNS INT

BEGIN

DECLARE total\_credits INT;

SET total\_credits=0;

SELECT SUM(tot\_cred) INTO total\_credits

FROM student

JOIN takes ON student.ID=takes.ID

WHERE takes.semester='Fall' AND takes.year=2009;

RETURN(total\_credits);

END

**(c)**

SELECT get\_total\_budget(1000);

|  |
| --- |
| budgetTotal |
| 595000 |

SELECT get\_total\_credits('Physics');

|  |
| --- |
| SUM(tot\_cred) |
| 56 |
| 46 |