Consider the following schema

I:

Salesperson(sid:integer, sname:string, age:real)

Sales(sid:integer, pid:integer, day:date)

Products(pid:integer, pname:string, rating:integer, category:string)

1. Count the no. of each product sales by salesperson with name “ramana”
2. Find the no. of products which are not listed in sales relation
3. Find the name and age of the salespersons who have done sales of all different products
4. Display the product details which are better in rating than product named “yippie” in category “noodles”
5. Find the name of salesperson who have done sales of both products “Maggie” and “Good day”

Salesperson:

|  |  |  |
| --- | --- | --- |
| **sid** | **sname** | **age** |
| 10 | ramana | 25 |
| 11 | raju | 30 |
| 12 | vimala | 28 |

Sales:

|  |  |  |
| --- | --- | --- |
| **sid** | **pid** | **day** |
| 10 | 121 | 01/11/15 |
| 10 | 111 | 01/11/15 |
| 10 | 111 | 02/11/15 |
| 11 | 121 | 03/11/15 |
| 12 | 121 | 03/11/15 |

Product:

|  |  |  |  |
| --- | --- | --- | --- |
| **pid** | **pname** | **rating** | **category** |
| 101 | yippie | 6 | noodles |
| 111 | maggie | 8 | noodles |
| 121 | Good day | 8 | biscuits |

II:

Consider the following schema

Player(pname:string, country:string, category:string)

Format (fname:string, oversperday:integer, days:integer)

Rankings(pname:string, fname:string, rank:integer)

1. Find the player name who are listed for all formats in Rankings table
2. Display the name and category of player who has highest rank for each format
3. Find the player names who are listed in rankings table only for “ODI” format but NOT for “Test” format
4. Add an attribute age to players table and put age of player named “kohli” to 27
5. Count the no. of players in each country

Players :

|  |  |  |
| --- | --- | --- |
| **pname** | **country** | **category** |
| Ashwin | India | Bowler |
| Kohli | India | Batsman |
| De villiers | South Africa | Batsman |

Rankings

|  |  |  |
| --- | --- | --- |
| **pname** | **fname** | **rank** |
| kohli | ODI | 2 |
| kohli | Twenty20 | 2 |
| Ashwin | ODI | 10 |
| Ashwin | Test | 2 |
| Ashwin | Twenty20 | 5 |
| Devilliers | ODI | 1 |

Format

|  |  |  |
| --- | --- | --- |
| **Fname** | **oversperday** | **days** |
| ODI | 100 | 1 |
| Twenty20 | 40 | 1 |
| Test | 90 | 5 |

III:

Consider the following schema

Faculty (fid:integer, fname:string, qualification:string, age:real)

Teaches(fid:integer, cid:string, year-sem:string)

Course(cid:string, cname:string, textbook:string)

1. Find faculty names whose qualification is “M.Tech” and teaches at least two different courses
2. Find faculty names whose age is below 30 and teaches all the courses
3. Find course name and textbooks of the courses taught by either “kavitha” or “venu”
4. Count the no. of courses for each “year-sem”
5. Alter the table courses to add a new attribute “credits” and update the credits to 4 for the courses taught by faculty with fid 10

Faculty

|  |  |  |  |
| --- | --- | --- | --- |
| **fid** | **fname** | **qualification** | **age** |
| 10 | kavitha | M.Tech | 32 |
| 11 | vanitha | M.C.A | 29 |
| 12 | venu | M.Tech | 33 |

Teaches

|  |  |  |
| --- | --- | --- |
| **fid** | **cid** | **Year-sem** |
| 10 | CS201 | 2-1 |
| 10 | CS301 | 3-1 |
| 12 | CS401 | 4-1 |
| 11 | CS301 | 3-1 |
| 11 | CS302 | 3-1 |

Course

|  |  |  |
| --- | --- | --- |
| **cid** | **cname** | **textbook** |
| CS201 | DBMS | Database concepts |
| CS301 | COA | Computer architecture |
| CS401 | JAVA | Complete reference JAVA |
| CS302 | FLAT | Formal languages |