1. Tampilkan seluruh ID dan Name dari Instructor beserta Jumlah Course yang telah diajar oleh Instructor tersebut baik Instructor tersebut pernah mengajar suatu course atau belum pernah sama sekali!

|  |  |
| --- | --- |
| 1  2  3 | SELECT instructor.ID, instructor.name, COUNT(course.dept\_name) AS Jml\_course\_yang\_diajar  FROM instructor full outer join course ON instructor.dept\_name = course.dept\_name  GROUP BY instructor.ID, instructor.name; |

|  |
| --- |
|  |

**Pembahasan dan Analisis**

**(ditulis tangan)** Kode implementasi digunakan untuk mengambil data xxx secara terurut. Data diambil dari tabel xxx dan xxx melalui operasi *join.* Baris ke-1 digunakan untuk xxx. Baris ke-2 menunjukkan xxx. Baris ke-3 menunjukkan xxx. Baris ke-4 menunjukkan xxx.

Dari percobaan dan analisis yang telah dilakukan, apabila baris 3 tidak ada maka hasilnya yyy. Sedangkan apabila baris 4 tidak ada maka hasilnya yyy, sehingga supaya dapat mengambil data xxx harus yyy dengan mempertimbangkan zzz. Dilanjutkan dengan penjelasan lain, dst.

1. Tampilkan Departement Name dan pembulatan ke bawah dari Rata-Rata Gaji Dosen-Dosen yang ada di tiap department dengan format tampilan rata-rata gaji tersebut menggunakan awalan 'Rp ' di depannya!

|  |  |
| --- | --- |
| 1  2  3 | SELECT department.dept\_name, CONCAT('Rp ',FLOOR(AVG(instructor.salary))) AS Rerata\_Gaji\_Department  FROM department right outer join instructor ON department.dept\_name = instructor.dept\_name  GROUP BY department.dept\_name; |

|  |
| --- |
|  |

**Pembahasan dan Analisis**

**(ditulis tangan)** Kode implementasi digunakan untuk mengambil data xxx secara terurut. Data diambil dari tabel xxx dan xxx melalui operasi *join.* Baris ke-1 digunakan untuk xxx. Baris ke-2 menunjukkan xxx. Baris ke-3 menunjukkan xxx. Baris ke-4 menunjukkan xxx.

Dari percobaan dan analisis yang telah dilakukan, apabila baris 3 tidak ada maka hasilnya yyy. Sedangkan apabila baris 4 tidak ada maka hasilnya yyy, sehingga supaya dapat mengambil data xxx harus yyy dengan mempertimbangkan zzz. Dilanjutkan dengan penjelasan lain, dst.

1. Tampilkan Course ID, Course Title, dan pembulatan ke atas dari rata-rata sks(tot\_cred) mahasiswa yang mengambil mata kuliah (course) tersebut dengan Course ID di atas 500!

|  |  |
| --- | --- |
| 1  2  3  4 | SELECT course.course\_id, course.title, CEILING(AVG(student.tot\_cred)) AS Rerata\_Total\_SKS  FROM course join takes on course.course\_id = takes.course\_id  Join student on takes.ID = student.ID  GROUP BY course.course\_id, course.title  HAVING course.course\_id >= 500; |

|  |
| --- |
|  |

**Pembahasan dan Analisis**

**(ditulis tangan)** Kode implementasi digunakan untuk mengambil data xxx secara terurut. Data diambil dari tabel xxx dan xxx melalui operasi *join.* Baris ke-1 digunakan untuk xxx. Baris ke-2 menunjukkan xxx. Baris ke-3 menunjukkan xxx. Baris ke-4 menunjukkan xxx.

Dari percobaan dan analisis yang telah dilakukan, apabila baris 3 tidak ada maka hasilnya yyy. Sedangkan apabila baris 4 tidak ada maka hasilnya yyy, sehingga supaya dapat mengambil data xxx harus yyy dengan mempertimbangkan zzz. Dilanjutkan dengan penjelasan lain, dst.