1. Tambahkan data kuliah baru (course dengan nama “Game Programming) beserta nilainya (ke table takes) oleh student bernama “Manber” menggunakan subquery dengan ketentuan:

Tabel takes hanya berisi ID student sehingga pengisian ID Student dari “Manber” harus dilakukan melalui subquery (tidak dengan cara mengisikan ID secara langsung/hard code).

Tabel takes hanya berisi ID course sehingga pengisian ID Course dari “Game Programming” harus dilakukan melalui subquery (tidak dengan cara mengisikan ID secara langsung/hardd code).Buat tabel copy\_instructor dengan struktur tabel yang sama dengan tabel instructor .

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41 | BEGIN TRANSACTION  BEGIN TRY  PRINT 'INSERT NEW GAME PROGRAMMING INTO COURSE'  INSERT INTO course (course\_id,title,dept\_name)  VALUES (1,'Game Programming',(  SELECT S.dept\_name  FROM student S  WHERE S.name='Manber'  ))  PRINT 'COMPLETE'  PRINT 'INSERT NEW GAME PROGRAMMING INTO SECTION'  INSERT INTO section (course\_id,sec\_id,semester,year)  VALUES ((  SELECT C.course\_id  FROM course C  WHERE C.title='Game Programming' AND C.dept\_name='Civil Eng.'),4,'Winter',2016)  PRINT 'COMPLETE'  PRINT 'INSERT NEW GAME PROGRAMMING INTO TAKES'  INSERT INTO takes (ID,course\_id,sec\_id,semester,year)(  SELECT ST.ID, C.course\_id, SE.sec\_id, SE.semester, SE.year  FROM student ST  JOIN course C  ON ST.dept\_name=C.dept\_name  JOIN section SE  ON C.course\_id=SE.course\_id  WHERE ST.name='Manber' AND C.title='Game Programming'  )  PRINT 'COMPLETE'  PRINT 'UPDATE GRADE FOR NEW GAME PROGRAMMING'  UPDATE takes  SET grade='E'  WHERE ID=1000 AND course\_id=1  COMMIT TRAN  PRINT 'COMPLETE'  END TRY  BEGIN CATCH  PRINT 'ERROR WHILE PROCESSING:ROLLBACK ALL'  IF @@TRANCOUNT > 0  ROLLBACK TRANSACTION  END CATCH |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |

Pembahasan dan analisis

Kode implementasi di atas digunakan untuk memasukkan course baru bernama Game Programming ke table course, memasukkan course baru bernama Game Programming ke table section, dan memasukkan course baru bernama Game Programming ke table takes menggunakan transaction dan try-catch. Pada block try (2-36), program akan mencetak ‘INSERT NEW GAME PROGRAMMING INTO COURSE’ lalu memasukkan data 1, ‘Game Programming’, dan hasil subquery baris 5-9 (yang menghasilkan data ‘Civil Eng.’ dari column student.dept\_name) ke columns course.course\_id, course.title, serta course.dept\_name. Setelah itu, program mencetak ‘COMPLETE’, ‘INSERT NEW GAME PROGRAMMING INTO SECTION’, lalu memasukkan data hasil subquery baris 13-17 (yang menghasilkan data 1 dari column course.course\_id), 4, ‘Winter’, dan 2016 ke columns section.course\_id, section.sec\_id, section.semester, serta section.year. Kemudian, program mencetak ‘COMPLETE’, ‘INSERT NEW GAME PROGRAMMING INTO TAKES’, lalu memasukkan data hasil subquery baris 20-28 (yang menghasilkan 1000, 1, 4, ‘Winter’, dan 2016 masing-masing dari columns student.ID, course.course\_id, section.sec\_id, section.semester, dan section.year) ke columns takes.ID, takes.course\_id, takes.sec\_id, takes.semester, serta takes.year. Terakhir, program mencetak ‘Complete’, ‘UPDATE GRADE FOR NEW GAME PROGRAMMING’, dan update pada table takes dengan merubah grade dengan takes.ID = 1000 dan takes.course\_id = 1 menjadi ‘E’, melakukan COMMIT pada transaction, dan mencetak ‘COMPLETE’. Program di atas menjalankan block try secara sempurna karena tidak terdapat error pada block try.

Pada block catch (baris 37-41), program mencetak ‘ERROR WHILE PROCESSING:ROLLBACK ALL’ dan melakukan ROLLBACK pada transaction bila variable TRANCOUNT > 1 (sedang dilakukan transaction). Block catch hanya dieksekusi bila terjadi error pada block try.

1. Apabila data kuliah baru berhasil ditambahkan maka update total credit (tot\_cred) pada table student sesuai SKS (credit) dari “Game Programming” melalui. Untuk mengetahui banyak SKS yang ditambahkan gunakan subquery.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30 | BEGIN TRANSACTION  BEGIN TRY  PRINT 'UPDATE CREDIT FOR NEW GAME PROGRAMMING'  UPDATE course  SET credits=  (  SELECT DISTINCT C.credits  FROM course C  WHERE C.title='Game Programming' AND C.dept\_name!='Civil Eng.'  )  WHERE title='Game Programming' AND dept\_name='Civil Eng.'  PRINT 'COMPLETE'  PRINT 'UPDATE TOT\_CRED FOR MANBER'  UPDATE student  SET tot\_cred+=  (  SELECT credits  FROM course  WHERE course\_id=1  )  WHERE name='Manber'  COMMIT TRAN  PRINT 'COMPLETE'  END TRY  BEGIN CATCH  PRINT 'ERROR WHILE PROCESSING:ROLLBACK ALL'  IF @@TRANCOUNT > 0  ROLLBACK TRANSACTION  END CATCH |

|  |
| --- |
|  |
|  |
|  |

Pembahasan dan analisis

Kode implementasi di atas digunakan untuk me-update course.credit dari course.title ‘Game Programming’ dan me-update student.tot\_cred sebesar course.credit yang dimasukkan sebelumnya menggunakan transaction dan try-catch. Pada block try (baris 2-25), program mencetak ‘UPDATE CREDIT FOR NEW GAME PROGRAMMING’ dan melakukan update pada course.credits dengan nilai sebesar hasil subquery baris 6-11 (yang menghasilkan nilai 4 dari column course.credits) pada row yang memiliki course.title dengan nama ‘Game Programming’ dan course.dept\_name = ‘Civil Eng.’. Setelah itu, program mencetak ‘COMPLETE’, ‘UPDATE TOT\_CRED FOR MANBER’, melakukan update pada student.tot\_cred dengan cara menambahkan nilai student.tot\_cred dari column student.name yang bernilai ‘Manber’ dengan hasil subquery baris 17-21, melakukan COMMIT pada transaction, dan mencetak ‘COMPLETE’. Program di atas menjalankan block try secara sempurna karena tidak terdapat error pada block try.

Pada block catch (baris 26-30), program mencetak ‘ERROR WHILE PROCESSING:ROLLBACK ALL’ dan melakukan ROLLBACK pada transaction bila variable TRANCOUNT > 1 (sedang dilakukan transaction). Block catch hanya dieksekusi bila terjadi error pada block try.**hasan 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