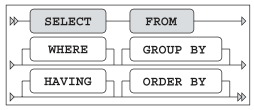
Retrieval: The Basics

Anda akan belajar cara *mengambil* data dari database Anda. Untuk pengambilan data, bahasa SQL menawarkan perintah SELECT. Perintah SELECT biasanya disebut sebagai *query.* Perintah SELECT memiliki enam klausa utama yaitu SELECT, WHERE, ORDER BY, FROM, GROUP BY, dan HAVING. Anda dapat menulis kueri sebagai pernyataan SQL independen, tetapi kueri juga dapat muncul di dalam perintah SQL lainnya. Ini disebut *subquery.* kondisi SQL memiliki tiga kemungkinan hasil TRUE, FALSE, atau UNKNOWN.

**Perintah SELECT**



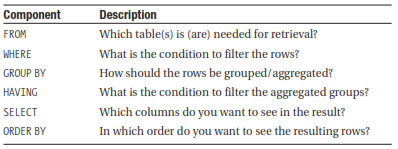
Gambar 1. Enam klausa utama perintah select

Aturan sintaks utama berikut

• Ada urutan wajib yang telah ditentukan sebelumnya dari enam klausa ini.

• Klausa SELECT dan FROM adalah wajib.

• WHERE, GROUP BY, HAVING, dan ORDER BY adalah klausa opsional.



Menurut standar ANSI/ISO SQL, enam klausa ini harus diproses dalam urutan berikut: FROM, WHERE, GROUP BY, HAVING, SELECT, ORDER BY. Perhatikan bahwa ini *bukan* urutan dalam yang harus Anda tentukan dalam kueri Anda.

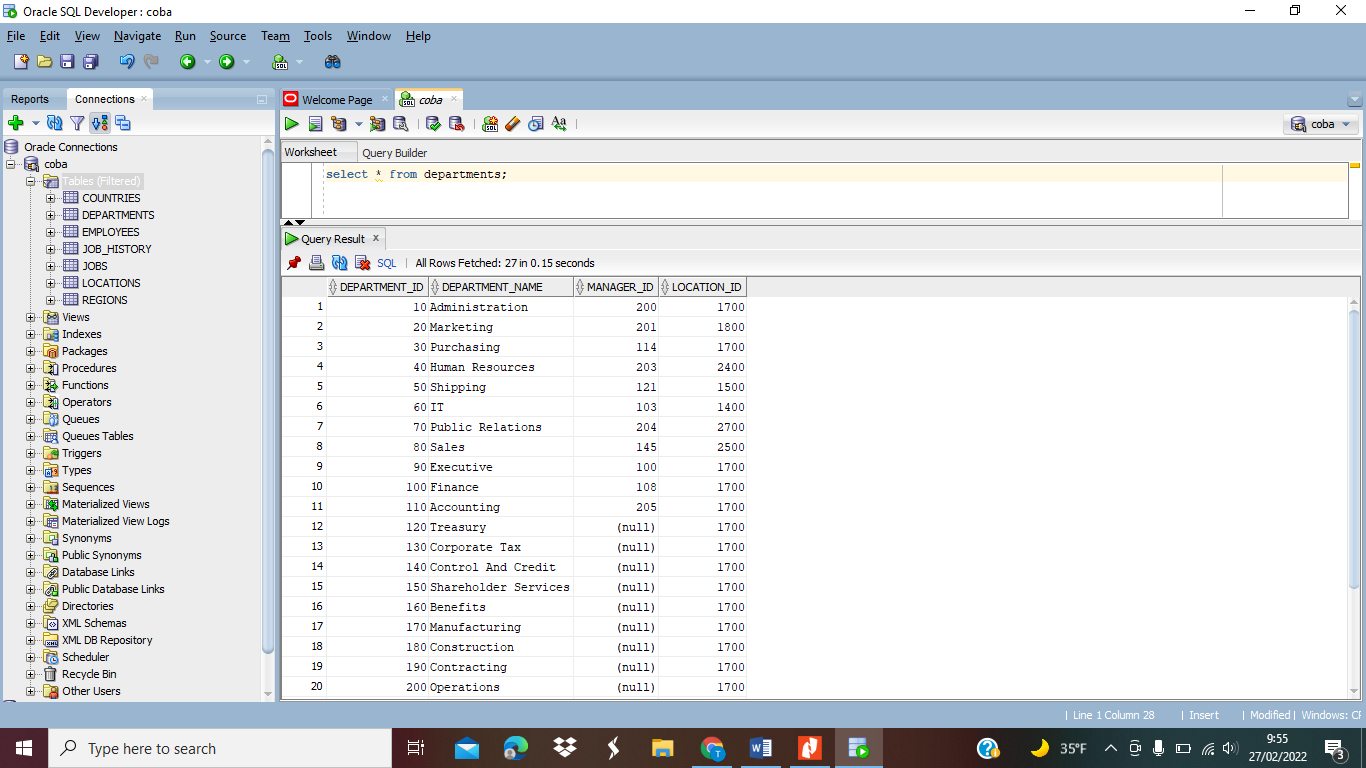
• **SELECT:** Dengan klausa SELECT dari perintah SELECT, menentukan kolom yang Anda ingin ditampilkan dalam hasil kueri dan, secara opsional, judul kolom mana yang Anda tuju. Klausa ini mengimplementasikan *proyeksi* relasional operator

• **WHERE:** Klausa WHERE memungkinkan Anda untuk merumuskan kondisi memfilter baris dari tabel dasar; mengimplementasikan operator *restriksi* relasional. Anda dapat menggunakan berbagai operator dalam kondisi klausa WHERE Anda—seperti BETWEEN, LIKE, IN, CASE, NOT, AND, dan OR—

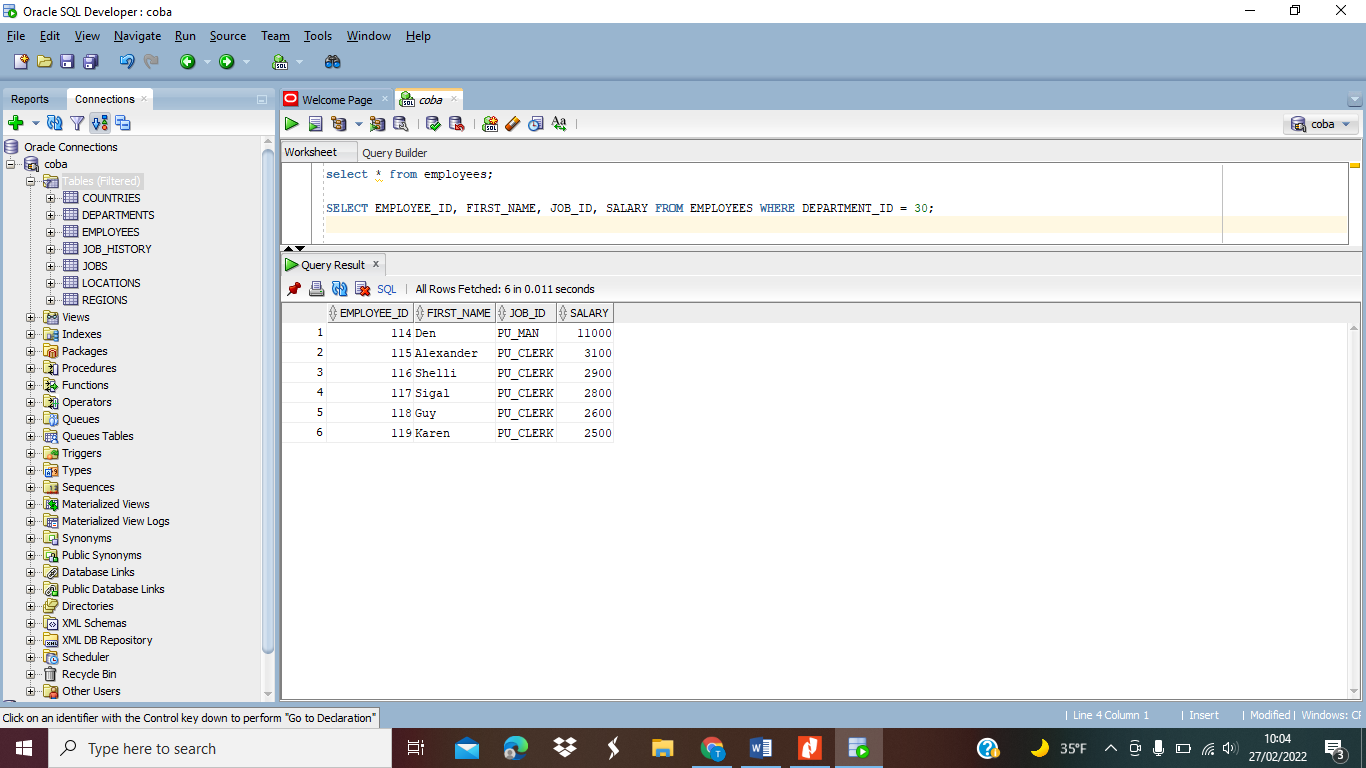
• **ORDER BY:** Dengan klausa ORDER BY, Anda menentukan urutan di mana Anda ingin melihat baris dalam hasil kueri Anda.

Klausa FROM memungkinkan Anda untuk menentukan tabel mana yang ingin Anda akses. Dalam bab ini, kita akan bekerja dengan kueri yang hanya mengakses satu tabel, atau mengakses beberapa tabel dalam satu kueri.

**The SELECT Clause**

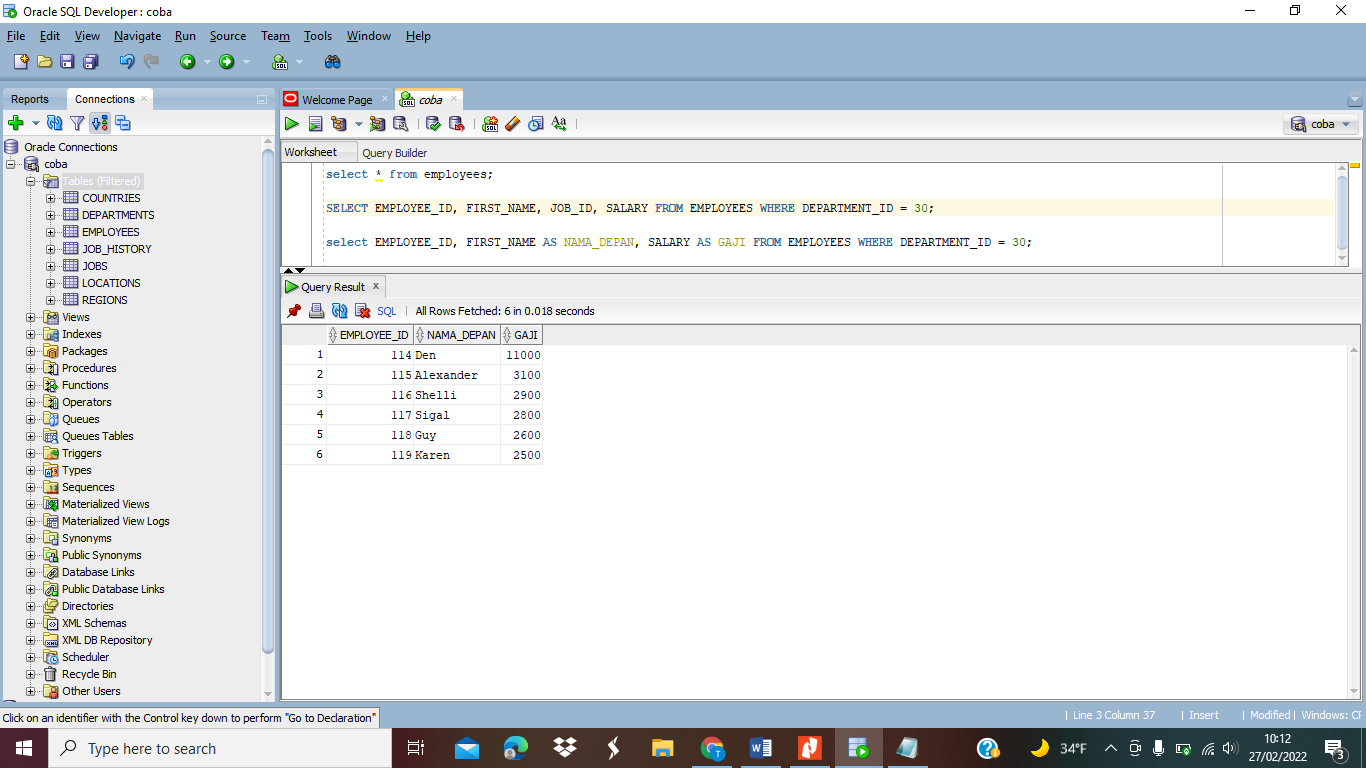


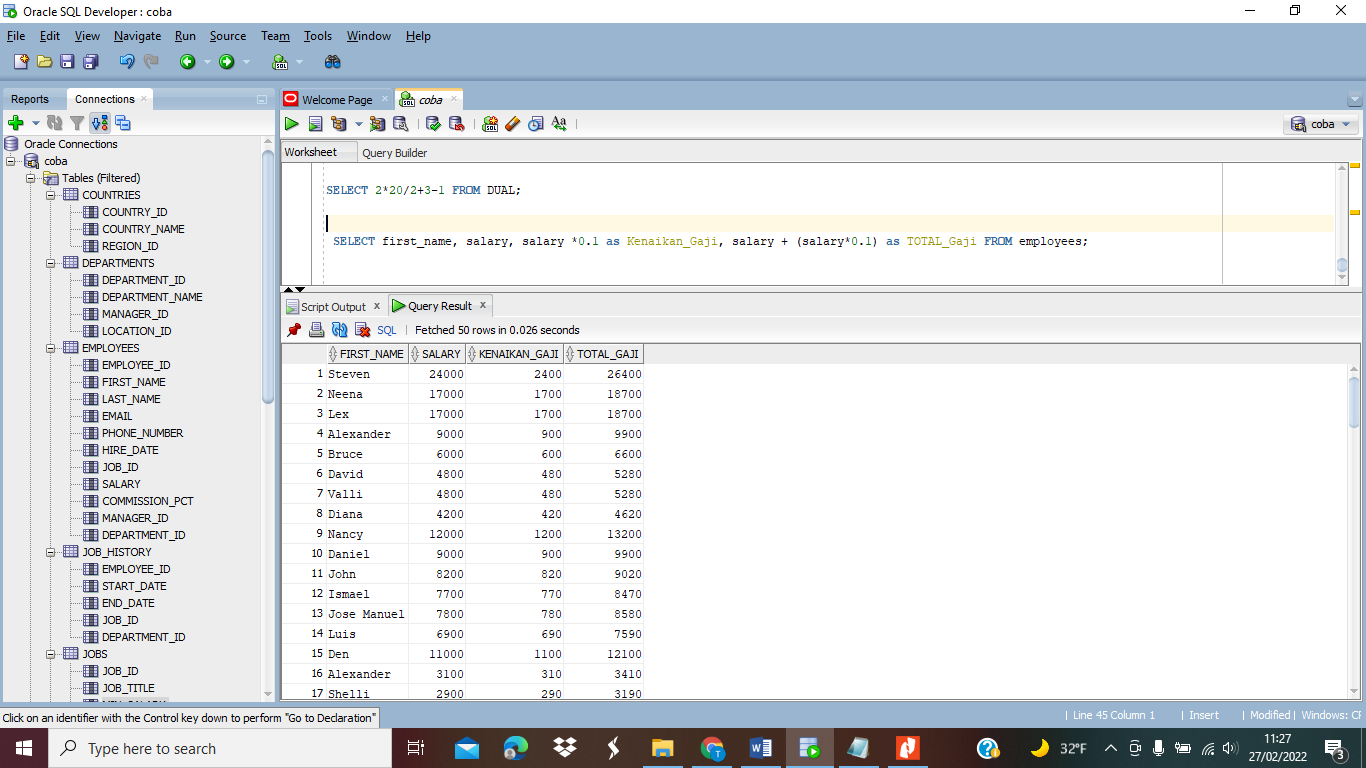
Tanda asterisk (\*) berarti menampilkan *semua* kolom dari tabel DEPARTMENTS



**Menggunakan Kolom Alias**

Mengubah Judul Kolom



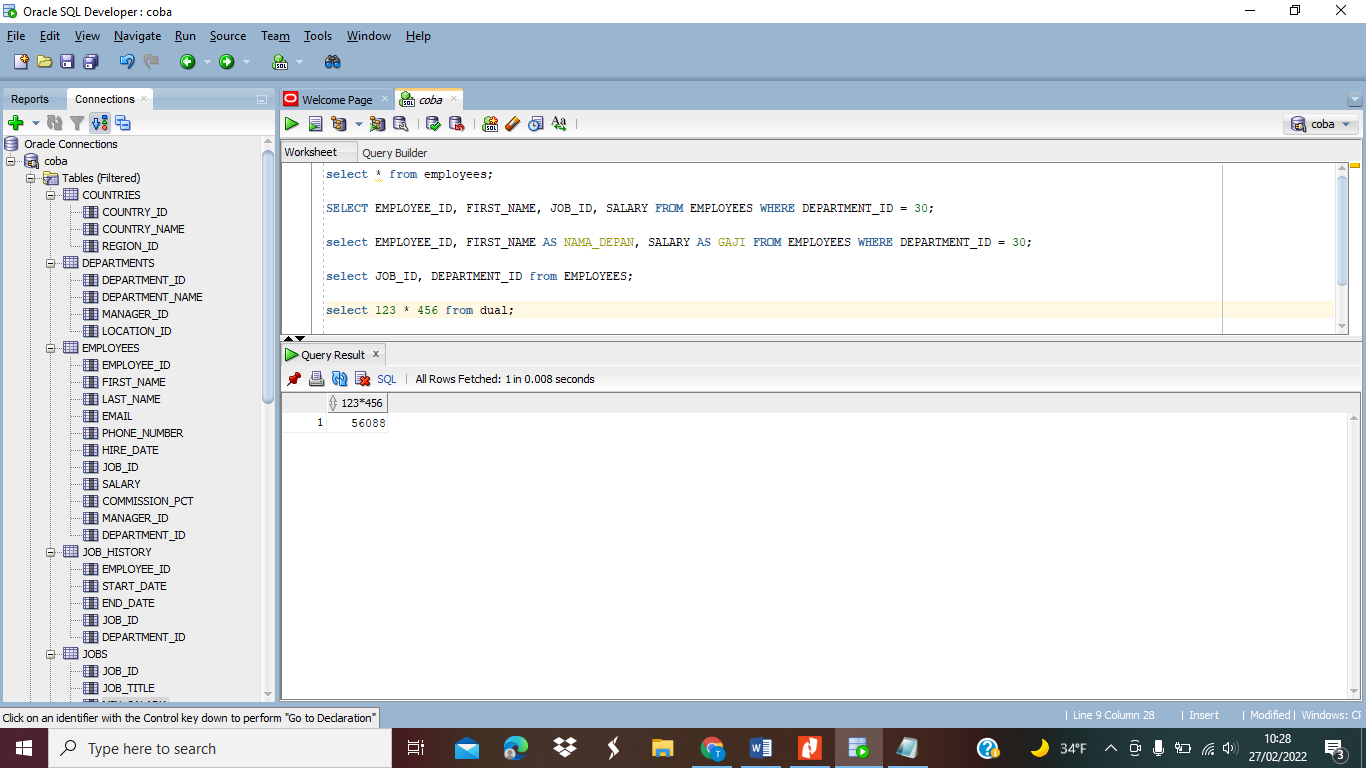
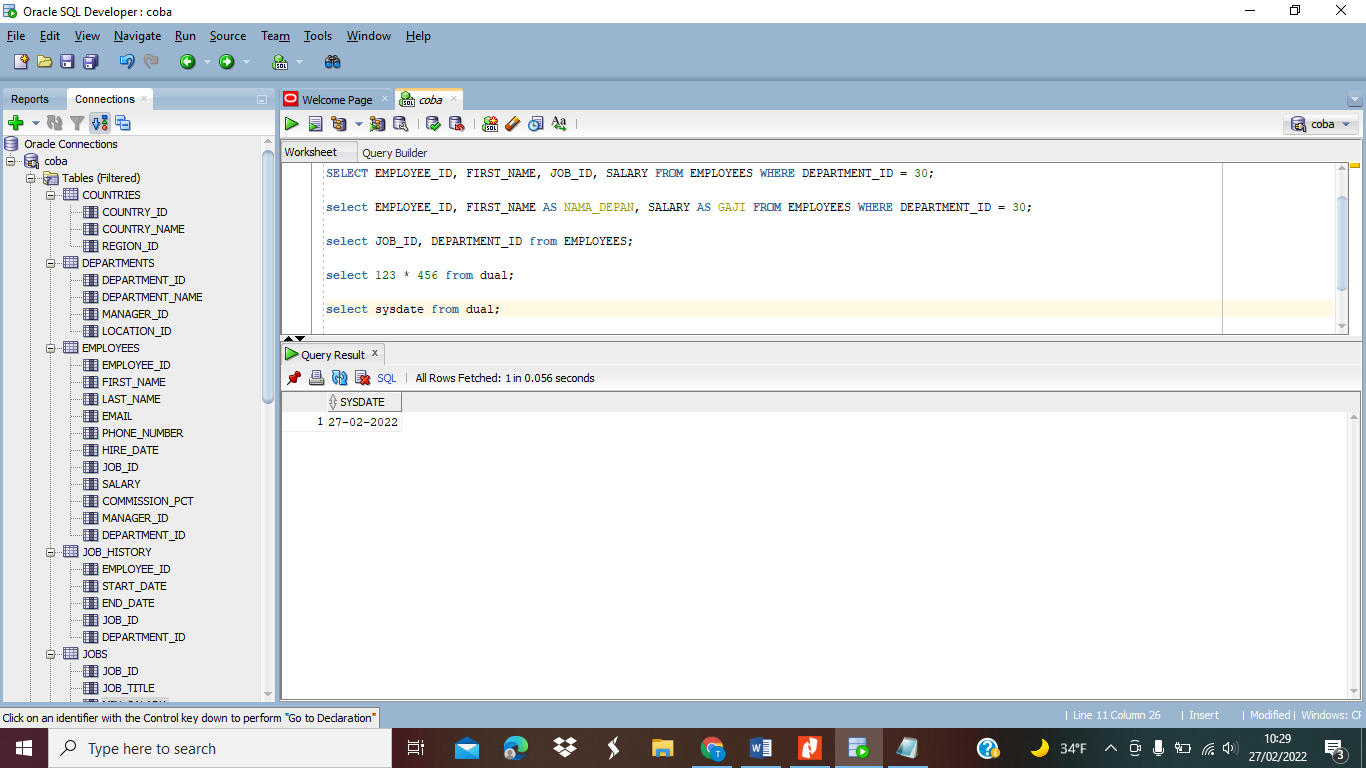


**The DISTINCT Keyword**

|  |  |
| --- | --- |
| **Distinct** Digunakan untuk menghilangkan duplikasi nilai.  **Perintah Distinct** dapat anda gunakan untuk menampilkan data yang berbeda – beda, atau dalam artian lain, **perintah Distinct** ini tidak akan menampilkan data yang sama sebanyak dua kali. Hal ini berguna jika anda hanya ingin menampilkan record dengan data yang berbeda untuk anda ambil intinya saja |  |

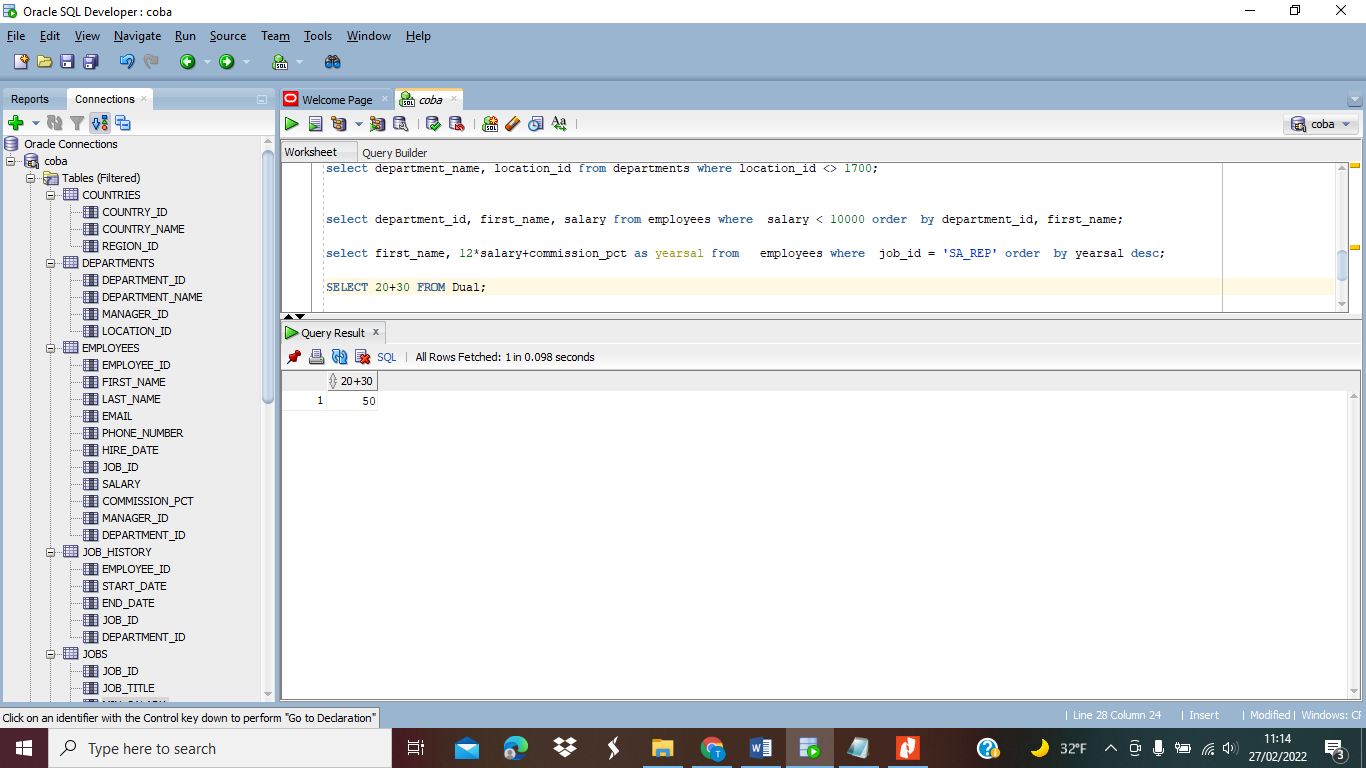
The DUAL Table

Lebih masuk akal untuk mengeksekusi kueri seperti yang ditunjukkan pada Listing 4-7 terhadap tabel dummy, dengan hanya satu baris dan satu kolom. Anda dapat membuat tabel seperti itu sendiri, tetapi Oracle DBMS menyediakan tabel dummy standar untuk tujuan ini, bernama DUAL, yang disimpan dalam kamus data. Karena Oracle DBMS mengetahui bahwa tabel DUAL hanya berisi satu baris, Anda biasanya mendapatkan hasil kinerja yang lebih baik dengan menggunakan tabel DUAL daripada tabel dummy yang Anda buat sendiri.

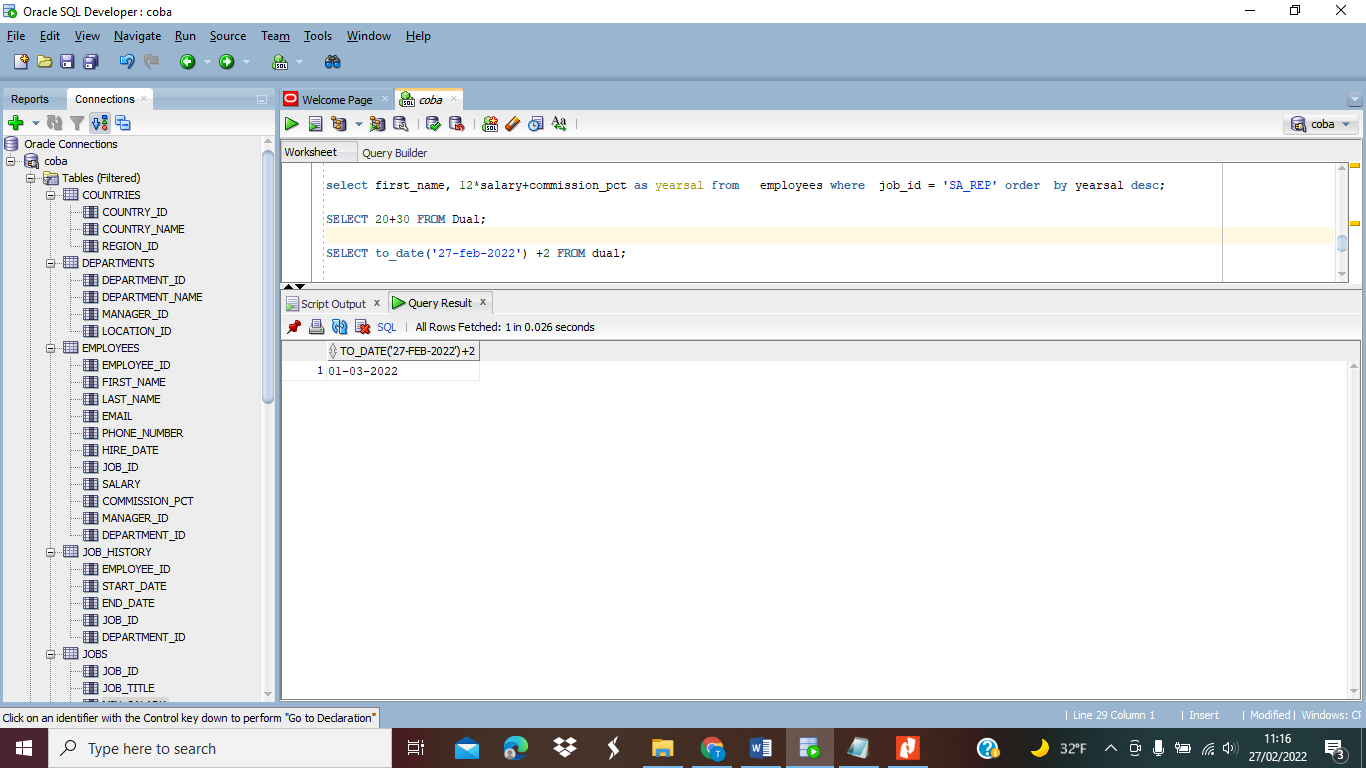
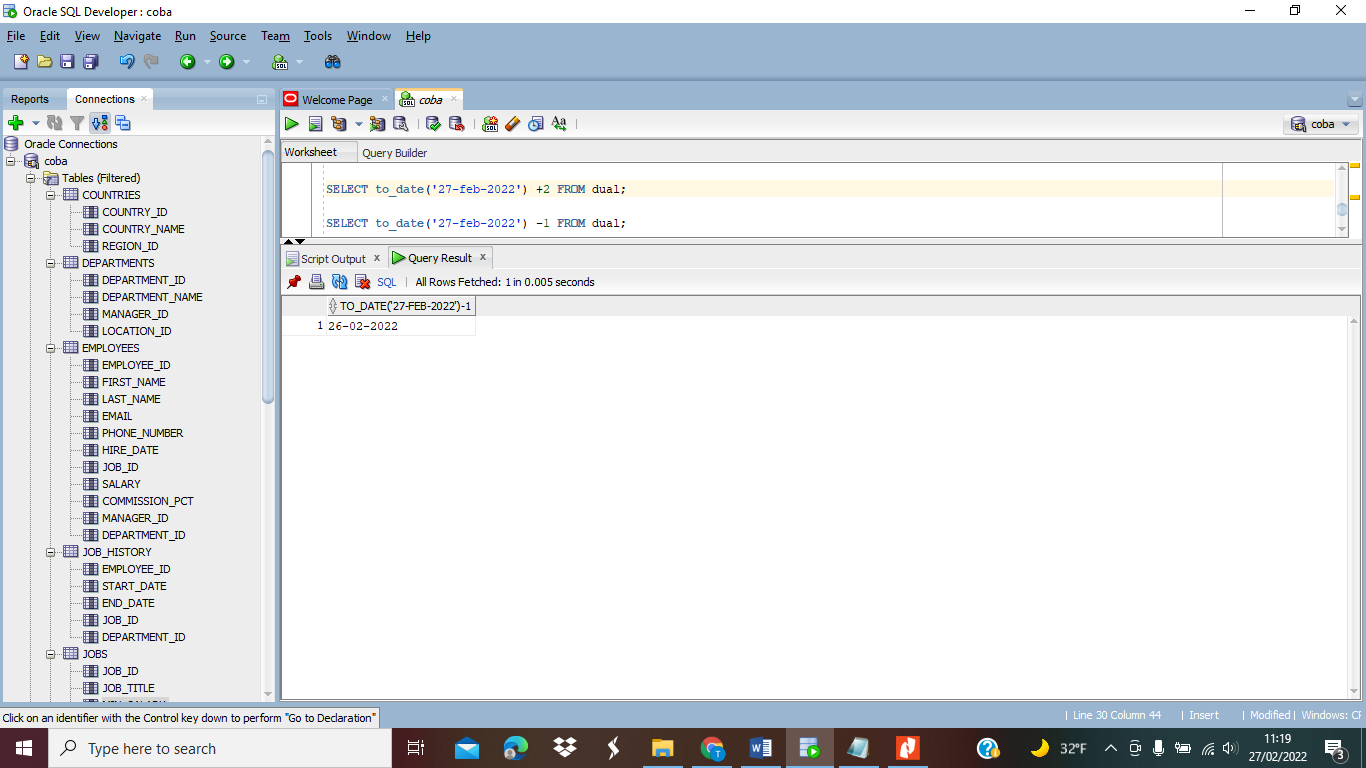
 

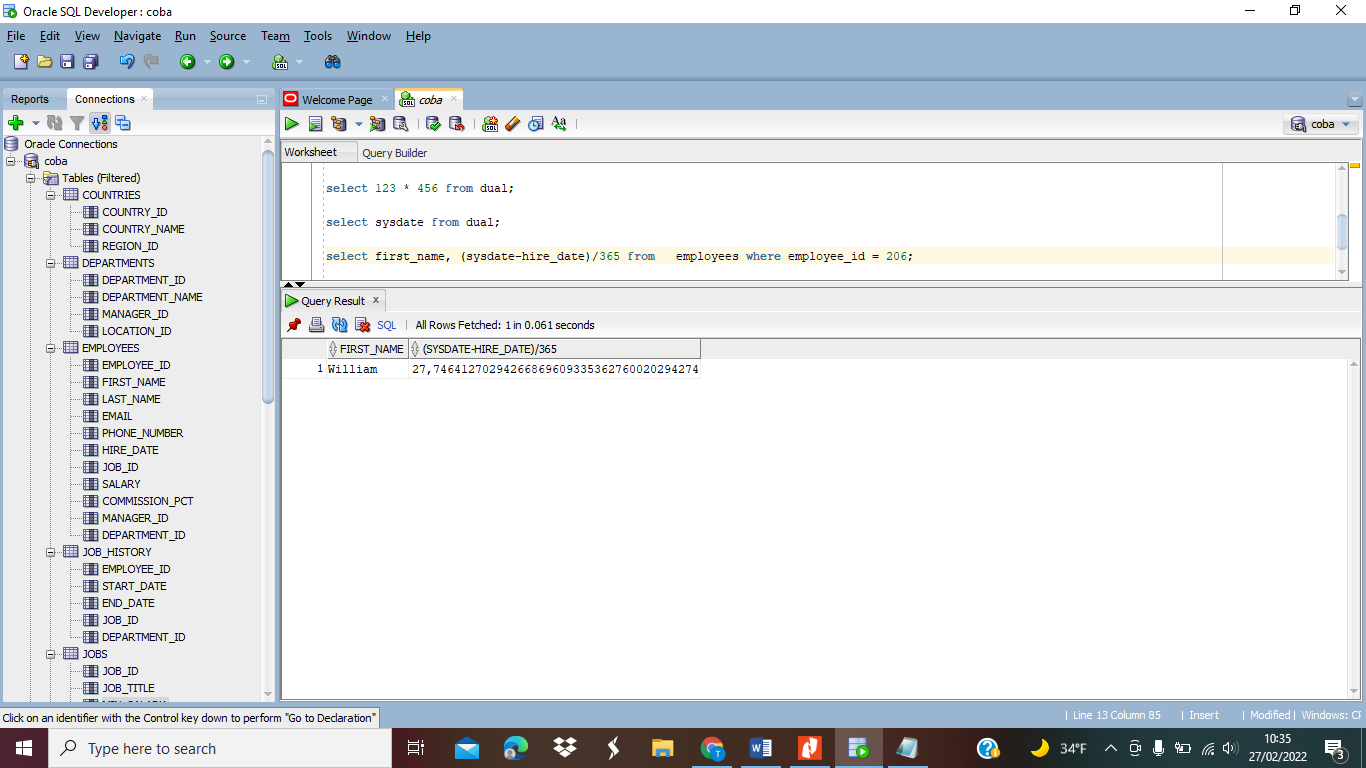
Kueri kedua menunjukkan contoh penggunaan tanggal sistem. Anda dapat merujuk ke tanggal sistem di Oracle dengan kata kunci SYSDATE. Sebenarnya, lebih tepatnya, SYSDATE adalah fungsi yang mengembalikan tanggal sistem. Fungsi-fungsi ini juga disebut sebagai kolom semu. contoh penggunaan SYSDATE untuk mendapatkan lama bekerja seorang karyawan, berdasarkan awal bekerja yang disimpan dalam kolom hire\_date pada tabel EMPLOYEES.

**Ekpresi Aritmetika dengan Operand**

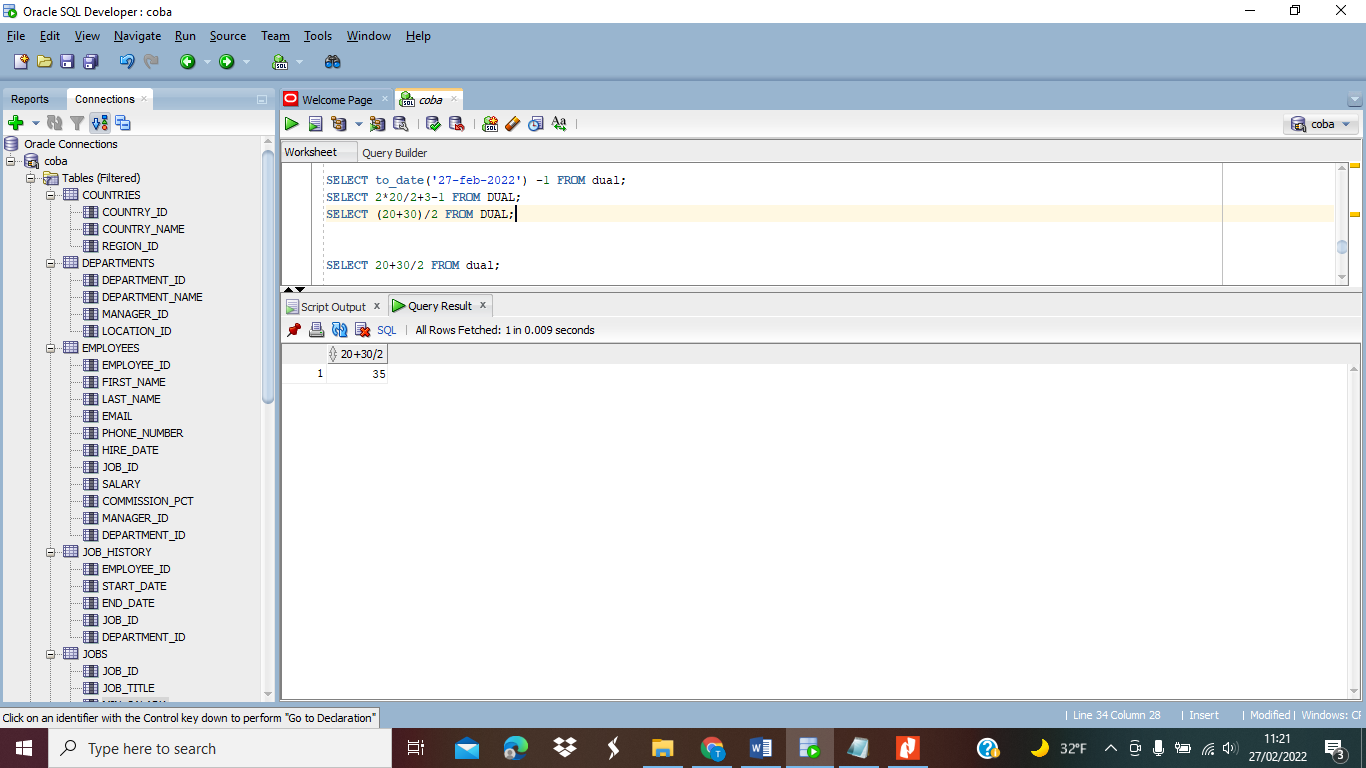
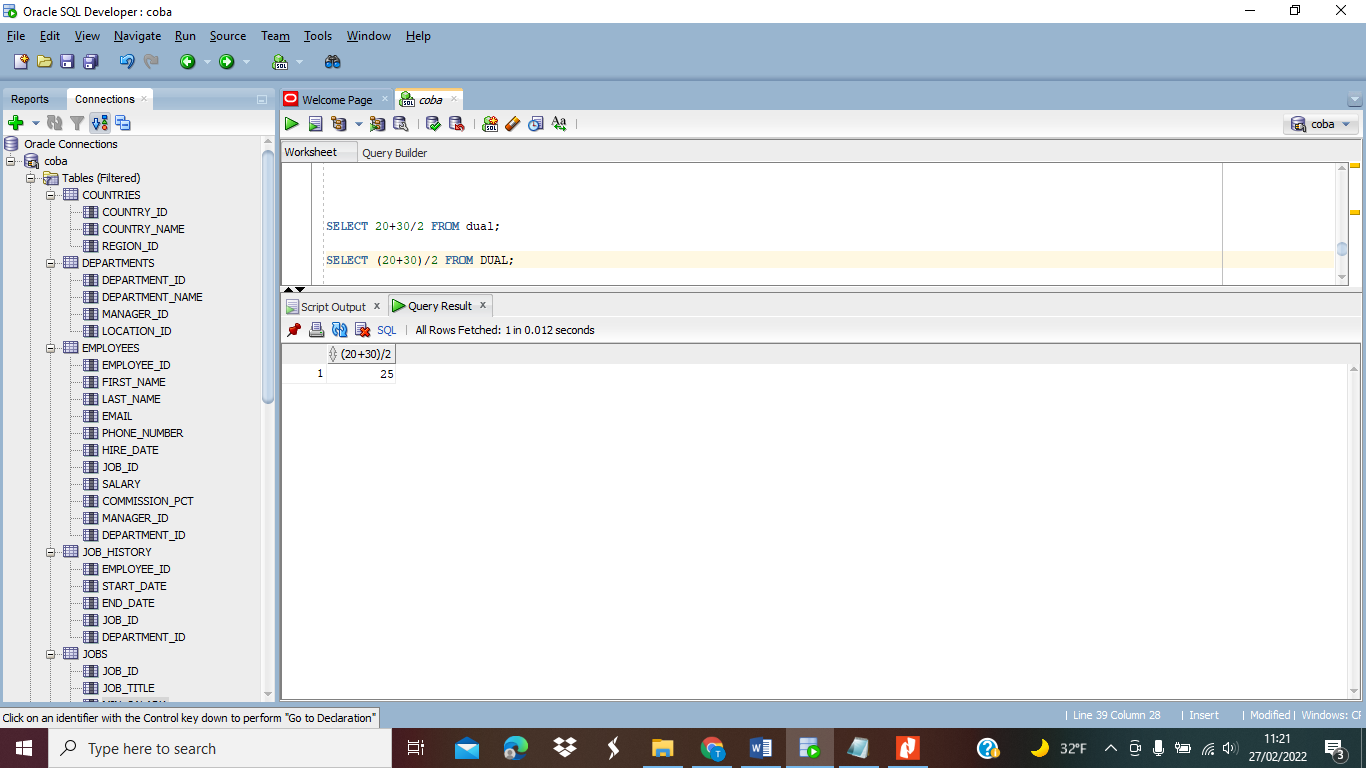
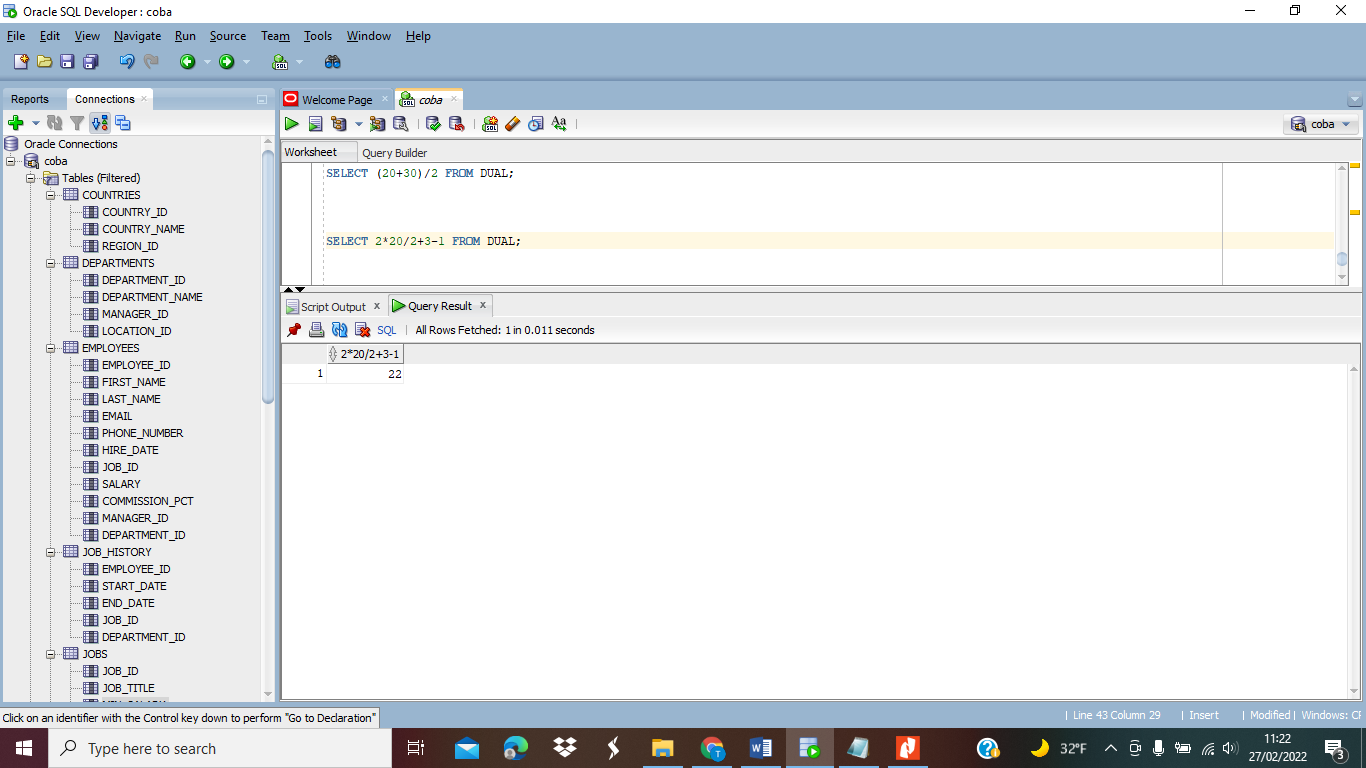


**Operari Aritmatika dengan Data Tanggal**



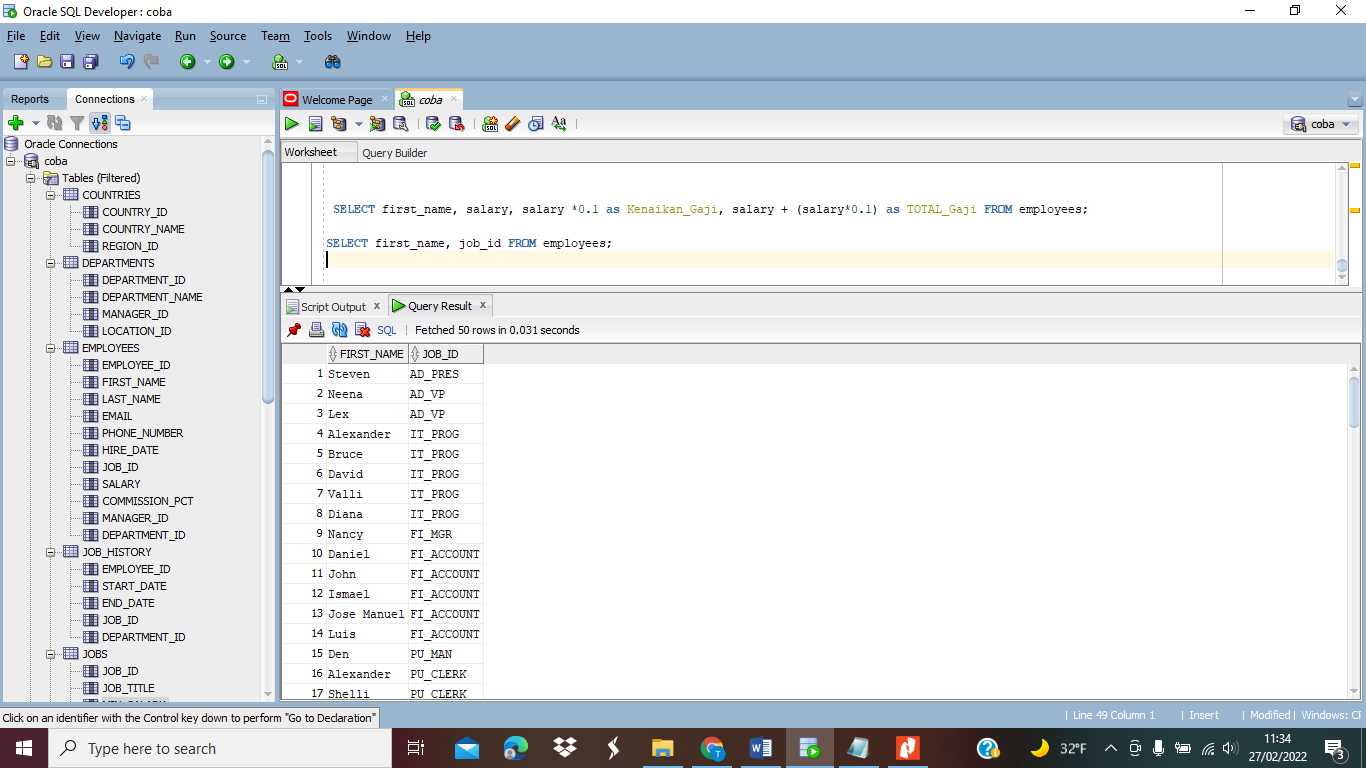
**Urutan Pengerjaan Operator**

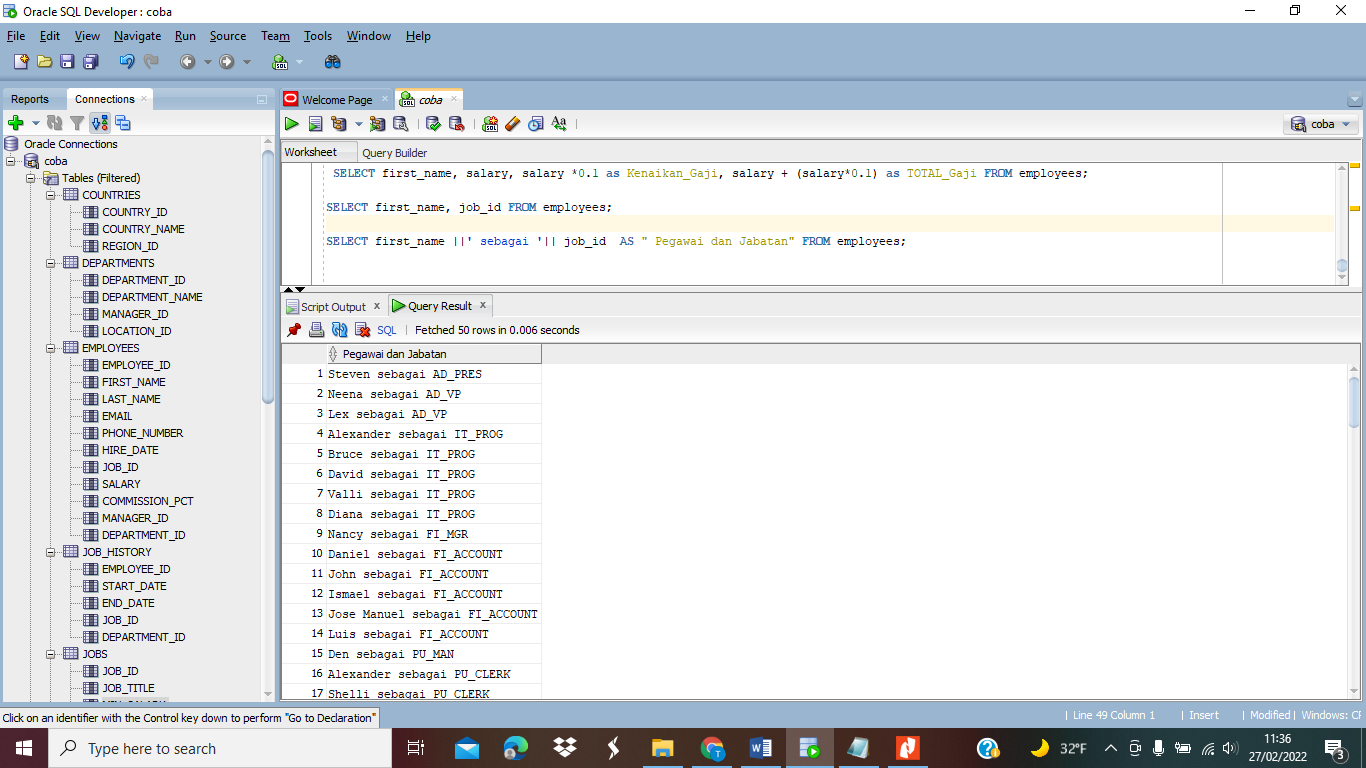
**Concatination**

Digunakan untuk mengkombinasikan beberapa kolom atau kolom dengan suatu text menjadi satu kolom output.

Contoh tanpa Concatination :



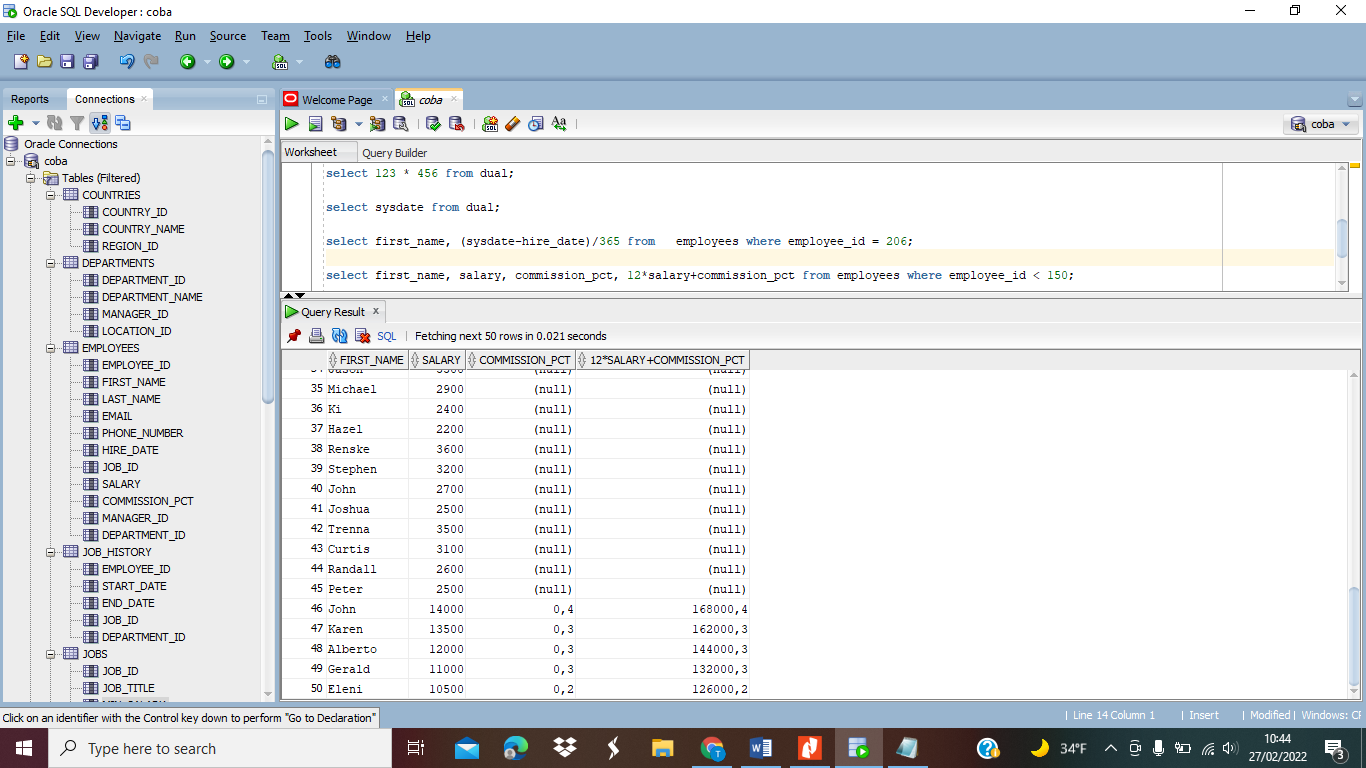
Menggunakan Concatination



Null Values in Expressions

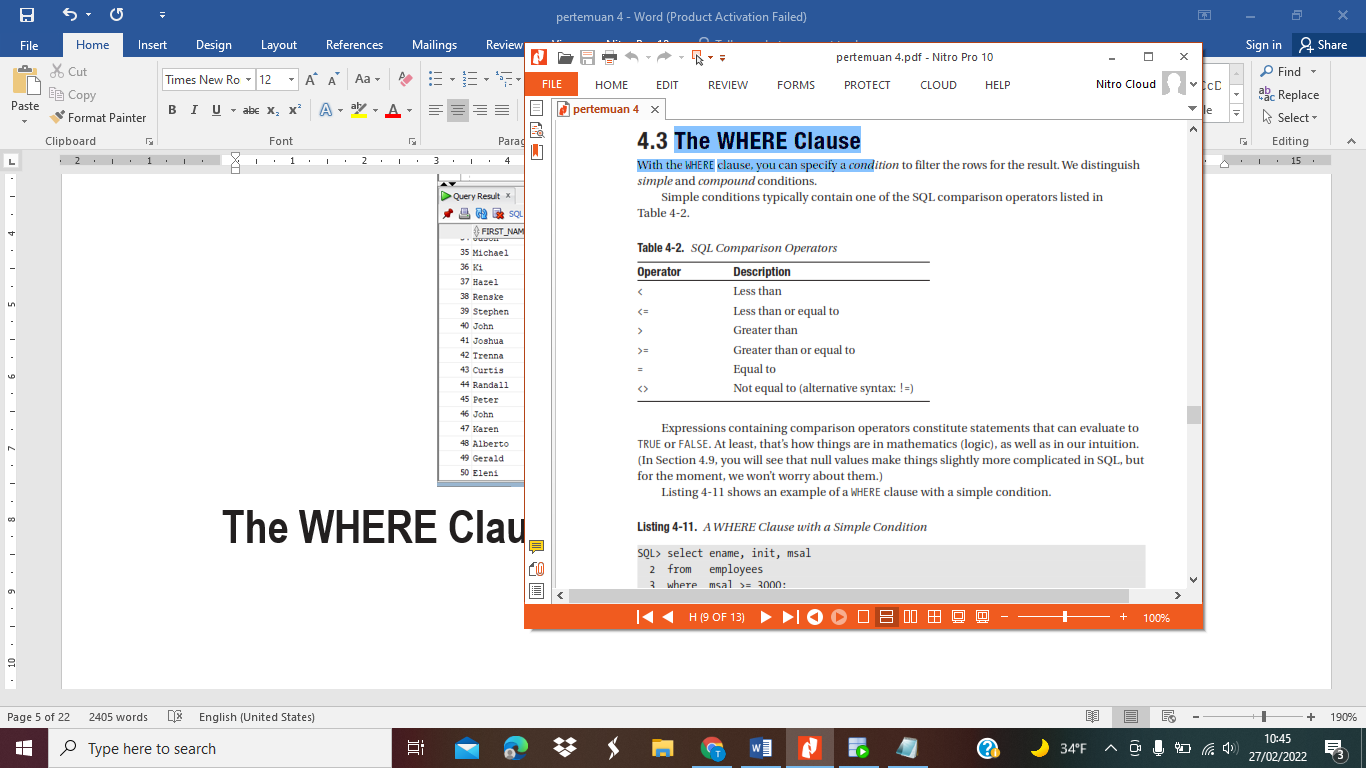
Kolom yang belum ada nilainya di dalam database.

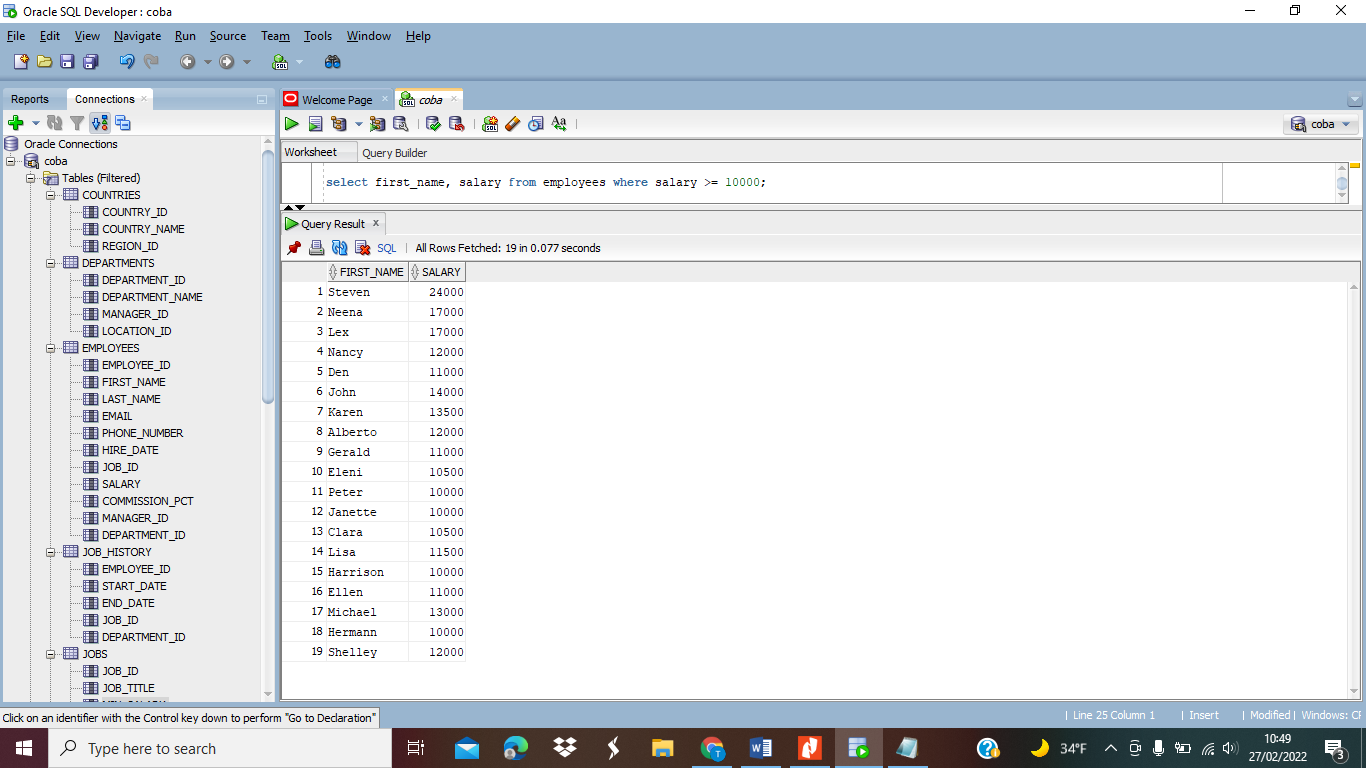
Anda harus selalu mempertimbangkan kemungkinan nilai nol muncul dalam ekspresi. Jika satu atau lebih variabel dalam ekspresi dievaluasi ke nilai nol, hasil ekspresi secara keseluruhan menjadi tidak diketahui.

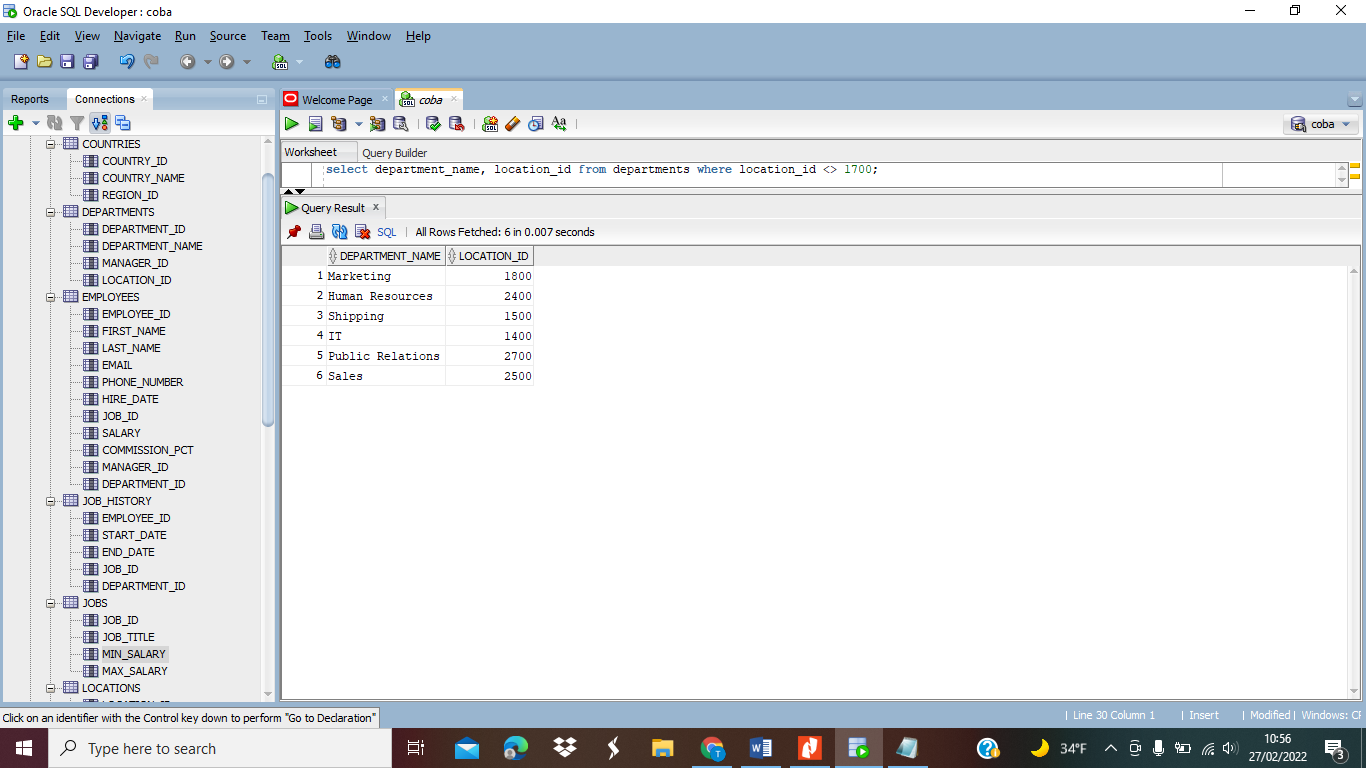


**The WHERE Clause**

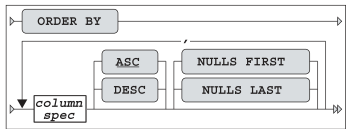
Dengan klausa WHERE, Anda dapat menentukan kondisi untuk memfilter baris hasil. Kami membedakan kondisi sederhana dan majemuk. Kondisi sederhana biasanya berisi salah satu operator perbandingan SQL. Ekspresi yang mengandung operator perbandingan merupakan pernyataan yang dapat mengevaluasi ke TRUE atau FALSE. Setidaknya, begitulah dalam matematika (logika), serta dalam intuisi kita.

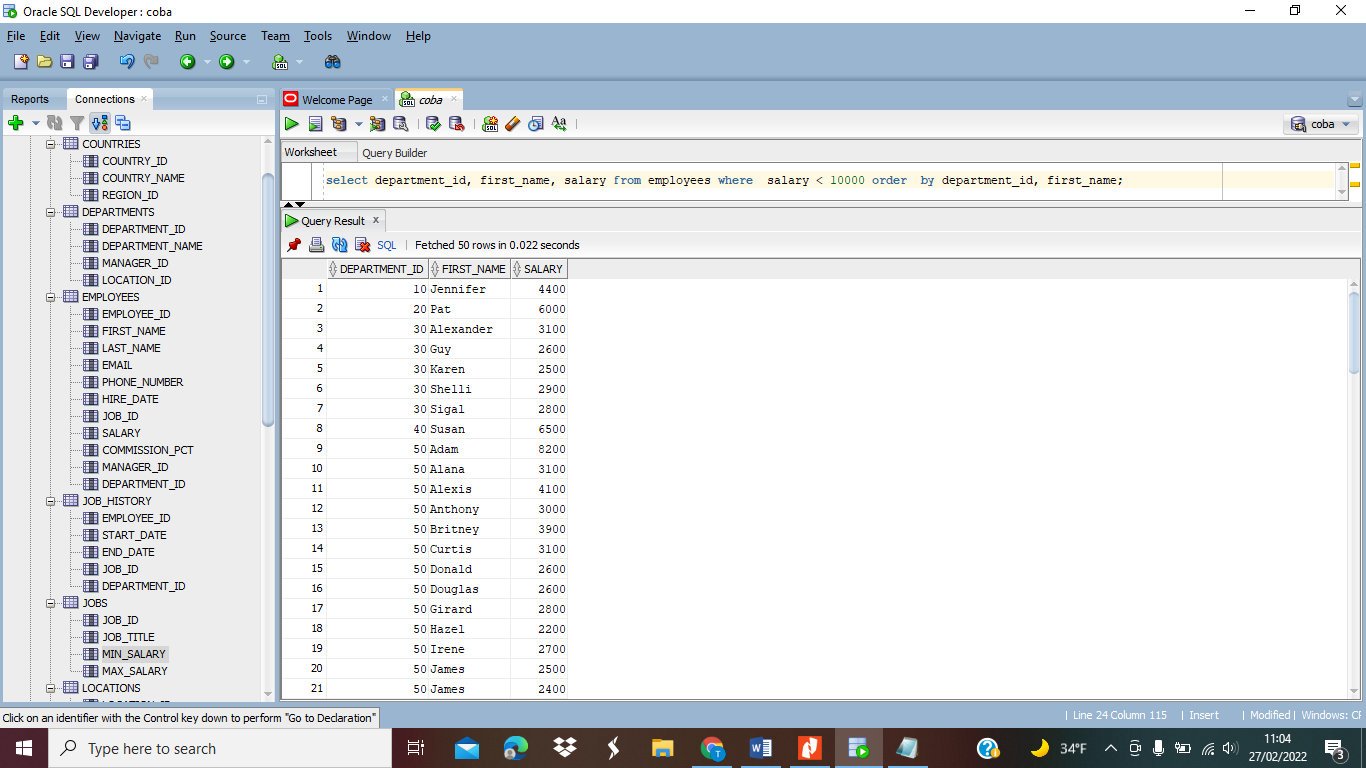


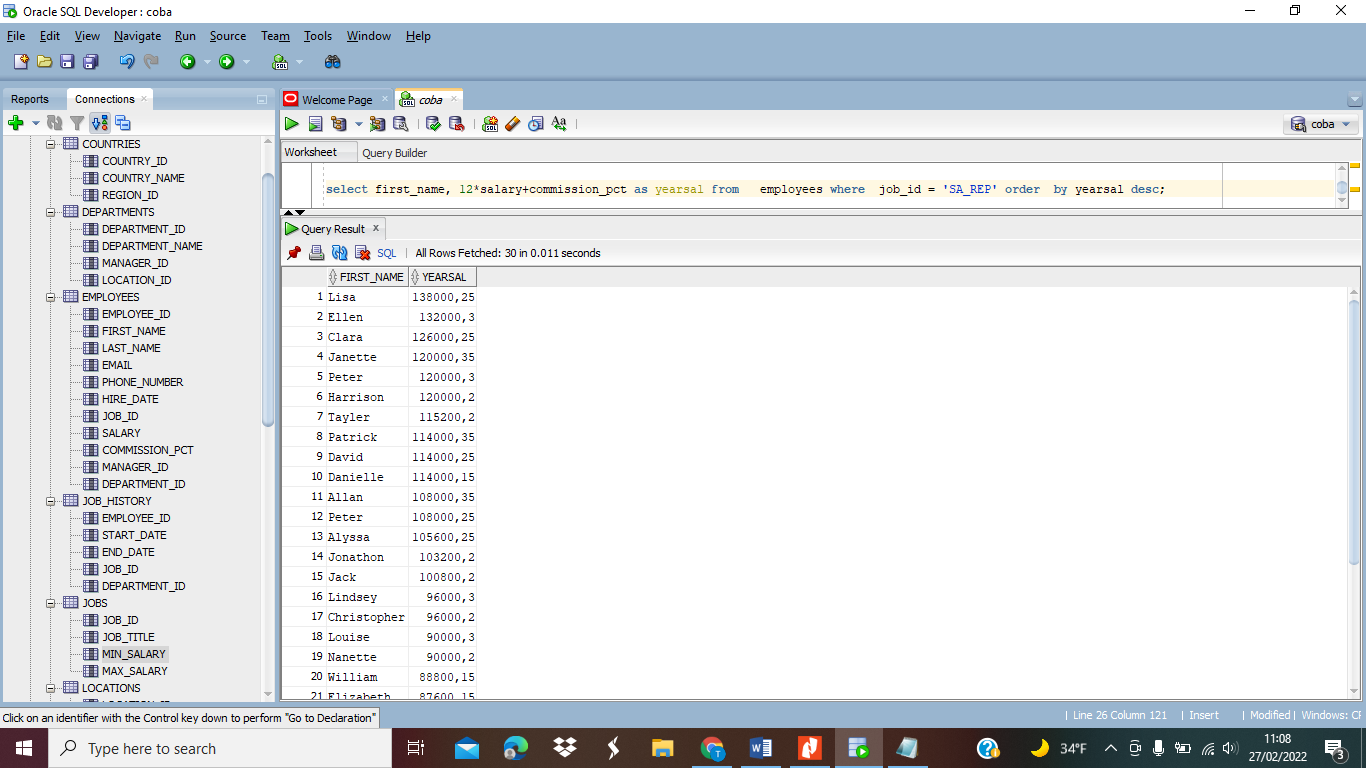




**The ORDER BY Clause**







TUGAS

