Jawaban Quiz SQL: Rahmat-Agung-Hadiwardoyo

1. Buat query untuk menampilkan employee yang memiliki gaji tertinggi dan ter-rendah. Gunakan union, min, max. (name masih berdasarkan first\_name, belum digabung dengan last\_name)

**Query:**

**select** employee\_id,**concat**(first\_name, ' ', last\_name) **AS** full\_name,salary

**from** employees

**where** salary = (**select** **MIN**(salary) **from** employees )

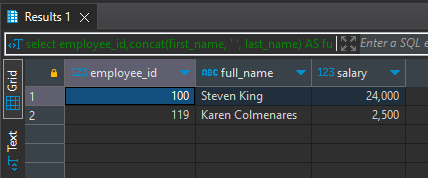
**union**

**select** employee\_id, **concat**(first\_name, ' ', last\_name) **AS** full\_name,salary

**from** employees

**where** salary = (**select** **MAX**(salary) **from** employees )

**order** **by** salary **desc**

**Hasil:**

1. Buat query untuk menampilkan department yang memiliki gaji tertinggi dan ter-rendah. Gunakan union, min, max.

**Query:**

**select** a.department\_id,b.department\_name, a.salary

**from** employees a **join** departments b

**on** a.department\_id = b.department\_id

**where** salary = (**select** **MIN**(salary) **from** employees )

**union**

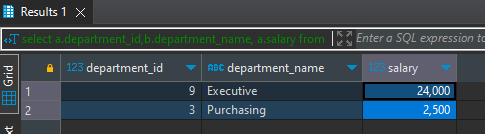
**select** a.department\_id,b.department\_name, a.salary

**from** employees a **join** departments b

**on** a.department\_id = b.department\_id

**where** salary = (**select** **MAX**(salary) **from** employees )

**order** **by** salary **desc**

**Hasil:**

1. Buat query untuk menampilkan jumlah employee berdasarkan job role nya.

**Query:**

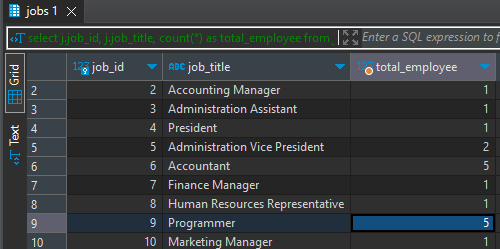
**select** j.job\_id, j.job\_title, **count**(\*) **as** total\_employee

**from** jobs j **join** employees e

**on** j.job\_id = e.job\_id

**group** **by** j.job\_id

**order** **by** j.job\_id

**Hasil:**

1. Buat query untuk menampilkan Total employee tiap department dan jobs yang totalnya melebihi dari 3 orang.

**Query:**

**select** d.department\_id, d.department\_name, j.job\_id, j.job\_title, **count**(e.employee\_id) **as** total\_employee

**from** employees e **join** departments d

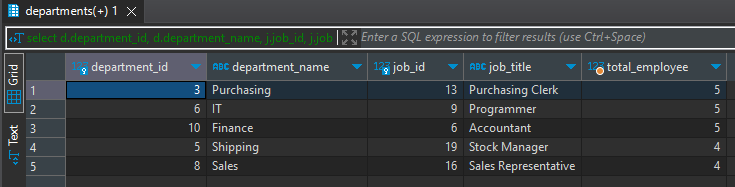
**on** d.department\_id = e.department\_id **join** jobs j

**on** e.job\_id = j.job\_id

**group** **by** d.department\_id, j.job\_id

**having** **count**(e.employee\_id) >3

**order** **by** total\_employee **desc**

**Hasil:**

1. Buat query untuk menampilkan employee yang memiliki masa kerja <= 25 tahun. Gunakan function age & extract.

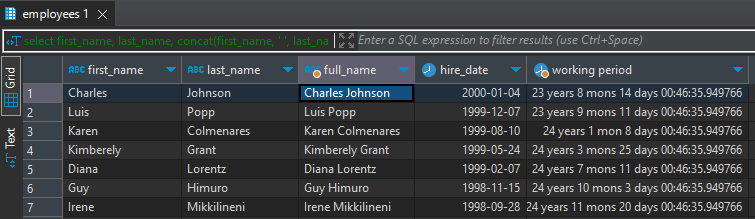
**Query:**

**select** first\_name, last\_name, **concat**(first\_name, ' ', last\_name) **as** full\_name,hire\_date , **age**(**now**(), hire\_date ) **as** "working period"

**from** employees

**where** **age**(**now**(), hire\_date ) <= '25 years'

**order** **by** hire\_date **desc**

**Hasil:**

1. Buat script update untuk menaikan gaji pegawai yang masa kerjanya <=25 tahun, naikan gaji sekarang + 100, dan tampilkan.

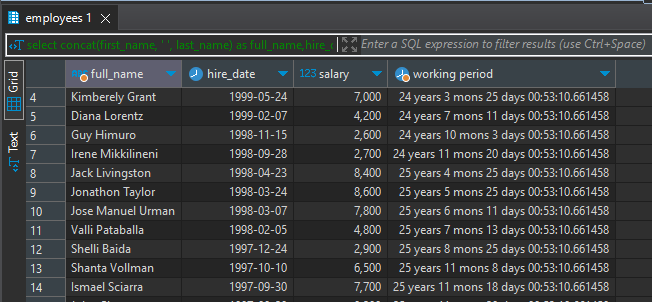
**Query:**

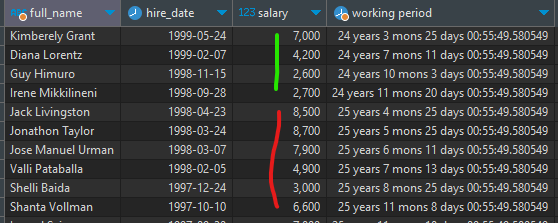
**update** employees **set** salary = salary + 100 **where** **age**(**now**(),hire\_date) >= '25 years'

**select** **concat**(first\_name, ' ', last\_name) **as** full\_name,hire\_date , salary , **age**(**now**(), hire\_date ) **as** "working period"

**from** employees

**order** **by** **age**(**now**(), hire\_date )

**Hasil:**



1. Buat script update untuk memindahkan department yang jumlah pegawai nya 1 orang, dipindahkan ke department Sales, kecuali department HRD.

**Query:**

**update** employees e

**set** department\_id = ( **select** department\_id **from** departments **where** department\_name ='Sales')

**where** e.department\_id **IN** (

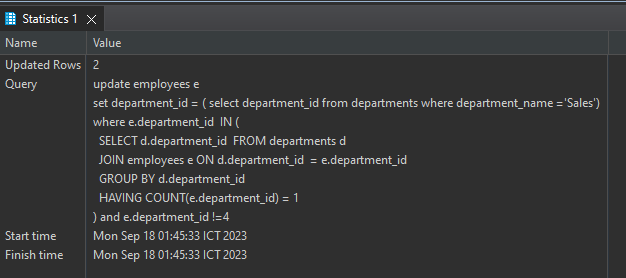
**SELECT** d.department\_id **FROM** departments d

**JOIN** employees e **ON** d.department\_id = e.department\_id

**GROUP** **BY** d.department\_id

**HAVING** **COUNT**(e.department\_id) = 1

) **and** e.department\_id !=4

**Hasil:**

