**From the following table, write a SQL query to find those employees who get a higher salary than the employee whose ID is 163. Return first name, last name.**

***Sample table*: employees**

postgres=# select first\_name , last\_name from employees where salary > (select salary from employees where employee\_id =163);

first\_name | last\_name

------------+-----------

Nancy | Greenberg

Den | Raphaely

John | Russell

Karen | Partners

Alberto | Errazuriz

Gerald | Cambrault

Peter | Tucker

Janette | King

Clara | Vishney

Lisa | Ozer

Ellen | Abel

Steven | King

Eleni | Zlotkey

Michael | Hartstein

Hermann | Baer

Neena | Kochhar

Lex | De Haan

Shelley | Higgins

Harrison | Bloom

Tayler | Fox

(20 rows)

**From the following table, write a SQL query to find those employees whose designation is the same as the employee whose ID is 169. Return first name, last name, department ID and job ID.**

postgres=# select first\_name,last\_name,department\_id,job\_id from employees where department\_id =(select department\_id from employees where employee\_id =169 );

first\_name | last\_name | department\_id | job\_id

-------------+------------+---------------+--------

John | Russell | 80 | SA\_MAN

Charles | Johnson | 80 | SA\_REP

Karen | Partners | 80 | SA\_MAN

Alberto | Errazuriz | 80 | SA\_MAN

Gerald | Cambrault | 80 | SA\_MAN

Peter | Tucker | 80 | SA\_REP

David | Bernstein | 80 | SA\_REP

Peter | Hall | 80 | SA\_REP

Oliver | Tuvault | 80 | SA\_REP

Janette | King | 80 | SA\_REP

Patrick | Sully | 80 | SA\_REP

Allan | McEwen | 80 | SA\_REP

Lindsey | Smith | 80 | SA\_REP

Louise | Doran | 80 | SA\_REP

Sarath | Sewall | 80 | SA\_REP

Clara | Vishney | 80 | SA\_REP

Danielle | Greene | 80 | SA\_REP

Mattea | Marvins | 80 | SA\_REP

David | Lee | 80 | SA\_REP

Sundar | Ande | 80 | SA\_REP

Amit | Banda | 80 | SA\_REP

Lisa | Ozer | 80 | SA\_REP

William | Smith | 80 | SA\_REP

Elizabeth | Bates | 80 | SA\_REP

Sundita | Kumar | 80 | SA\_REP

Ellen | Abel | 80 | SA\_REP

Alyssa | Hutton | 80 | SA\_REP

Eleni | Zlotkey | 80 | SA\_MAN

Christopher | Olsen | 80 | SA\_REP

Nanette | Cambrault | 80 | SA\_REP

Harrison | Bloom | 80 | SA\_REP

Tayler | Fox | 80 | SA\_REP

Jonathon | Taylor | 80 | SA\_REP

Jack | Livingston | 80 | SA\_REP

(34 rows)

**From the following table, write a SQL query to find those employees whose salary matches the smallest salary of any of the departments. Return first name, last name and department ID.**

postgres=# select first\_name,last\_name,department\_id from employees where salary = (select min(salary) from employees);

first\_name | last\_name | department\_id

------------+-----------+---------------

TJ | Olson | 50

(1 row)

**From the following table, write a SQL query to find those employees who earn more than the average salary. Return employee ID, first name, last name.**

postgres=# select employee\_id, first\_name,last\_name from employees where salary >(select avg(salary) from employees);

employee\_id | first\_name | last\_name

-------------+-------------+------------

103 | Alexander | Hunold

108 | Nancy | Greenberg

109 | Daniel | Faviet

110 | John | Chen

111 | Ismael | Sciarra

112 | Jose Manuel | Urman

113 | Luis | Popp

114 | Den | Raphaely

120 | Matthew | Weiss

121 | Adam | Fripp

122 | Payam | Kaufling

145 | John | Russell

178 | Kimberely | Grant

146 | Karen | Partners

147 | Alberto | Errazuriz

148 | Gerald | Cambrault

150 | Peter | Tucker

151 | David | Bernstein

152 | Peter | Hall

155 | Oliver | Tuvault

156 | Janette | King

157 | Patrick | Sully

158 | Allan | McEwen

159 | Lindsey | Smith

160 | Louise | Doran

161 | Sarath | Sewall

162 | Clara | Vishney

163 | Danielle | Greene

164 | Mattea | Marvins

165 | David | Lee

168 | Lisa | Ozer

171 | William | Smith

172 | Elizabeth | Bates

174 | Ellen | Abel

175 | Alyssa | Hutton

100 | Steven | King

149 | Eleni | Zlotkey

153 | Christopher | Olsen

201 | Michael | Hartstein

204 | Hermann | Baer

105 | David | Austin

203 | Susan | Mavris

101 | Neena | Kochhar

102 | Lex | De Haan

205 | Shelley | Higgins

206 | William | Gietz

154 | Nanette | Cambrault

169 | Harrison | Bloom

170 | Tayler | Fox

176 | Jonathon | Taylor

177 | Jack | Livingston

(51 rows)

**From the following table, write a SQL query to find those employees who report that manager whose first name is ‘Payam’. Return first name, last name, employee ID and salary.**

postgres=# select first\_name,last\_name, employee\_id from employees where manager\_id = (select employee\_id from employees where first\_name='Payam');

first\_name | last\_name | employee\_id

------------+------------+-------------

Jason | Mallin | 133

Michael | Rogers | 134

Ki | Gee | 135

Hazel | Philtanker | 136

Kelly | Chung | 188

Jennifer | Dilly | 189

Timothy | Gates | 190

Randall | Perkins | 191

(8 rows)

**From the following tables, write a SQL query to find all those employees who work in the Finance department. Return department ID, name (first), job ID and department name.**

**postgres=# select e.department\_id , e.first\_name , e.job\_id , d.department\_name**

**postgres-# FROM employees e , departments d**

**postgres-# WHERE e.department\_id = d.department\_id**

**postgres-# AND d.department\_name = 'Finance';**

**department\_id | first\_name | job\_id | department\_name**

**---------------+-------------+------------+-----------------**

**100 | Nancy | FI\_MGR | Finance**

**100 | Daniel | FI\_ACCOUNT | Finance**

**100 | John | FI\_ACCOUNT | Finance**

**100 | Ismael | FI\_ACCOUNT | Finance**

**100 | Jose Manuel | FI\_ACCOUNT | Finance**

**100 | Luis | FI\_ACCOUNT | Finance**

**(6 rows)**