

Q.1 Display all information in the tables EMP and DEPT.

Select \* from employees;

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id	region
100	Steven	King	SKING	515.123.4567	1987-06-17	AD_PRES	24000.00	NULL	NULL	90	East
101	Neena	Kochhar	NKOCHHAR	515.123.4568	1989-09-21	AD_VP	17000.00	NULL	100	90	North
102	Lex	De Haan	LDEHAAN	515.123.4569	1993-01-13	AD_VP	17000.00	NULL	100	90	NULL
103	Alexander	Hunold	AHUNOLD	590.423.4567	1990-01-03	IT_PROG	9000.00	NULL	102	60	East
104	Bruce	Ernst	BERNST	590.423.4568	1991-05-21	IT_PROG	6000.00	NULL	103	60	NULL
105	David	Austin	DAUSTIN	590.423.4569	1997-06-25	IT_PROG	4800.00	NULL	103	60	NULL
106	Valli	Pataballa	VPATABAL	590.423.4560	1998-02-05	IT_PROG	4800.00	NULL	103	60	South
107	Diana	Lorentz	DLORENTZ	590.423.5567	1999-02-07	IT_PROG	4200.00	NULL	103	60	South
108	Nancy	Greenberg	NGREENBE	515.124.4569	1994-08-17	FI_MGR	12000.00	NULL	101	100	South
109	Daniel	Faviet	DFAVIET	515.124.4169	1994-08-16	FI_ACCOUNT	9000.00	NULL	108	100	NULL
110	John	Chen	JCHEN	515.124.4269	1997-09-28	FI_ACCOUNT	8200.00	NULL	108	100	NULL
111	Ismael	Sciarra	ISCIARRA	515.124.4369	1997-09-30	FI_ACCOUNT	7700.00	NULL	108	100	West
112	Jose Manuel	Urman	JMURMAN	515.124.4469	1998-03-07	FI_ACCOUNT	7800.00	NULL	108	100	NULL
113	Luis	Popp	LPOPP	515.124.4567	1999-12-07	FI_ACCOUNT	6900.00	NULL	108	100	NULL
114	Den	Raphaely	DRAPHEAL	515.127.4561	1994-12-07	PU_MAN	11000.00	NULL	100	30	West
115	Alexander	Khoo	AKHOO	515.127.4562	1995-05-18	PU_CLERK	3100.00	NULL	114	30	West
116	Shelli	Baida	SBAIDA	515.127.4563	1997-12-24	PU_CLERK	2900.00	NULL	114	30	NULL
117	Sigal	Tobias	STOBIAS	515.127.4564	1997-07-24	PU_CLERK	2800.00	NULL	114	30	West
118	Guy	Himuro	GHIMURO	515.127.4565	1998-11-15	PU_CLERK	2600.00	NULL	114	30	West
119	Karen	Colmenares	KCOLMENA	515.127.4566	1999-08-10	PU_CLERK	2500.00	NULL	114	30	East
120	Matthew	Weiss	MWEISS	650.123.1234	1996-07-18	ST_MAN	8000.00	NULL	100	50	North
121	Adam	Fripp	AFRIPP	650.123.2234	1997-04-10	ST_MAN	8200.00	NULL	100	50	NULL
122	Payam	Kauffing	PKAUFLIN	650.123.3234	1995-05-01	ST_MAN	7900.00	NULL	100	50	NULL
123	Shanta	Vollman	SVOLLMAN	650.123.4234	1997-10-10	ST_MAN	6500.00	NULL	100	50	South
124	Kevin	Mourgos	KMOURGOS	650.123.5234	1999-11-16	ST_MAN	5800.00	NULL	100	50	NULL
125	Julia	Nayer	JNAYER	650.124.1214	1997-07-16	ST_CLERK	3200.00	NULL	120	50	East
126	Irene	Mikilineni	IMIKILIN	650.124.1224	1998-09-28	ST_CLERK	2700.00	NULL	120	50	North
127	James	Landry	JLANDRY	650.124.1334	1999-01-14	ST_CLERK	2400.00	NULL	120	50	West
128	Steven	Markle	SMARKLE	650.124.1434	2000-03-08	ST_CLERK	2200.00	NULL	120	50	East
129	Laura	Bissot	LBISSOT	650.124.5234	1997-08-20	ST_CLERK	3300.00	NULL	121	50	East
130	Mozhe	Atkinson	MATKINSO	650.124.6234	1997-10-30	ST_CLERK	2800.00	NULL	121	50	NULL
131	James	Marlow	JAMFLOW	650.124.7234	1997-02-16	ST_CLERK	2500.00	NULL	121	50	South
132	TJ	Olson	TJOLSON	650.124.8234	1999-04-10	ST_CLERK	2100.00	NULL	121	50	East
133	Jason	Mallin	JMALLIN	650.127.1934	1996-06-14	ST_CLERK	3300.00	NULL	122	50	South
134	Michael	Rogers	MROGERS	650.127.1834	1998-08-26	ST_CLERK	2900.00	NULL	122	50	West
135	Ki	Gee	KGEE	650.127.1734	1999-12-12	ST_CLERK	2400.00	NULL	122	50	NULL

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id	region
149	Eleni	Zlotkey	EZLOTKEY	011.44.1344....	2000-01-29	SA_MAN	10500.00	0.20	100	80	South
150	Peter	Tucker	PTUCKER	011.44.1344....	1997-01-30	SA_REP	10000.00	0.30	145	80	East
151	David	Bernstein	DBERNSTE	011.44.1344....	1997-03-24	SA_REP	9500.00	0.25	145	80	NULL
152	Peter	Hall	PHALL	011.44.1344....	1997-08-20	SA_REP	9000.00	0.25	145	80	North
153	Christopher	Olsen	COLSEN	011.44.1344....	1998-03-30	SA_REP	8000.00	0.20	145	80	South
154	Nanette	Cambraut	NCAMBRAU	011.44.1344....	1998-12-09	SA_REP	7500.00	0.20	145	80	NULL
155	Oliver	Tuvault	OTUVAULT	011.44.1344....	1999-11-23	SA_REP	7000.00	0.15	145	80	NULL
156	Janette	King	JKING	011.44.1345....	1996-01-30	SA_REP	10000.00	0.35	146	80	East
157	Patrick	Sully	PSULLY	011.44.1345....	1996-03-04	SA_REP	9500.00	0.35	146	80	East
158	Allan	McEwen	AMCEWEN	011.44.1345....	1996-08-01	SA_REP	9000.00	0.35	146	80	West
159	Lindsey	Smith	LSMITH	011.44.1345....	1997-03-10	SA_REP	8000.00	0.30	146	80	East
160	Louise	Doran	LDORAN	011.44.1345....	1997-12-15	SA_REP	7500.00	0.30	146	80	NULL
161	Sarath	Sewall	SSEWALL	011.44.1345....	1998-11-03	SA_REP	7000.00	0.25	146	80	East
162	Clara	Vishney	CVISHNEY	011.44.1346....	1997-11-11	SA_REP	10500.00	0.25	147	80	South
163	Danielle	Greene	DGREENE	011.44.1346....	1999-03-19	SA_REP	9500.00	0.15	147	80	West
164	Mattea	Marvins	MMARVINS	011.44.1346....	2000-01-24	SA_REP	7200.00	0.10	147	80	NULL
165	David	Lee	DLEE	011.44.1346....	2000-02-23	SA_REP	6800.00	0.10	147	80	North
166	Sundar	Ande	SANDE	011.44.1346....	2000-03-24	SA_REP	6400.00	0.10	147	80	North
167	Amit	Banda	ABANDA	011.44.1346....	2000-04-21	SA_REP	6200.00	0.10	147	80	South
168	Lisa	Ozer	LOZER	011.44.1343....	1997-03-11	SA_REP	11500.00	0.25	148	80	NULL
169	Harrison	Bloom	HBLOOM	011.44.1343....	1998-03-23	SA_REP	10000.00	0.20	148	80	NULL
170	Taylor	Fox	TFOX	011.44.1343....	1998-01-24	SA_REP	9600.00	0.20	148	80	NULL
171	William	Smith	WSMITH	011.44.1343....	1999-02-23	SA_REP	7400.00	0.15	148	80	East
172	Elizabeth	Bates	EBATES	011.44.1343....	1999-03-24	SA_REP	7300.00	0.15	148	80	East
173	Sundita	Kumar	SKUMAR	011.44.1343....	2000-04-21	SA_REP	6100.00	0.10	148	80	South
174	Ellen	Abel	EABEL	011.44.1644....	1996-05-11	SA_REP	11000.00	0.30	149	80	South
175	Alyssa	Hutton	AHUTTON	011.44.1644....	1997-03-19	SA_REP	8800.00	0.25	149	80	East
176	Jonathon	Taylor	JTAYLOR	011.44.1644....	1998-03-24	SA_REP	8600.00	0.20	149	80	South
177	Jack	Livingston	JLIVINGS	011.44.1644....	1998-04-23	SA_REP	8400.00	0.20	149	80	West
178	Kimberely	Grant	KGRANT	011.44.1644....	1999-05-24	SA_REP	7000.00	0.15	149	NULL	North
179	Charles	Johnson	CJOHNSON	011.44.1644....	2000-01-04	SA_REP	6200.00	0.10	149	80	East
180	Winston	Taylor	WTAYLOR	650.507.9876	1998-01-24	SH_CLERK	3200.00	NULL	120	50	East
181	Jean	Fleaur	JFLEAUR	650.507.9877	1998-02-23	SH_CLERK	3100.00	NULL	120	50	NULL
182	Martha	Sullivan	MSULLIVA	650.507.9878	1999-06-21	SH_CLERK	2500.00	NULL	120	50	North
183	Girard	Geoni	GGEONI	650.507.9879	2000-02-03	SH_CLERK	2800.00	NULL	120	50	East
184	Nandita	Sarchand	NSARCHAN	650.509.1876	1996-01-27	SH_CLERK	4200.00	NULL	121	50	East

185	Alexis	Bull	ABULL	650.509.2876	1997-02-20	SH_CLERK	4100.00	NULL	121	50	East
186	Julia	Dellinger	JDELLING	650.509.3876	1998-06-24	SH_CLERK	3400.00	NULL	121	50	West
187	Anthony	Cabrio	ACABRIO	650.509.4876	1999-02-07	SH_CLERK	3000.00	NULL	121	50	North
188	Kelly	Chung	KCHUNG	650.505.1876	1997-06-14	SH_CLERK	3800.00	NULL	122	50	NULL
189	Jennifer	Dilly	JDILLY	650.505.2876	1997-08-13	SH_CLERK	3600.00	NULL	122	50	North
190	Timothy	Gates	TGATES	650.505.3876	1998-07-11	SH_CLERK	2900.00	NULL	122	50	NULL
191	Randall	Perkins	RPERKINS	650.505.4876	1999-12-19	SH_CLERK	2500.00	NULL	122	50	NULL
192	Sarah	Bell	SBELL	650.501.1876	1996-02-04	SH_CLERK	4000.00	NULL	123	50	South
193	Britney	Everett	BEVERETT	650.501.2876	1997-03-03	SH_CLERK	3900.00	NULL	123	50	East
194	Samuel	McCain	SMCCAIN	650.501.3876	1998-07-01	SH_CLERK	3200.00	NULL	123	50	NULL
195	Vance	Jones	VJONES	650.501.4876	1999-03-17	SH_CLERK	2800.00	NULL	123	50	South
196	Alana	Walsh	AWALSH	650.507.9811	1998-04-24	SH_CLERK	3100.00	NULL	124	50	NULL
197	Kevin	Feeney	KFEENEY	650.507.9822	1998-05-23	SH_CLERK	3000.00	NULL	124	50	South
198	Donald	OConnell	DOCONNEL	650.507.9833	1999-06-21	SH_CLERK	2600.00	NULL	124	50	East
199	Douglas	Grant	DGRANT	650.507.9844	2000-01-13	SH_CLERK	2600.00	NULL	124	50	South
200	Jennifer	Whalen	JWHALEN	515.123.4444	1987-09-17	AD_ASST	4400.00	NULL	101	10	NULL
201	Michael	Hartstein	MHARTSTE	515.123.5555	1996-02-17	MK_MAN	13000.00	NULL	100	20	NULL
202	Pat	Fay	PFAY	603.123.6666	1997-08-17	MK_REP	6000.00	NULL	201	20	West
203	Susan	Mavris	SMAVRIS	515.123.7777	1994-06-07	HR_REP	6500.00	NULL	101	40	South
204	Hermann	Baer	HBAER	515.123.8888	1994-06-07	PR_REP	10000.00	NULL	101	70	East
205	Shelley	Higgins	SHIGGINS	515.123.8080	1994-06-07	AC_MGR	12000.00	NULL	101	110	East
206	William	Gietz	WGIEZT	515.123.8181	1994-06-07	AC_ACCOUNT	8300.00	NULL	205	110	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Select \* from departments;

	department_id	department_name	manager_id	location_id
▶	10	Administration	200	1700
	20	Marketing	201	1800
	30	Purchasing	114	1700
	40	Human Resources	203	2400
	50	Shipping	121	1500
	60	IT	103	1400
	70	Public Relations	204	2700
	80	Sales	145	2500
	90	Executive	100	1700
	100	Finance	108	1700
	110	Accounting	205	1700
	120	Treasury	NULL	1700
	130	Corporate Tax	NULL	1700
	140	Control And Credit	NULL	1700
	150	Shareholder Servi...	NULL	1700
	160	Benefits	NULL	1700
	170	Manufacturing	NULL	1700
	180	Construction	NULL	1700
	190	Contracting	NULL	1700
	200	Operations	NULL	1700
	210	IT Support	NULL	1700
	220	NOC	NULL	1700
	230	IT Helpdesk	NULL	1700
	240	Government Sales	NULL	1700
	250	Retail Sales	NULL	1700
	260	Recruiting	NULL	1700
	270	Payroll	NULL	1700
*	NULL	NULL	NULL	NULL

2. Display only the hire date and employee name for each employee.

```
SELECT hire_date, concat(first_name,' ',last_name) as employee_name
FROM employees;
```

hire_date	employee_name	hire_date	employee_name	hire_date	employee_name
1987-06-17	Steven King	1997-10-26	Stephen Stiles		
1989-09-21	Neena Kochhar	1998-02-12	John Seo		
1993-01-13	Lex De Haan	1998-04-06	Joshua Patel		
1990-01-03	Alexander Hunold	1995-10-17	Trenna Rajs		
1991-05-21	Bruce Ernst	1997-01-29	Curtis Davies		
1997-06-25	David Austin	1998-03-15	Randall Matos		
1998-02-05	Valli Pataballa	1998-07-09	Peter Vargas		
1999-02-07	Diana Lorentz	1996-10-01	John Russell		
1994-08-17	Nancy Greenberg	1997-01-05	Karen Partners	1998-03-24	Jonathon Taylor
1994-08-16	Daniel Faviet	1997-03-10	Alberto Errazuriz	1998-04-23	Jack Livingston
1997-09-28	John Chen	1999-10-15	Gerald Cambrault	1999-05-24	Kimberely Grant
1997-09-30	Ismael Sciarra	2000-01-29	Eleni Zlotkey	2000-01-04	Charles Johnson
1998-03-07	Jose Manuel Urman	1997-01-30	Peter Tucker	1998-01-24	Winston Taylor
1999-12-07	Luis Popp	1997-03-24	David Bernstein	1998-02-23	Jean Fleaur
1994-12-07	Den Raphaely	1997-08-20	Peter Hall	1999-06-21	Martha Sullivan
1995-05-18	Alexander Khoo	1998-03-30	Christopher Olsen	2000-02-03	Girard Geoni
1997-12-24	Shelli Baida	1998-12-09	Nanette Cambrault	1996-01-27	Nandita Sarchand
1997-07-24	Sigal Tobias	1999-11-23	Oliver Tuvault	1997-02-20	Alexis Bull
1998-11-15	Guy Himuro	1996-01-30	Janette King	1998-06-24	Julia Dellinger
1999-08-10	Karen Colmenares	1996-03-04	Patrick Sully	1999-02-07	Anthony Cabrio
1996-07-18	Matthew Weiss	1996-08-01	Allan McEwen	1997-06-14	Kelly Chung
1997-04-10	Adam Fripp	1997-03-10	Lindsey Smith	1997-08-13	Jennifer Dilly
1995-05-01	Payam Kaufling	1997-12-15	Louise Doran	1998-07-11	Timothy Gates
1997-10-10	Shanta Vollman	1998-11-03	Sarath Sewall	1999-12-19	Randall Perkins
1999-11-16	Kevin Mourgos	1997-11-11	Clara Vishney	1996-02-04	Sarah Bell
1997-07-16	Julia Nayer	1999-03-19	Danielle Greene	1997-03-03	Britney Everett
1998-09-28	Irene Mikkilineni	2000-01-24	Mattea Marvins	1998-07-01	Samuel McCain
1999-01-14	James Landry	2000-02-23	David Lee	1999-03-17	Vance Jones
2000-03-08	Steven Markle	2000-03-24	Sundar Ande	1998-04-24	Alana Walsh
1997-08-20	Laura Bissot	2000-04-21	Amit Banda	1998-05-23	Kevin Feeney
1997-10-30	Mozhe Atkinson	1997-03-11	Lisa Ozer	1999-06-21	Donald OConnell
1997-02-16	James Marlow	1998-03-23	Harrison Bloom	2000-01-13	Douglas Grant
1999-04-10	TJ Olson	1998-01-24	Tayler Fox	1987-09-17	Jennifer Whalen
1996-06-14	Jason Mallin	1999-02-23	William Smith	1996-02-17	Michael Hartstein
1998-08-26	Michael Rogers	1999-03-24	Elizabeth Bates	1997-08-17	Pat Fay
1999-12-12	Ki Gee	2000-04-21	Sundita Kumar	1994-06-07	Susan Mavris
2000-02-06	Hazel Philtanker	1996-05-11	Ellen Abel	1994-06-07	Hermann Baer
1995-07-14	Renske Ladwig	1997-03-19	Alyssa Hutton	1994-06-07	Shelley Higgins
				1994-06-07	William Gietz

3. Display the ename concatenated with the job ID, separated by a comma and space, and name the column Employee and Title

```
SELECT CONCAT(first_name, ', ', job_id) AS "Employee and Title"
FROM employees;
```

Employee and Title	Employee and Title	Employee and Title	Employee and Title
Steven, AD_PRES	Mozhe, ST_CLERK	Louise, SA_REP	
Neena, AD_VP	James, ST_CLERK	Sarath, SA_REP	
Lex, AD_VP	TJ, ST_CLERK	Clara, SA_REP	
Alexander, IT_PROG	Jason, ST_CLERK	Danielle, SA_REP	
Bruce, IT_PROG	Michael, ST_CLERK	Mattea, SA_REP	
David, IT_PROG	Ki, ST_CLERK	David, SA_REP	
Valli, IT_PROG	Hazel, ST_CLERK	Sundar, SA_REP	
Diana, IT_PROG	Renske, ST_CLERK	Amit, SA_REP	
Nancy, FI_MGR	Stephen, ST_CLERK	Lisa, SA_REP	
Daniel, FI_ACCOUNT	John, ST_CLERK	Harrison, SA_REP	
John, FI_ACCOUNT	Joshua, ST_CLERK	Taylor, SA_REP	
Ismael, FI_ACCOUNT	Trenna, ST_CLERK	William, SA_REP	
Jose Manuel, FI_ACCOUNT	Curtis, ST_CLERK	Elizabeth, SA_REP	
Luis, FI_ACCOUNT	Randall, ST_CLERK	Sundita, SA_REP	Timothy, SH_CLERK
Den, PU_MAN	Peter, ST_CLERK	Ellen, SA_REP	Randall, SH_CLERK
Alexander, PU_CLERK	John, SA_MAN	Alyssa, SA_REP	Sarah, SH_CLERK
Shelli, PU_CLERK	Karen, SA_MAN	Jonathon, SA_REP	Britney, SH_CLERK
Sigal, PU_CLERK	Alberto, SA_MAN	Jack, SA_REP	Samuel, SH_CLERK
Guy, PU_CLERK	Gerald, SA_MAN	Kimberely, SA_REP	Vance, SH_CLERK
Karen, PU_CLERK	Eleni, SA_MAN	Charles, SA_REP	Alana, SH_CLERK
Matthew, ST_MAN	Peter, SA_REP	Winston, SH_CLERK	Kevin, SH_CLERK
Adam, ST_MAN	David, SA_REP	Jean, SH_CLERK	Donald, SH_CLERK
Payam, ST_MAN	Peter, SA_REP	Martha, SH_CLERK	Douglas, SH_CLERK
Shanta, ST_MAN	Christopher, SA_REP	Girard, SH_CLERK	Jennifer, AD_ASST
Kevin, ST_MAN	Nanette, SA_REP	Nandita, SH_CLERK	Michael, MK_MAN
Julia, ST_CLERK	Oliver, SA_REP	Alexis, SH_CLERK	Pat, MK_REP
Irene, ST_CLERK	Janette, SA_REP	Julia, SH_CLERK	Susan, HR_REP
James, ST_CLERK	Patrick, SA_REP	Anthony, SH_CLERK	Hermann, PR_REP
Steven, ST_CLERK	Allan, SA_REP	Kelly, SH_CLERK	Shelley, AC_MGR
Laura, ST_CLERK	Lindsey, SA_REP	Jennifer, SH_CLERK	William, AC_ACCOUNT
Mozhe, ST_CLERK			

4. Display the hire date, name and department number for all clerks.

```
SELECT hire_date, concat(first_name,' ',last_name) as employee_name,  
department_id  
FROM employees  
WHERE job_id = 'ST_CLERK' or 'PU_CLERK';
```

	hire_date	employee_name	department_id
►	1997-07-16	Julia Nayer	50
	1998-09-28	Irene Mikkilineni	50
	1999-01-14	James Landry	50
	2000-03-08	Steven Markle	50
	1997-08-20	Laura Bissot	50
	1997-10-30	Mozhe Atkinson	50
	1997-02-16	James Marlow	50
	1999-04-10	TJ Olson	50
	1996-06-14	Jason Mallin	50
	1998-08-26	Michael Rogers	50
	1999-12-12	Ki Gee	50
	2000-02-06	Hazel Philtanker	50
	1995-07-14	Renske Ladwig	50
	1997-10-26	Stephen Stiles	50
	1998-02-12	John Seo	50
	1998-04-06	Joshua Patel	50
	1995-10-17	Trenna Rajs	50
	1997-01-29	Curtis Davies	50
	1998-03-15	Randall Matos	50
	1998-07-09	Peter Vargas	50

5. Create a query to display all the data from the EMP table. Separate each column by a comma. Name the column THE OUTPUT

```
SELECT CONCAT(employee_id, ',', first_name, ',', last_name, ',', email, ',',  
              phone_number, ',', hire_date, ',', job_id, ',', salary, ',',  
              commission_pct, ',', manager_id, ',', department_id) AS "THE OUTPUT"  
FROM employees;
```

6. Display the names and salaries of all employees with a salary greater than 2000.

```
SELECT CONCAT(first_name, ',', last_name) AS employee_name, salary  
FROM employees  
WHERE salary > 2000;
```

first_name	last_name	salary
Steven	King	24000.00
Neena	Kochhar	17000.00
Lex	De Haan	17000.00
Alexander	Hunold	9000.00
Bruce	Ernst	6000.00
David	Austin	4800.00
Valli	Pataballa	4800.00
Diana	Lorentz	4200.00
Nancy	Greenberg	12000.00
Daniel	Faviet	9000.00
John	Chen	8200.00
Ismael	Sciarra	7700.00
Jose Manuel	Urman	7800.00
Luis	Popp	6900.00
Den	Raphaely	11000.00
Alexander	Khoo	3100.00
Shelli	Baida	2900.00
Sigal	Tobias	2800.00
Guy	Himuro	2600.00
Karen	Colmenares	2500.00
Matthew	Weiss	8000.00
Adam	Fripp	8200.00
Payam	Kaufling	7900.00
Shanta	Vollman	6500.00
Kevin	Mourgos	5800.00
Julia	Nayer	3200.00
Irene	Mikkilineni	2700.00
James	Landry	2400.00
Steven	Markle	2200.00
Laura	Bissot	3300.00
Mozhe	Atkinson	2800.00
James	Marlow	2500.00

TJ	Olson	2100.00
Jason	Mallin	3300.00
Michael	Rogers	2900.00
Ki	Gee	2400.00
Hazel	Philtanker	2200.00
Renske	Ladwig	3600.00
Stephen	Stiles	3200.00
John	Seo	2700.00
Joshua	Patel	2500.00
Trenna	Rajs	3500.00
Curtis	Davies	3100.00
Randall	Matos	2600.00
Peter	Vargas	2500.00
John	Russell	14000.00
Karen	Partners	13500.00
Alberto	Errazuriz	12000.00
Gerald	Cambrault	11000.00
Eleni	Zlotkey	10500.00
Peter	Tucker	10000.00
David	Bernstein	9500.00
Peter	Hall	9000.00
Christopher	Olsen	8000.00
Nanette	Cambrault	7500.00
Oliver	Tuvault	7000.00
Janette	King	10000.00
Patrick	Sully	9500.00
Allan	McEwen	9000.00
Lindsey	Smith	8000.00
Louise	Doran	7500.00
Sarath	Sewall	7000.00



Clara	Vishney	10500.00
Danielle	Greene	9500.00
Mattea	Marvins	7200.00
David	Lee	6800.00
Sundar	Ande	6400.00
Amit	Banda	6200.00
Lisa	Ozer	11500.00
Harrison	Bloom	10000.00
Tayler	Fox	9600.00
William	Smith	7400.00
Elizabeth	Bates	7300.00
Sundita	Kumar	6100.00
Ellen	Abel	11000.00
Alyssa	Hutton	8800.00
Jonathon	Taylor	8600.00
Jack	Livingston	8400.00
Kimberely	Grant	7000.00
Charles	Johnson	6200.00
Winston	Taylor	3200.00
Jean	Fleaur	3100.00
Martha	Sullivan	2500.00
Girard	Geoni	2800.00
Nandita	Sarchand	4200.00
Alexis	Bull	4100.00
Julia	Dellinger	3400.00
Anthony	Cabrio	3000.00
Kelly	Chung	3800.00
Jennifer	Dilly	3600.00
Timothy	Gates	2900.00

Randall	Perkins	2500.00
Sarah	Bell	4000.00
Britney	Everett	3900.00
Samuel	McCain	3200.00
Vance	Jones	2800.00
Alana	Walsh	3100.00
Kevin	Feeney	3000.00
Donald	OConnell	2600.00
Douglas	Grant	2600.00
Jennifer	Whalen	4400.00
Michael	Hartstein	13000.00
Pat	Fay	6000.00
Susan	Mavris	6500.00
Hermann	Baer	10000.00
Shelley	Higgins	12000.00
William	Gietz	8300.00

7. Display the names and dates of employees with the column headers "Name" and "Start Date"

```
SELECT CONCAT(first_name, ' ', last_name) AS "Name", hire_date AS "Start Date"
FROM employees;
```

Name	Start Date				
Steven King	1987-06-17				
Neena Kochhar	1989-09-21				
Lex De Haan	1993-01-13				
Alexander Hunold	1990-01-03				
Bruce Ernst	1991-05-21				
David Austin	1997-06-25				
Valli Pataballa	1998-02-05				
Diana Lorentz	1999-02-07				
Nancy Greenberg	1994-08-17				
Daniel Faviet	1994-08-16				
John Chen	1997-09-28				
Ismael Sciarra	1997-09-30				
Jose Manuel Urman	1998-03-07				
Luis Popp	1999-12-07				
Den Raphaely	1994-12-07				
Alexander Khoo	1995-05-18				
Shelli Baida	1997-12-24				
Sigal Tobias	1997-07-24				
Guy Himuro	1998-11-15				
Karen Colmenares	1999-08-10				
Matthew Weiss	1996-07-18				
Adam Fripp	1997-04-10				
Payam Kaufing	1995-05-01				
Shanta Vollman	1997-10-10				
Kevin Mourgos	1999-11-16				
Julia Nayer	1997-07-16				
Irene Mikkilineni	1998-09-28				
James Landry	1999-01-14				
Steven Markle	2000-03-08				
Laura Bissot	1997-08-20				
Mozhe Atkinson	1997-10-30				
James Marlow	1997-02-16				
TJ Olson	1999-04-10				
Jason Mallin	1996-06-14				
Michael Rogers	1998-08-26				
Ki Gee	1999-12-12				
Hazel Philtanker	2000-02-06				
Renske Ladwig	1995-07-14				
Stephen Stiles	1997-10-26				
John Seo	1998-02-12				
Joshua Patel	1998-04-06				
Trenna Rajs	1995-10-17				
Curtis Davies	1997-01-29				
Randall Matos	1998-03-15				
Peter Vargas	1998-07-09				
John Russell	1996-10-01				
Karen Partners	1997-01-05				
Alberto Errazuriz	1997-03-10				
Gerald Cambrault	1999-10-15				
Eleni Zlotkey	2000-01-29				
Peter Tucker	1997-01-30				
David Bernstein	1997-03-24				
Peter Hall	1997-08-20				
Christopher Olsen	1998-03-30				
Nanette Cambrault	1998-12-09				
Oliver Tuvault	1999-11-23				
Janette King	1996-01-30				
Patrick Sully	1996-03-04				
Allan McEwen	1996-08-01				
Lindsey Smith	1997-03-10				
Louise Doran	1997-12-15				
Sarath Sewall	1998-11-03				
Clara Vishney	1997-11-11				
Danielle Greene	1999-03-19				
Mattea Marvins	2000-01-24				
David Lee	2000-02-23				
Sundar Ande	2000-03-24				
Amit Banda	2000-04-21				
Lisa Ozer	1997-03-11				
Harrison Bloom	1998-03-23				
Tayler Fox	1998-01-24				
William Smith	1999-02-23				
Elizabeth Bates	1999-03-24				
Sundita Kumar	2000-04-21				
Ellen Abel	1996-05-11				
Alyssa Hutton	1997-03-19				
Jonathon Taylor	1998-03-24				
Jack Livingston	1998-04-23				
Kimberely Grant	1999-05-24				
Charles Johnson	2000-01-04				
Winston Taylor	1998-01-24				
Jean Fleaur	1998-02-23				
Martha Sullivan	1999-06-21				
Girard Geoni	2000-02-03				
Nandita Sarchand	1996-01-27				
Alexis Bull	1997-02-20				
Julia Dellinger	1998-06-24				
Anthony Cabrio	1999-02-07				
Kelly Chung	1997-06-14				
Jennifer Dilly	1997-08-13				
Timothy Gates	1998-07-11				
Randall Perkins	1999-12-19				



Sarah Bell	1996-02-04
Britney Everett	1997-03-03
Samuel McCain	1998-07-01
Vance Jones	1999-03-17
Alana W& Vance Jones	8-04-24
Kevin Feeney	1998-05-23
Donald OConnell	1999-06-21
Douglas Grant	2000-01-13
Jennifer Whalen	1987-09-17
Michael Hartstein	1996-02-17
Pat Fay	1997-08-17
Susan Mavris	1994-06-07
Hermann Baer	1994-06-07
Shelley Higgins	1994-06-07
William Gietz	1994-06-07

8. Display the names and hire dates of all employees in the order they were hired.

```
select concat(first_name, ' ', last_name) as name, hire_date as "hire date"
from employees
order by hire_date;
```

Steven King	1987-06-17	Britney Everett	1997-03-03	Harrison Bloom	1998-03-23
Jennifer Whalen	1987-09-17	Alberto Errazuriz	1997-03-10	Jonathon Taylor	1998-03-24
Neena Kochhar	1989-09-21	Lindsey Smith	1997-03-10	Christopher Olsen	1998-03-30
Alexander Hunold	1990-01-03	Lisa Ozer	1997-03-11	Joshua Patel	1998-04-06
Bruce Ernst	1991-05-21	Alyssa Hutton	1997-03-19	Jack Livingston	1998-04-23
Lex De Haan	1993-01-13	David Bernstein	1997-03-24	Alana Walsh	1998-04-24
Susan Mavris	1994-06-07	Adam Fripp	1997-04-10	Kevin Feeney	1998-05-23
Hermann Baer	1994-06-07	Kelly Chung	1997-06-14	Julia Dellinger	1998-06-24
Shelley Higgins	1994-06-07	David Austin	1997-06-25	Samuel McCain	1998-07-01
William Gietz	1994-06-07	Julia Nayer	1997-07-16	Peter Vargas	1998-07-09
Daniel Faviet	1994-08-16	Sigal Tobias	1997-07-24	Timothy Gates	1998-07-11
Nancy Greenberg	1994-08-17	Jennifer Dilly	1997-08-13	Michael Rogers	1998-08-26
Den Raphaely	1994-12-07	Pat Fay	1997-08-17	Irene Mikilineni	1998-09-28
Payam Kaufling	1995-05-01	Laura Bissot	1997-08-20	Sarath Sewall	1998-11-03
Alexander Khoo	1995-05-18	Peter Hall	1997-08-20	Guy Himuro	1998-11-15
Renske Ladwig	1995-07-14	John Chen	1997-09-28	Nanette Cambrault	1998-12-09
Trenna Rajs	1995-10-17	Ismael Sciarra	1997-09-30	James Landry	1999-01-14
Nandita Sarchand	1996-01-27	Shanta Vollman	1997-10-10	Diana Lorentz	1999-02-07
Janette King	1996-01-30	Stephen Stiles	1997-10-26	Anthony Cabrio	1999-02-07
Sarah Bell	1996-02-04	Mozhe Atkinson	1997-10-30	William Smith	1999-02-23
Michael Hartstein	1996-02-17	Clara Vishney	1997-11-11	Vance Jones	1999-03-17
Patrick Sully	1996-03-04	Louise Doran	1997-12-15	Danielle Greene	1999-03-19
Ellen Abel	1996-05-11	Shelli Baida	1997-12-24	Elizabeth Bates	1999-03-24
Jason Mallin	1996-06-14	Tayler Fox	1998-01-24	TJ Olson	1999-04-10
Matthew Weiss	1996-07-18	Winston Taylor	1998-01-24	Kimberely Grant	1999-05-24
Allan McEwen	1996-08-01	Valli Pataballa	1998-02-05	Martha Sullivan	1999-06-21
John Russell	1996-10-01	John Seo	1998-02-12	Donald OConnell	1999-06-21
Karen Partners	1997-01-05	Jean Fleaur	1998-02-23	Karen Colmenares	1999-08-10
Curtis Davies	1997-01-29	Jose Manuel Urman	1998-03-07	Gerald Cambrault	1999-10-15
Peter Tucker	1997-01-30	Randall Matos	1998-03-15	Kevin Mourqos	1999-11-16
James Marlow	1997-02-16				
Alexis Bull	1997-02-20				
Oliver Tuvault	1999-11-23				
Luis Popp	1999-12-07				
Ki Gee	1999-12-12				
Randall Perkins	1999-12-19				
Charles Johnson	2000-01-04				
Douglas Grant	2000-01-13				
Mattea Marvins	2000-01-24				
Eleni Zlotkey	2000-01-29				
Girard Geoni	2000-02-03				
Hazel Philtanker	2000-02-06				
David Lee	2000-02-23				
Steven Markle	2000-03-08				
Sundar Ande	2000-03-24				
Amit Banda	2000-04-21				
Sundita Kumar	2000-04-21				

9. Display the names and salaries of all employees in reverse salary order.

```
select concat(first_name, ' ', last_name) as name, salary
from employees
order by salary desc;
```

Steven King	24000.00	Matthew Weiss	8000.00	Kelly Chung	3800.00
Neena Kochhar	17000.00	Christopher Olsen	8000.00	Renske Ladwig	3600.00
Lex De Haan	17000.00	Lindsey Smith	8000.00	Jennifer Dilly	3600.00
John Russell	14000.00	Payam Kaufling	7900.00	Trenna Rajs	3500.00
Karen Partners	13500.00	Jose Manuel Urman	7800.00	Julia Dellinger	3400.00
Michael Hartstein	13000.00	Ismael Sciarra	7700.00	Laura Bissot	3300.00
Nancy Greenberg	12000.00	Nanette Cambrault	7500.00	Jason Mallin	3300.00
Alberto Errazuriz	12000.00	Louise Doran	7500.00	Julia Nayer	3200.00
Shelley Higgins	12000.00	William Smith	7400.00	Stephen Stiles	3200.00
Lisa Ozer	11500.00	Elizabeth Bates	7300.00	Winston Taylor	3200.00
Den Raphaely	11000.00	Mattea Marvins	7200.00	Samuel McCain	3200.00
Gerald Cambrault	11000.00	Oliver Tuvault	7000.00	Alexander Khoo	3100.00
Ellen Abel	11000.00	Sarath Sewall	7000.00	Curtis Davies	3100.00
Eleni Zlotkey	10500.00	Kimberely Grant	7000.00	Jean Fleaur	3100.00
Clara Vishney	10500.00	Luis Popp	6900.00	Alana Walsh	3100.00
Peter Tucker	10000.00	David Lee	6800.00	Anthony Cabrio	3000.00
Janette King	10000.00	Shanta Vollman	6500.00	Kevin Feeney	3000.00
Harrison Bloom	10000.00	Susan Mavris	6500.00	Shelli Baida	2900.00
Hermann Baer	10000.00	Sundar Aude	6400.00	Michael Rogers	2900.00
Taylor Fox	9600.00	Amit Banda	6200.00	Timothy Gates	2900.00
David Bernstein	9500.00	Charles Johnson	6200.00	Sigal Tobias	2800.00
Patrick Sully	9500.00	Sundita Kumar	6100.00	Mozhe Atkinson	2800.00
Danielle Greene	9500.00	Bruce Ernst	6000.00	Girard Geoni	2800.00
Alexander Hunold	9000.00	Pat Fay	6000.00	Vance Jones	2800.00
Daniel Faviet	9000.00	Kevin Mourgous	5800.00	Irene Mikkilineni	2700.00
Peter Hall	9000.00	David Austin	4800.00	John Seo	2700.00
Allan McEwen	9000.00	Valli Pataballa	4800.00	Guy Himuro	2600.00
Alyssa Hutton	8800.00	Jennifer Whalen	4400.00	Randall Matos	2600.00
Jonathon Taylor	8600.00	Diana Lorentz	4200.00	Donald OConnell	2600.00
Jack Livingston	8400.00	Nandita Sarchand	4200.00	Douglas Grant	2600.00
William Gietz	8300.00	Alexis Bull	4100.00	Karen Colmenares	2500.00
John Chen	8200.00	Sarah Bell	4000.00	James Marlow	2500.00
Adam Fripp	8200.00	Britney Everett	3900.00	Joshua Patel	2500.00

10. Display 'ename " and"deptno " who are all earned commission and display salary in reverse order.

```
select concat(first_name, ' and ', department_id) as "ename and deptno", salary
from employees
where commission_pct is not null
order by salary desc;
```

ename and deptno	salary
John and 80	14000.00
Karen and 80	13500.00
Alberto and 80	12000.00
Lisa and 80	11500.00
Gerald and 80	11000.00
Ellen and 80	11000.00
Eleni and 80	10500.00
Clara and 80	10500.00
Peter and 80	10000.00
Janette and 80	10000.00
Harrison and 80	10000.00
Taylor and 80	9600.00
David and 80	9500.00
Patrick and 80	9500.00
Danielle and 80	9500.00
Peter and 80	9000.00
Allan and 80	9000.00
Alyssa and 80	8800.00
Jonathon and 80	8600.00
Jack and 80	8400.00
Christopher and 80	8000.00
Lindsey and 80	8000.00
Nanette and 80	7500.00
Louise and 80	7500.00
William and 80	7400.00
Elizabeth and 80	7300.00
Mattea and 80	7200.00
Oliver and 80	7000.00
Sarath and 80	7000.00
NULL	7000.00
David and 80	6800.00
Sundar and 80	6400.00
Amit and 80	6200.00
Charles and 80	6200.00
Sundita and 80	6100.00

11. Display the last name and job title of all employees who do not have a manager

```
select last_name, job_id  
from employees  
where manager_id is null;
```

last_name	job_id
King	AD_PRES

12. Display the last name, job, and salary for all employees whose job is sales representative or stock clerk and whose salary is not equal to \$2,500, \$3,500, or \$5,000

last_name	job_id	salary			
Tucker	SA_REP	10000.00			
Bernstein	SA_REP	9500.00			
Hall	SA_REP	9000.00			
Olsen	SA_REP	8000.00			
Cambrault	SA_REP	7500.00			
Tuvault	SA_REP	7000.00			
King	SA_REP	10000.00			
Sully	SA_REP	9500.00			
McEwen	SA_REP	9000.00			
Smith	SA_REP	8000.00			
Doran	SA_REP	7500.00			
Sewall	SA_REP	7000.00			
Vishney	SA_REP	10500.00			
Greene	SA_REP	9500.00			
Marvins	SA_REP	7200.00			
Lee	SA_REP	6800.00			
Ande	SA_REP	6400.00			
Banda	SA_REP	6200.00			
Ozer	SA_REP	11500.00			
Bloom	SA_REP	10000.00			
Fox	SA_REP	9600.00			
Smith	SA_REP	7400.00			
Bates	SA_REP	7300.00			
Kumar	SA_REP	6100.00			
Abel	SA_REP	11000.00			
Hutton	SA_REP	8800.00			
Taylor	SA_REP	8600.00			
Livingston	SA_REP	8400.00			
Grant	SA_REP	7000.00	Dilly	SH_CL...	3600.00
Johnson	SA_REP	6200.00	Gates	SH_CL...	2900.00
Taylor	SH_CL...	3200.00	Bell	SH_CL...	4000.00
Fleaur	SH_CL...	3100.00	Everett	SH_CL...	3900.00
Geoni	SH_CL...	2800.00	McCain	SH_CL...	3200.00
Sarchand	SH_CL...	4200.00	Jones	SH_CL...	2800.00
Bull	SH_CL...	4100.00	Walsh	SH_CL...	3100.00
Dellinger	SH_CL...	3400.00	Feeney	SH_CL...	3000.00
Cabrio	SH_CL...	3000.00	OConnell	SH_CL...	2600.00
Chung	SH_CL...	3800.00	Grant	SH_CL...	2600.00



13. Display the maximum, minimum and average salary and commission earned.

```
select
  max(salary) as max_salary,
  min(salary) as min_salary,
  avg(salary) as avg_salary,
  max(commission_pct) as max_commission,
  min(commission_pct) as min_commission,
  avg(commission_pct) as avg_commission
from employees;
```

	max_salary	min_salary	avg_salary	max_commission	min_commission	avg_commission
▶	24000.00	2100.00	6461.682243	0.40	0.10	0.222857

14. Display the department number, total salary payout and total commission payout for each department.

```
select department_id,
       sum(salary) as total_salary_payout,
       sum(commission_pct) as total_commission_payout
from employees
group by department_id;
```

	department_id	total_salary_payout	total_commission_payout
▶	NULL	7000.00	0.15
	10	4400.00	NULL
	20	19000.00	NULL
	30	24900.00	NULL
	40	6500.00	NULL
	50	156400.00	NULL
	60	28800.00	NULL
	70	10000.00	NULL
	80	304500.00	7.65
	90	58000.00	NULL
	100	51600.00	NULL
	110	20300.00	NULL

15. Display the department number and number of employees in each department

```
select department_id, count(*) as num_employees
from employees
group by department_id;
```

	department_id	num_employees
▶	NULL	1
	10	1
	20	2
	30	6
	40	1
	50	45
	60	5
	70	1
	80	34
	90	3
	100	6
	110	2

16. Display the department number and total salary of employees in each department.

```
select department_id, sum(salary) as toaL_salary
from employees
group by department_id;
```

	department_id	toaL_salary
▶	NULL	7000.00
	10	4400.00
	20	19000.00
	30	24900.00
	40	6500.00
	50	156400.00
	60	28800.00
	70	10000.00
	80	304500.00
	90	58000.00
	100	51600.00
	110	20300.00

17. Display the employee's name who doesn't earn a commission. Order the result set without using the column name

```
select concat(first_name, ' ', last_name)
from employees
where commission_pct is null
order by 1;
```

concat(first_name, ' ', last_name)		
Adam Fripp	Kevin Feeney	
Alana Walsh	Kevin Mourgos	
Alexander Hunold	Ki Gee	
Alexander Khoo	Laura Bissot	
Alexis Bull	Lex De Haan	
Anthony Cabrio	Luis Popp	
Britney Everett	Martha Sullivan	
Bruce Ernst	Matthew Weiss	
Curtis Davies	Michael Hartstein	
Daniel Faviat	Michael Rogers	
David Austin	Mozhe Atkinson	
Den Raphaely	Nancy Greenberg	
Diana Lorentz	Nandita Sarchand	
Donald OConnell	Neena Kochhar	
Douglas Grant	Pat Fay	
Girard Geoni	Payam Kaufling	
Guy Himuro	Peter Vargas	
Hazel Philtanker	Randall Matos	
Hermann Baer	Randall Perkins	
Irene Mikilineni	Renske Ladwig	
Ismael Sciarra	Samuel McCain	
James Landry	Sarah Bell	
James Marlow	Shanta Vollman	
Jason Mallin	Shelley Higgins	Steven Markle
Jean Fleur	Shelli Baida	Susan Mavris
Jennifer Dilly	Sigal Tobias	Timothy Gates
Jennifer Whalen	Stephen Stiles	TJ Olson
John Chen	Steven King	Trenna Rajs
John Seo	Steven Markle	Valli Pataballa
Jose Manuel Urman	Susan Mavris	Vance Jones
Joshua Patel	Timothy Gates	William Gietz
Julia Dellinger	TJ Olson	Winston Taylor

18. Display the employees name, department id and commission. If an Employee doesn't earn the commission, then display as 'No commission'. Name the columns appropriately

```
select concat(first_name, ' ', last_name) as employee_name,
       salary as salary,
       coalesce(cast(commission_pct * 2 as char), 'No commission') as
double_commission
from employees;
```

	employee_name	department	commission			
►	Steven King	90	No commission	Ki Gee	50	No commission
	Neena Kochhar	90	No commission	Hazel Philtanker	50	No commission
	Lex De Haan	90	No commission	Renske Ladwig	50	No commission
	Alexander Hunold	60	No commission	Stephen Stiles	50	No commission
	Bruce Ernst	60	No commission	John Seo	50	No commission
	David Austin	60	No commission	Joshua Patel	50	No commission
	Valli Pataballa	60	No commission	Trenna Rajs	50	No commission
	Diana Lorentz	60	No commission	Curtis Davies	50	No commission
	Nancy Greenberg	100	No commission	Randall Matos	50	No commission
	Daniel Faviet	100	No commission	Peter Vargas	50	No commission
	John Chen	100	No commission	John Russell	80	0.40
	Ismael Sciarra	100	No commission	Karen Partners	80	0.30
	Jose Manuel Urman	100	No commission	Alberto Errazuriz	80	0.30
	Luis Popp	100	No commission	Gerald Cambault	80	0.30
	Den Raphaely	30	No commission	Eleni Zlotkey	80	0.20
	Alexander Khoo	30	No commission	Peter Tucker	80	0.30
	Shelli Baida	30	No commission	David Bernstein	80	0.25
	Sigal Tobias	30	No commission	Peter Hall	80	0.25
	Guy Himuro	30	No commission	Christopher Olsen	80	0.20
	Karen Colmenares	30	No commission	Nanette Cambault	80	0.20
	Matthew Weiss	50	No commission	Oliver Tuvault	80	0.15
	Adam Fripp	50	No commission	Janette King	80	0.35
	Payam Kaufling	50	No commission	Patrick Sully	80	0.35
	Shanta Vollman	50	No commission	Allan McEwen	80	0.35
	Kevin Mourgous	50	No commission	Lindsey Smith	80	0.30
	Julia Nayer	50	No commission	Louise Doran	80	0.30
	Irene Mikkilineni	50	No commission	Sarath Sewall	80	0.25
	James Landry	50	No commission	Clara Vishney	80	0.25
	Steven Markle	50	No commission	Danielle Greene	80	0.15
	Laura Bissot	50	No commission	Mattea Marvins	80	0.10
	Mozhe Atkinson	50	No commission	David Lee	80	0.10
	James Marlow	50	No commission	Sundar Ande	80	0.10
	TJ Olson	50	No commission	Amit Banda	80	0.10
	Jason Mallin	50	No commission	Lisa Ozer	80	0.25
	Michael Rogers	50	No commission	Harrison Bloom	80	0.20

employee_name	department	commission
Taylor Fox	80	0.20
William Smith	80	0.15
Elizabeth Bates	80	0.15
Sundita Kumar	80	0.10
Ellen Abel	80	0.30
Alyssa Hutton	80	0.25
Jonathon Taylor	80	0.20
Jack Livingston	80	0.20
Kimberely Grant	NULL	0.15
Charles Johnson	80	0.10
Winston Taylor	50	No commission
Jean Fleaur	50	No commission
Martha Sullivan	50	No commission
Girard Geoni	50	No commission
Nandita Sarchand	50	No commission
Alexis Bull	50	No commission
Julia Dellinger	50	No commission
Anthony Cabrio	50	No commission
Kelly Chung	50	No commission
Jennifer Dilly	50	No commission
Timothy Gates	50	No commission
Randall Perkins	50	No commission
Sarah Bell	50	No commission
Britney Everett	50	No commission
Samuel McCain	50	No commission
Vance Jones	50	No commission
Alana Walsh	50	No commission
Kevin Feeney	50	No commission
Donald OConnell	50	No commission
Douglas Grant	50	No commission
Jennifer Whalen	10	No commission
Michael Hartstein	20	No commission
Pat Fay	20	No commission
Susan Mavris	40	No commission
Hermann Baer	70	No commission

Taylor Fox	80	0.20
William Smith	80	0.15
Elizabeth Bates	80	0.15
Sundita Kumar	80	0.10
Ellen Abel	80	0.30
Alyssa Hutton	80	0.25
Jonathon Taylor	80	0.20
Jack Livingston	80	0.20
Kimberely Grant	NULL	0.15
Charles Johnson	80	0.10
Winston Taylor	50	No commission
Jean Fleaur	50	No commission
Martha Sullivan	50	No commission
Girard Geoni	50	No commission
Nandita Sarchand	50	No commission
Alexis Bull	50	No commission
Julia Dellinger	50	No commission
Anthony Cabrio	50	No commission
Kelly Chung	50	No commission
Jennifer Dilly	50	No commission
Timothy Gates	50	No commission
Randall Perkins	50	No commission
Sarah Bell	50	No commission
Britney Everett	50	No commission
Samuel McCain	50	No commission
Vance Jones	50	No commission
Alana Walsh	50	No commission
Kevin Feeney	50	No commission
Donald OConnell	50	No commission
Douglas Grant	50	No commission
Jennifer Whalen	10	No commission
Michael Hartstein	20	No commission
Pat Fay	20	No commission
Susan Mavris	40	No commission
Hermann Baer	70	No commission

19. Display the employee's name, salary and commission multiplied by 2. If an Employee doesn't earn the commission, then display as 'No commission. Name the columns appropriately

```
select concat(first_name, ' ', last_name) as employee_name,
       salary,
       ifnull(cast(commission_pct * 2 as char), 'No commission') as
double_commission
from employees;
```

employee_name	salary	double_commission			
Steven King	24000.00	No commission	Stephen Stiles	3200.00	No commission
Neena Kochhar	17000.00	No commission	John Seo	2700.00	No commission
Lex De Haan	17000.00	No commission	Joshua Patel	2500.00	No commission
Alexander Hunold	9000.00	No commission	Trenna Rajs	3500.00	No commission
Bruce Ernst	6000.00	No commission	Curtis Davies	3100.00	No commission
David Austin	4800.00	No commission	Randall Matos	2600.00	No commission
Valli Pataballa	4800.00	No commission	Peter Vargas	2500.00	No commission
Diana Lorentz	4200.00	No commission	John Russell	14000.00	0.80
Nancy Greenberg	12000.00	No commission	Karen Partners	13500.00	0.60
Daniel Faviet	9000.00	No commission	Alberto Errazuriz	12000.00	0.60
John Chen	8200.00	No commission	Gerald Cambrault	11000.00	0.60
Ismael Sciarra	7700.00	No commission	Eleni Zlotkey	10500.00	0.40
Jose Manuel Urman	7800.00	No commission	Peter Tucker	10000.00	0.60
Luis Popp	6900.00	No commission	David Bernstein	9500.00	0.50
Den Raphaely	11000.00	No commission	Peter Hall	9000.00	0.50
Alexander Khoo	3100.00	No commission	Christopher Olsen	8000.00	0.40
Shelli Baida	2900.00	No commission	Nanette Cambrault	7500.00	0.40
Sigal Tobias	2800.00	No commission	Oliver Tuvault	7000.00	0.30
Guy Himuro	2600.00	No commission	Janette King	10000.00	0.70
Karen Colmenares	2500.00	No commission	Patrick Sully	9500.00	0.70
Matthew Weiss	8000.00	No commission	Allan McEwen	9000.00	0.70
Adam Frip	8200.00	No commission	Lindsey Smith	8000.00	0.60
Payam Kauffing	7900.00	No commission	Louise Doran	7500.00	0.60
Shanta Vollman	6500.00	No commission	Sarath Sewall	7000.00	0.50
Kevin Mourg	5800.00	No commission	Clara Vishney	10500.00	0.50
Julia Nayer	3200.00	No commission	Danielle Greene	9500.00	0.30
Irene Mikkilineni	2700.00	No commission	Mattea Marvins	7200.00	0.20
James Landry	2400.00	No commission	David Lee	6800.00	0.20
Steven Markle	2200.00	No commission	Sundar Ande	6400.00	0.20
Laura Bissot	3300.00	No commission	Amit Banda	6200.00	0.20
Mozhe Atkinson	2800.00	No commission	Lisa Ozer	11500.00	0.50
James Marlow	2500.00	No commission	Harrison Bloom	10000.00	0.40
TJ Olson	2100.00	No commission	Taylor Fox	9600.00	0.40
Jason Mallin	3300.00	No commission	William Smith	7400.00	0.30
Michael Rogers	2900.00	No commission	Elizabeth Bates	7300.00	0.30
Ki Gee	2400.00	No commission	Sundita Kumar	6100.00	0.20
Hazel Philtanker	2200.00	No commission	Ellen Abel	11000.00	0.60
Renske Ladwig	3600.00	No commission	Alyssa Hutton	8800.00	0.50



Jonathon Taylor	8600.00	0.40
Jack Livingston	8400.00	0.40
Kimberely Grant	7000.00	0.30
Charles Johnson	6200.00	0.20
Winston Taylor	3200.00	No commission
Jean Fleaur	3100.00	No commission
Martha Sullivan	2500.00	No commission
Girard Geoni	2800.00	No commission
Nandita Sarchand	4200.00	No commission
Alexis Bull	4100.00	No commission
Julia Dellinger	3400.00	No commission
Anthony Cabrio	3000.00	No commission
Kelly Chung	3800.00	No commission
Jennifer Dilly	3600.00	No commission
Timothy Gates	2900.00	No commission
Randall Perkins	2500.00	No commission
Sarah Bell	4000.00	No commission
Britney Everett	3900.00	No commission
Samuel McCain	3200.00	No commission
Vance Jones	2800.00	No commission
Alana Walsh	3100.00	No commission
Kevin Feeney	3000.00	No commission
Donald OConnell	2600.00	No commission
Douglas Grant	2600.00	No commission
Jennifer Whalen	4400.00	No commission
Michael Hartstein	13000.00	No commission
Pat Fay	6000.00	No commission
Susan Mavris	6500.00	No commission
Hermann Baer	10000.00	No commission
Shelley Higgins	12000.00	No commission
William Gietz	8300.00	No commission

20. Display the employee's name, department id who have the first name same as another employee in the same department

```
select e1.first_name, e1.last_name, e1.department_id
from employees e1
join employees e2
on e1.first_name = e2.first_name
and e1.department_id = e2.department_id
and e1.employee_id <> e2.employee_id
order by e1.department_id, e1.first_name;
```

first_name	last_name	department_id
James	Marlow	50
James	Landry	50
Julia	Dellinger	50
Julia	Nayer	50
Kevin	Feeney	50
Kevin	Mourgos	50
Randall	Perkins	50
Randall	Matos	50
David	Bernstein	80
David	Lee	80
Peter	Tucker	80
Peter	Hall	80

21. Display the sum of salaries of the employees working under each Manager.

```
select manager_id, sum(salary) as total_salary
from employees
where manager_id is not null
group by manager_id
order by manager_id;
```

manager_id	total_salary
100	155400.00
101	44900.00
102	9000.00
103	19800.00
108	39600.00
114	13900.00
120	22100.00
121	25400.00
122	23600.00
123	25900.00
124	23000.00
145	51000.00
146	51000.00
147	46600.00
148	51900.00
149	50000.00
201	6000.00
205	8300.00

22. Select the Managers name, the count of employees working under and the department ID of the manager.

```

select m.first_name || ' ' || m.last_name as manager_name,
       count(e.employee_id) as employee_count,
       m.department_id
from employees e
join employees m
on e.manager_id = m.employee_id
group by m.first_name, m.last_name, m.department_id
order by m.department_id;

```

manager_name	employee_count	department_id
0	1	20
0	5	30
0	8	50
0	8	50
0	8	50
0	8	50
0	8	50
0	8	50
0	4	60
0	6	80
0	6	80
0	6	80
0	6	80
0	6	80
0	14	90
0	5	90
0	1	90
0	5	100
0	1	110

23. Select the employee name, department id, and the salary. Group the result with the manager name and the employee last name should have second letter 'a'!

```
select concat(e.first_name, ' ', e.last_name) as employee_name,
       e.department_id,
       e.salary,
       concat(m.first_name, ' ', m.last_name) as manager_name
from employees e
join employees m
on e.manager_id = m.employee_id
where e.last_name like '_a%'
group by manager_name, e.first_name, e.last_name, e.department_id, e.salary
order by manager_name;
```

employee_name	department_id	salary	manager_name
Anthony Cabrio	50	3000.00	Adam Fripp
James Marlow	50	2500.00	Adam Fripp
Nandita Sarchand	50	4200.00	Adam Fripp
Amit Banda	80	6200.00	Alberto Errazuriz
Mattea Marvins	80	7200.00	Alberto Errazuriz
Valli Pataballa	60	4800.00	Alexander Hunold
Shelli Baida	30	2900.00	Den Raphaely
Jonathon Taylor	80	8600.00	Eleni Zlotkey
Elizabeth Bates	80	7300.00	Gerald Cambrault
Nanette Cambrault	80	7500.00	John Russell
Peter Hall	80	9000.00	John Russell
Alana Walsh	50	3100.00	Kevin Mourgos
Curtis Davies	50	3100.00	Kevin Mourgos
Peter Vargas	50	2500.00	Kevin Mourgos
Randall Matos	50	2600.00	Kevin Mourgos
Trenna Rajs	50	3500.00	Kevin Mourgos
James Landry	50	2400.00	Matthew Weiss
Julia Nayer	50	3200.00	Matthew Weiss
Steven Markle	50	2200.00	Matthew Weiss
Winston Taylor	50	3200.00	Matthew Weiss
Pat Fay	20	6000.00	Michael Hartstein
Daniel Faviet	100	9000.00	Nancy Greenberg
Hermann Baer	70	10000.00	Neena Kochhar
Susan Mavris	40	6500.00	Neena Kochhar
Jason Mallin	50	3300.00	Payam Kaufling
Timothy Gates	50	2900.00	Payam Kaufling
Joshua Patel	50	2500.00	Shanta Vollman
Renske Ladwig	50	3600.00	Shanta Vollman
Den Raphaely	30	11000.00	Steven King
Gerald Cambrault	80	11000.00	Steven King
Karen Partners	80	13500.00	Steven King
Michael Hartstein	20	13000.00	Steven King
Payam Kaufling	50	7900.00	Steven King

24. Display the average of sum of the salaries and group the result with the department id. Order the result with the department id.

```
select department_id,  
       avg(sum(salary)) over () as avg_salary  
from employees  
group by department_id  
order by department_id;
```

department_id	avg_salary
NULL	57616.666667
10	57616.666667
20	57616.666667
30	57616.666667
40	57616.666667
50	57616.666667
60	57616.666667
70	57616.666667
80	57616.666667
90	57616.666667
100	57616.666667
110	57616.666667

25. Select the maximum salary of each department along with the department id

```
select department_id, max(salary) as max_salary  
from employees  
group by department_id  
order by department_id;
```

department_id	max_salary
NULL	7000.00
10	4400.00
20	13000.00
30	11000.00
40	6500.00
50	8200.00
60	9000.00
70	10000.00
80	14000.00
90	24000.00
100	12000.00
110	12000.00

26. Display the commission, if not null display 10% of salary, if null display a default value 1

```
select employee_id,
       salary,
       case
         when commission_pct is not null then salary * 0.10
         else 1
       end as commission_value
from employees;
```

employee_id	salary	commission_value	employee_id	salary	commission_value
100	24000.00	1	138	3200.00	1
101	17000.00	1	139	2700.00	1
102	17000.00	1	140	2500.00	1
103	9000.00	1	141	3500.00	1
104	6000.00	1	142	3100.00	1
105	4800.00	1	143	2600.00	1
106	4800.00	1	144	2500.00	1
107	4200.00	1	145	14000.00	1400.0000
108	12000.00	1	146	13500.00	1350.0000
109	9000.00	1	147	12000.00	1200.0000
110	8200.00	1	148	11000.00	1100.0000
111	7700.00	1	149	10500.00	1050.0000
112	7800.00	1	150	10000.00	1000.0000
113	6900.00	1	151	9500.00	950.0000
114	11000.00	1	152	9000.00	900.0000
115	3100.00	1	153	8000.00	800.0000
116	2900.00	1	154	7500.00	750.0000
117	2800.00	1	155	7000.00	700.0000
118	2600.00	1	156	10000.00	1000.0000
119	2500.00	1	157	9500.00	950.0000
120	8000.00	1	158	9000.00	900.0000
121	8200.00	1	159	8000.00	800.0000
122	7900.00	1	160	7500.00	750.0000
123	6500.00	1	161	7000.00	700.0000
124	5800.00	1	162	10500.00	1050.0000
125	3200.00	1	163	9500.00	950.0000
126	2700.00	1	164	7200.00	720.0000
127	2400.00	1	165	6800.00	680.0000
128	2200.00	1	166	6400.00	640.0000
129	3300.00	1	167	6200.00	620.0000
130	2800.00	1	168	11500.00	1150.0000
131	2500.00	1	169	10000.00	1000.0000
132	2100.00	1	170	9600.00	960.0000
133	3300.00	1	171	7400.00	740.0000
134	2900.00	1	172	7300.00	730.0000
135	2400.00	1	173	6100.00	610.0000
136	2200.00	1	174	11000.00	1100.0000
137	3600.00	1	175	8800.00	880.0000



176	8600.00	860.0000
177	8400.00	840.0000
178	7000.00	700.0000
179	6200.00	620.0000
180	3200.00	1
181	3100.00	1
182	2500.00	1
183	2800.00	1
184	4200.00	1
185	4100.00	1
186	3400.00	1
187	3000.00	1
188	3800.00	1
189	3600.00	1
190	2900.00	1
191	2500.00	1
192	4000.00	1
193	3900.00	1
194	3200.00	1
195	2800.00	1
196	3100.00	1
197	3000.00	1
198	2600.00	1
199	2600.00	1
200	4400.00	1
201	13000.00	1
202	6000.00	1
203	6500.00	1
204	10000.00	1
205	12000.00	1
206	8300.00	1

27. Write a query that displays the employee's last names only from the string's 2-5th position with the first letter capitalized and all other letters lowercase, Give each column an appropriate label.

```
select concat(upper(substr(last_name, 2, 1)), lower(substr(last_name, 3, 4))) as
formatted_name
from employees;
```

formatted_name	formatted_name	
Ing	Tiles	
Ochha	Eo	
E haa	Atel	
Unold	Ajs	
Rnst	Avies	
Ustin	Atos	
Ataba	Argas	
Orent	Ussel	Aylor
Reenb	Artne	Iving
Aviet	Rrazu	Rant
Hen	Ambra	Ohnso
Ciarr	Lotke	Aylor
Rman	Ucker	Leaur
Opp	Ernst	Ulliv
Aphae	All	Eoni
Hoo	Lsen	Archa
Aida	Ambra	Ull
Obias	Uvaul	Ellin
Imuro	Ing	Abrio
Olmen	Ully	Hung
Eiss	Cewen	Illy
Ripp	Mith	Ates
Aufli	Oran	Erkin
Ollma	Ewall	Ell
Ourgo	Ishne	Veret
Ayer	Reene	Ccain
Ikkil	Arvin	Ones
Andry	Ee	Alsh
Arkle	Nde	Eeney
Issot	Anda	Conne
Tkins	Zer	Rant
Arlow	Loom	Halen
Lson	Ox	Artst
Allin	Mith	Ay
Ogers	Ates	Avris
Ee	Umar	Aer
Hilta	Bel	Iggin
Adwia	Utton	Ietz

28. Write a query that displays the employee's first name and last name along with a " in between for e.g.: first name : Ram; last name : Kumar then Ram-Kumar. Also displays the month on which the employee has joined.

```
select concat(first_name, '-', last_name) as full_name,
       monthname(hire_date) as joining_month
from employees;
```

full_name	joining_month	full_name	joining_month	full_name	joining_month
Steven-King	June	Stephen-Stiles	October	Jonathon-Taylor	March
Neena-Kochhar	September	John-Seo	February	Jack-Livingston	April
Lex-De Haan	January	Joshua-Patel	April	Kimberely-Grant	May
Alexander-Hunold	January	Trenna-Rajs	October	Charles-Johnson	January
Bruce-Ernst	May	Curtis-Davies	January	Winston-Taylor	January
David-Austin	June	Randall-Matos	March	Jean-Fleur	February
Valli-Pataballa	February	Peter-Vargas	July	Martha-Sullivan	June
Diana-Lorentz	February	John-Russell	October	Girard-Geoni	February
Nancy-Greenberg	August	Karen-Partners	January	Nandita-Sarchand	January
Daniel-Faviet	August	Alberto-Erazuriz	March	Alexis-Bull	February
John-Chen	September	Gerald-Cambrault	October	Julia-Dellinger	June
Ismael-Sciarra	September	Eleni-Zlotkey	January	Anthony-Cabrio	February
Jose Manuel-Urman	March	Peter-Tucker	January	Kelly-Chung	June
Luis-Popp	December	David-Bernstein	March	Jennifer-Dilly	August
Den-Raphaely	December	Peter-Hall	August	Timothy-Gates	July
Alexander-Khoo	May	Christopher-Olsen	March	Randall-Perkins	December
Shelli-Baida	December	Nanette-Cambrault	December	Sarah-Bell	February
Sigal-Tobias	July	Oliver-Tuvault	November	Britney-Everett	March
Guy-Himuro	November	Janette-King	January	Samuel-McCain	July
Karen-Colmenares	August	Patrick-Sully	March	Vance-Jones	March
Matthew-Weiss	July	Allan-McEwen	August	Alana-Walsh	April
Adam-Fripp	April	Lindsey-Smith	March	Kevin-Feeney	May
Payam-Kaufling	May	Louise-Doran	December	Donald-OConnell	June
Shanta-Vollman	October	Sarath-Sewall	November	Douglas-Grant	January
Kevin-Mourgos	November	Clara-Vishney	November	Jennifer-Whalen	September
Julia-Nayer	July	Danielle-Greene	March	Michael-Hartstein	February
Irene-Mikkilineni	September	Mattea-Marvins	January	Pat-Fay	August
James-Landry	January	David-Lee	February	Susan-Mavris	June
Steven-Markle	March	Sundar-Ande	March	Hermann-Baer	June
Laura-Bissot	August	Amit-Banda	April	Shelley-Higgins	June
Mozhe-Atkinson	October	Lisa-Ozer	March	William-Gietz	June
James-Marlow	February	Harrison-Bloom	March		
TJ-Olson	April	Tayler-Fox	January		
Jason-Mallin	June	William-Smith	February		
Michael-Rogers	August	Elizabeth-Bates	March		
Ki-Gee	December	Sundita-Kumar	April		
Hazel-Philtanker	February	Ellen-Abel	May		
Renske-Ladwig	July	Alyssa-Hutton	March		

29. Write a query to display the employee's last name and if half of the salary is greater than ten thousand then increase the salary by 10% else by 11.5% along with the bonus amount of 1500 each. Provide each column an appropriate label.

```
select last_name,
       salary,
       case
         when salary / 2 > 10000 then salary * 1.10 -- Increase by 10%
         else salary * 1.115 -- Increase by 11.5%
       end as updated_salary,
       1500 as bonus_amount
from employees;
```

last_name	salary	updated_salary	bonus_amount	last_name	salary	updated_salary	bonus_amount
King	24000.00	26400.00000	1500	Stiles	3200.00	3568.00000	1500
Kochhar	17000.00	18955.00000	1500	Seo	2700.00	3010.50000	1500
De Haan	17000.00	18955.00000	1500	Patel	2500.00	2787.50000	1500
Hunold	9000.00	10035.00000	1500	Rajs	3500.00	3902.50000	1500
Ernst	6000.00	6690.00000	1500	Davies	3100.00	3456.50000	1500
Austin	4800.00	5352.00000	1500	Matos	2600.00	2899.00000	1500
Pataballa	4800.00	5352.00000	1500	Vargas	2500.00	2787.50000	1500
Lorentz	4200.00	4683.00000	1500	Russell	14000.00	15610.00000	1500
Greenberg	12000.00	13380.00000	1500	Partners	13500.00	15052.50000	1500
Faviet	9000.00	10035.00000	1500	Errazuriz	12000.00	13380.00000	1500
Chen	8200.00	9143.00000	1500	Cambrault	11000.00	12265.00000	1500
Sciarra	7700.00	8585.50000	1500	Zlotkey	10500.00	11707.50000	1500
Urman	7800.00	8697.00000	1500	Tucker	10000.00	11150.00000	1500
Popp	6900.00	7693.50000	1500	Bernstein	9500.00	10592.50000	1500
Raphaely	11000.00	12265.00000	1500	Hall	9000.00	10035.00000	1500
Khoo	3100.00	3456.50000	1500	Olsen	8000.00	8920.00000	1500
Baida	2900.00	3233.50000	1500	Cambrault	7500.00	8362.50000	1500
Tobias	2800.00	3122.00000	1500	Tuvault	7000.00	7805.00000	1500
Himuro	2600.00	2899.00000	1500	King	10000.00	11150.00000	1500
Colmenares	2500.00	2787.50000	1500	Sully	9500.00	10592.50000	1500
Weiss	8000.00	8920.00000	1500	McEwen	9000.00	10035.00000	1500
Fripp	8200.00	9143.00000	1500	Smith	8000.00	8920.00000	1500
Kaufing	7900.00	8808.50000	1500	Doran	7500.00	8362.50000	1500
Vollman	6500.00	7247.50000	1500	Sewall	7000.00	7805.00000	1500
Mourgos	5800.00	6467.00000	1500	Vishney	10500.00	11707.50000	1500
Nayer	3200.00	3568.00000	1500	Greene	9500.00	10592.50000	1500
Mikkilineni	2700.00	3010.50000	1500	Marvins	7200.00	8028.00000	1500
Landry	2400.00	2676.00000	1500	Lee	6800.00	7582.00000	1500
Markle	2200.00	2453.00000	1500	Ande	6400.00	7136.00000	1500
Bissot	3300.00	3679.50000	1500	Banda	6200.00	6913.00000	1500
Atkinson	2800.00	3122.00000	1500	Ozer	11500.00	12822.50000	1500
Marlow	2500.00	2787.50000	1500	Bloom	10000.00	11150.00000	1500
Olson	2100.00	2341.50000	1500	Fox	9600.00	10704.00000	1500
Mallin	3300.00	3679.50000	1500	Smith	7400.00	8251.00000	1500
Rogers	2900.00	3233.50000	1500	Bates	7300.00	8139.50000	1500
Gee	2400.00	2676.00000	1500	Kumar	6100.00	6801.50000	1500
Philtanker	2200.00	2453.00000	1500	Abel	11000.00	12265.00000	1500
Ladwig	3600.00	4014.00000	1500	Hutton	8800.00	9812.00000	1500

Taylor	8600.00	9589.00000	1500
Livingston	8400.00	9366.00000	1500
Grant	7000.00	7805.00000	1500
Johnson	6200.00	6913.00000	1500
Taylor	3200.00	3568.00000	1500
Fleur	3100.00	3456.50000	1500
Sullivan	2500.00	2787.50000	1500
Geoni	2800.00	3122.00000	1500
Sarchand	4200.00	4683.00000	1500
Bull	4100.00	4571.50000	1500
Dellinger	3400.00	3791.00000	1500
Cabrio	3000.00	3345.00000	1500
Chung	3800.00	4237.00000	1500
Dilly	3600.00	4014.00000	1500
Gates	2900.00	3233.50000	1500
Perkins	2500.00	2787.50000	1500
Bell	4000.00	4460.00000	1500
Everett	3900.00	4348.50000	1500
McCain	3200.00	3568.00000	1500
Jones	2800.00	3122.00000	1500
Walsh	3100.00	3456.50000	1500
Feeney	3000.00	3345.00000	1500
OConnell	2600.00	2899.00000	1500
Grant	2600.00	2899.00000	1500
Whalen	4400.00	4906.00000	1500
Hartstein	13000.00	14495.00000	1500
Fay	6000.00	6690.00000	1500
Mavris	6500.00	7247.50000	1500
Baer	10000.00	11150.00000	1500
Higgins	12000.00	13380.00000	1500
Gietz	8300.00	9254.50000	1500

30. Display the employee ID by Appending two zeros after 2nd digit and 'E' in the end, department id, salary and the manager name all in Upper case, if the Manager name consists of 'z' replace it with '\$'!

```
select concat(left(e.employee_id, 2), '00', right(e.employee_id,
length(e.employee_id) - 2), 'E') as formatted_employee_id,
       e.department_id,
       e.salary,
       upper(replace(concat(m.first_name, ' ', m.last_name), 'z', '$')) as
manager_name
from employees e
left join employees m
on e.manager_id = m.employee_id
limit 1000;
```

formatted_employee_id	department_id	salary	manager_name	formatted_employee_id	department_id	salary	manager_name
10000E	90	24000.00	NULL	13006E	50	2200.00	PAYAM KAUFING
10001E	90	17000.00	STEVEN KING	13007E	50	3600.00	SHANTA VOLLMAN
10002E	90	17000.00	STEVEN KING	13008E	50	3200.00	SHANTA VOLLMAN
10003E	60	9000.00	LEX DE HAAN	13009E	50	2700.00	SHANTA VOLLMAN
10004E	60	6000.00	ALEXANDER HUNOLD	14000E	50	2500.00	SHANTA VOLLMAN
10005E	60	4800.00	ALEXANDER HUNOLD	14001E	50	3500.00	KEVIN MOURGOS
10006E	60	4800.00	ALEXANDER HUNOLD	14002E	50	3100.00	KEVIN MOURGOS
10007E	60	4200.00	ALEXANDER HUNOLD	14003E	50	2600.00	KEVIN MOURGOS
10008E	100	12000.00	NEENA KOCHHAR	14004E	50	2500.00	KEVIN MOURGOS
10009E	100	9000.00	NANCY GREENBERG	14005E	80	14000.00	STEVEN KING
11000E	100	8200.00	NANCY GREENBERG	14006E	80	13500.00	STEVEN KING
11001E	100	7700.00	NANCY GREENBERG	14007E	80	12000.00	STEVEN KING
11002E	100	7800.00	NANCY GREENBERG	14008E	80	11000.00	STEVEN KING
11003E	100	6900.00	NANCY GREENBERG	14009E	80	10500.00	STEVEN KING
11004E	30	11000.00	STEVEN KING	15000E	80	10000.00	JOHN RUSSELL
11005E	30	3100.00	DEN RAPHAELY	15001E	80	9500.00	JOHN RUSSELL
11006E	30	2900.00	DEN RAPHAELY	15002E	80	9000.00	JOHN RUSSELL
11007E	30	2800.00	DEN RAPHAELY	15003E	80	8000.00	JOHN RUSSELL
11008E	30	2600.00	DEN RAPHAELY	15004E	80	7500.00	JOHN RUSSELL
11009E	30	2500.00	DEN RAPHAELY	15005E	80	7000.00	JOHN RUSSELL
12000E	50	8000.00	STEVEN KING	15006E	80	10000.00	KAREN PARTNERS
12001E	50	8200.00	STEVEN KING	15007E	80	9500.00	KAREN PARTNERS
12002E	50	7900.00	STEVEN KING	15008E	80	9000.00	KAREN PARTNERS
12003E	50	6500.00	STEVEN KING	15009E	80	8000.00	KAREN PARTNERS
12004E	50	5800.00	STEVEN KING	16000E	80	7500.00	KAREN PARTNERS
12005E	50	3200.00	MATTHEW WEISS	16001E	80	7000.00	KAREN PARTNERS
12006E	50	2700.00	MATTHEW WEISS	16002E	80	10500.00	ALBERTO ERRA\$URI\$
12007E	50	2400.00	MATTHEW WEISS	16003E	80	9500.00	ALBERTO ERRA\$URI\$
12008E	50	2200.00	MATTHEW WEISS	16004E	80	7200.00	ALBERTO ERRA\$URI\$
12009E	50	3300.00	ADAM FRIPP	16005E	80	6800.00	ALBERTO ERRA\$URI\$
13000E	50	2800.00	ADAM FRIPP	16006E	80	6400.00	ALBERTO ERRA\$URI\$
13001E	50	2500.00	ADAM FRIPP	16007E	80	6200.00	ALBERTO ERRA\$URI\$
13002E	50	2100.00	ADAM FRIPP	16008E	80	11500.00	GERALD CAMBRAULT
13003E	50	3300.00	PAYAM KAUFING	16009E	80	10000.00	GERALD CAMBRAULT
13004E	50	2900.00	PAYAM KAUFING	17000E	80	9600.00	GERALD CAMBRAULT
13005E	50	2400.00	PAYAM KAUFING	17001E	80	7400.00	GERALD CAMBRAULT



17002E	80	7300.00	GERALD CAMBRAULT
17003E	80	6100.00	GERALD CAMBRAULT
17004E	80	11000.00	ELENI ZLOTKEY
17005E	80	8800.00	ELENI ZLOTKEY
17006E	80	8600.00	ELENI ZLOTKEY
17007E	80	8400.00	ELENI ZLOTKEY
17008E	NULL	7000.00	ELENI ZLOTKEY
17009E	80	6200.00	ELENI ZLOTKEY
18000E	50	3200.00	MATTHEW WEISS
18001E	50	3100.00	MATTHEW WEISS
18002E	50	2500.00	MATTHEW WEISS
18003E	50	2800.00	MATTHEW WEISS
18004E	50	4200.00	ADAM FRIPP
18005E	50	4100.00	ADAM FRIPP
18006E	50	3400.00	ADAM FRIPP
18007E	50	3000.00	ADAM FRIPP
18008E	50	3800.00	PAYAM KAUFILING
18009E	50	3600.00	PAYAM KAUFILING
19000E	50	2900.00	PAYAM KAUFILING
19001E	50	2500.00	PAYAM KAUFILING
19002E	50	4000.00	SHANTA VOLLMAN
19003E	50	3900.00	SHANTA VOLLMAN
19004E	50	3200.00	SHANTA VOLLMAN
19005E	50	2800.00	SHANTA VOLLMAN
19006E	50	3100.00	KEVIN MOURGOS
19007E	50	3000.00	KEVIN MOURGOS
19008E	50	2600.00	KEVIN MOURGOS
19009E	50	2600.00	KEVIN MOURGOS
20000E	10	4400.00	NEENA KOCHHAR
20001E	20	13000.00	STEVEN KING
20002E	20	6000.00	MICHAEL HARTSTEIN
20003E	40	6500.00	NEENA KOCHHAR
20004E	70	10000.00	NEENA KOCHHAR
20005E	110	12000.00	NEENA KOCHHAR
20006E	110	8300.00	SHELLEY HIGGINS

31. Write a query that displays the employee's last names with the first letter capitalized and all other letters lowercase, and the length of the names, for all employees whose name starts with J, A, or M. Give each column an appropriate label. Sort the results by the employees' last names

```
select concat(upper(left(last_name, 1)), lower(substring(last_name, 2))) as  
formatted_last_name,  
length(last_name) as name_length  
from employees  
where last_name like 'J%' or last_name like 'A%' or last_name like 'M%'  
order by last_name;
```

formatted_last_name	name_length
Abel	4
Ande	4
Atkinson	8
Austin	6
Johnson	7
Jones	5
Mallin	6
Markle	6
Marlow	6
Marvins	7
Matos	5
Mavris	6
Mccain	6
Mcewen	6
Mikkilineni	11
Mourgos	7

32. Create a query to display the last name and salary for all employees. Format the salary to be 15 characters long, left-padded with \$. Label the column SALARY

```
select last_name,
       lpad(salary, 15, '$') as salary
from employees;
```

last_name	salary	last_name	salary		
King	\$\$\$\$\$\$\$24000.00	Stiles	\$\$\$\$\$\$\$3200.00		
Kochhar	\$\$\$\$\$\$\$17000.00	Seo	\$\$\$\$\$\$\$2700.00		
De Haan	\$\$\$\$\$\$\$17000.00	Patel	\$\$\$\$\$\$\$2500.00		
Hunold	\$\$\$\$\$\$\$9000.00	Rajs	\$\$\$\$\$\$\$3500.00		
Ernst	\$\$\$\$\$\$\$6000.00	Davies	\$\$\$\$\$\$\$3100.00		
Austin	\$\$\$\$\$\$\$4800.00	Matos	\$\$\$\$\$\$\$2600.00		
Pataballa	\$\$\$\$\$\$\$4800.00	Vargas	\$\$\$\$\$\$\$2500.00		
Lorentz	\$\$\$\$\$\$\$4200.00	Russell	\$\$\$\$\$\$\$14000.00	Taylor	\$\$\$\$\$\$\$8600.00
Greenberg	\$\$\$\$\$\$\$12000.00	Partners	\$\$\$\$\$\$\$13500.00	Livingston	\$\$\$\$\$\$\$8400.00
Faviet	\$\$\$\$\$\$\$9000.00	Errazuriz	\$\$\$\$\$\$\$12000.00	Grant	\$\$\$\$\$\$\$7000.00
Chen	\$\$\$\$\$\$\$8200.00	Cambrault	\$\$\$\$\$\$\$11000.00	Johnson	\$\$\$\$\$\$\$6200.00
Sciarra	\$\$\$\$\$\$\$7700.00	Zlotkey	\$\$\$\$\$\$\$10500.00	Taylor	\$\$\$\$\$\$\$3200.00
Urman	\$\$\$\$\$\$\$7800.00	Tucker	\$\$\$\$\$\$\$10000.00	Fleaur	\$\$\$\$\$\$\$3100.00
Popp	\$\$\$\$\$\$\$6900.00	Bernstein	\$\$\$\$\$\$\$9500.00	Sullivan	\$\$\$\$\$\$\$2500.00
Raphaely	\$\$\$\$\$\$\$11000.00	Hall	\$\$\$\$\$\$\$9000.00	Geoni	\$\$\$\$\$\$\$2800.00
Khoo	\$\$\$\$\$\$\$3100.00	Olsen	\$\$\$\$\$\$\$8000.00	Sarchand	\$\$\$\$\$\$\$4200.00
Baida	\$\$\$\$\$\$\$2900.00	Cambrault	\$\$\$\$\$\$\$7500.00	Bull	\$\$\$\$\$\$\$4100.00
Tobias	\$\$\$\$\$\$\$2800.00	Tuvault	\$\$\$\$\$\$\$7000.00	Dellinger	\$\$\$\$\$\$\$3400.00
Himuro	\$\$\$\$\$\$\$2600.00	King	\$\$\$\$\$\$\$10000.00	Cabrio	\$\$\$\$\$\$\$3000.00
Colmenares	\$\$\$\$\$\$\$2500.00	Sully	\$\$\$\$\$\$\$9500.00	Chung	\$\$\$\$\$\$\$3800.00
Weiss	\$\$\$\$\$\$\$8000.00	McEwen	\$\$\$\$\$\$\$9000.00	Dilly	\$\$\$\$\$\$\$3600.00
Fripp	\$\$\$\$\$\$\$8200.00	Smith	\$\$\$\$\$\$\$8000.00	Gates	\$\$\$\$\$\$\$2900.00
Kaufling	\$\$\$\$\$\$\$7900.00	Doran	\$\$\$\$\$\$\$7500.00	Perkins	\$\$\$\$\$\$\$2500.00
Vollman	\$\$\$\$\$\$\$6500.00	Sewall	\$\$\$\$\$\$\$7000.00	Bell	\$\$\$\$\$\$\$4000.00
Mourgos	\$\$\$\$\$\$\$5800.00	Vishney	\$\$\$\$\$\$\$10500.00	Everett	\$\$\$\$\$\$\$3900.00
Nayer	\$\$\$\$\$\$\$3200.00	Greene	\$\$\$\$\$\$\$9500.00	McCain	\$\$\$\$\$\$\$3200.00
Mikkilineni	\$\$\$\$\$\$\$2700.00	Marvins	\$\$\$\$\$\$\$7200.00	Jones	\$\$\$\$\$\$\$2800.00
Landry	\$\$\$\$\$\$\$2400.00	Lee	\$\$\$\$\$\$\$6800.00	Walsh	\$\$\$\$\$\$\$3100.00
Markle	\$\$\$\$\$\$\$2200.00	Ande	\$\$\$\$\$\$\$6400.00	Feeney	\$\$\$\$\$\$\$3000.00
Bissot	\$\$\$\$\$\$\$3300.00	Banda	\$\$\$\$\$\$\$6200.00	OConnell	\$\$\$\$\$\$\$2600.00
Atkinson	\$\$\$\$\$\$\$2800.00	Ozer	\$\$\$\$\$\$\$11500.00	Grant	\$\$\$\$\$\$\$2600.00
Marlow	\$\$\$\$\$\$\$2500.00	Bloom	\$\$\$\$\$\$\$10000.00	Whalen	\$\$\$\$\$\$\$4400.00
Olson	\$\$\$\$\$\$\$2100.00	Fox	\$\$\$\$\$\$\$9600.00	Hartstein	\$\$\$\$\$\$\$13000.00
Mallin	\$\$\$\$\$\$\$3300.00	Smith	\$\$\$\$\$\$\$7400.00	Fay	\$\$\$\$\$\$\$6000.00
Rogers	\$\$\$\$\$\$\$2900.00	Bates	\$\$\$\$\$\$\$7300.00	Mavris	\$\$\$\$\$\$\$6500.00
Gee	\$\$\$\$\$\$\$2400.00	Kumar	\$\$\$\$\$\$\$6100.00	Baer	\$\$\$\$\$\$\$10000.00
Philtanker	\$\$\$\$\$\$\$2200.00	Abel	\$\$\$\$\$\$\$11000.00	Higgins	\$\$\$\$\$\$\$12000.00
Ladwig	\$\$\$\$\$\$\$3600.00	Hutton	\$\$\$\$\$\$\$8800.00	Gietz	\$\$\$\$\$\$\$8300.00

33. Display the employee's name if it is a palindrome

```
select first_name  
from employees  
where lower(first_name) = reverse(lower(first_name));
```

	first_name

34. Display First names of all employees with initcaps.

```
select concat(upper(left(first_name, 1)), lower(substring(first_name, 2))) as
formatted_first_name
from employees;
```

formatted_first_name	formatted_first_name	
Steven	Stephen	
Neena	John	
Lex	Joshua	
Alexander	Trenna	
Bruce	Curtis	
David	Randall	
Valli	Peter	
Diana	John	Jonathon
Nancy	Karen	Jack
Daniel	Alberto	Kimberely
John	Gerald	Charles
Ismael	Eleni	Winston
Jose manuel	Peter	Jean
Luis	David	Martha
Den	Peter	Girard
Alexander	Christopher	Nandita
Shelli	Nanette	Alexis
Sigal	Oliver	Julia
Guy	Janette	Anthony
Karen	Patrick	Kelly
Matthew	Allan	Jennifer
Adam	Lindsey	Timothy
Payam	Louise	Randall
Shanta	Sarath	Sarah
Kevin	Clara	Britney
Julia	Danielle	Samuel
Irene	Mattea	Vance
James	David	Alana
Steven	Sundar	Kevin
Laura	Amit	Donald
Mozhe	Lisa	Douglas
James	Harrison	Jennifer
Tj	Tayler	Michael
Jason	William	Pat
Michael	Elizabeth	Susan
Ki	Sundita	Hermann
Hazel	Ellen	Shelley
Renske	Alyssa	William

35. From the LOCATIONS table, extract the word between first and second space from the STREET ADDRESS column.

```
select substring_index(substring_index(street_address, ' ', 2), ' ', -1) as  
extracted_word  
from locations;
```

extracted_word
Via
Calle
Shinjuku-ku
Kamiya-cho
Jabberwocky
Interiors
Zagora
Charade
Spadina
Boxwood
Laogianggen
Vileparle
Victoria
Clementi
Arthur
Centre,
Chester
7031
Frei
Rue
921
Breughelstraat
Escobedo

36. Extract first letter from First Name column and append it with the Last Name. Also add "@systechusa.com" at the end. Name the column as e-mail address. All characters should be in lower case. Display this along with their First Name.

```
select first_name,
       lower(concat(left(first_name, 1), last_name, '@systechusa.com')) as
email_address
from employees;
```

first_name	email_address
Steven	sking@systechusa.com
Neena	nkochhar@systechusa.com
Lex	lde haan@systechusa.com
Alexander	ahunold@systechusa.com
Bruce	bernst@systechusa.com
David	daustin@systechusa.com
Valli	vpataballa@systechusa.com
Diana	dlorentz@systechusa.com
Nancy	ngreenberg@systechusa.com
Daniel	dfaviet@systechusa.com
John	jchen@systechusa.com
Ismael	isciarra@systechusa.com
Jose Manuel	jurman@systechusa.com
Luis	lpopp@systechusa.com
Den	draphaely@systechusa.com
Alexander	akhoo@systechusa.com
Shelli	sbaida@systechusa.com
Sigal	stobias@systechusa.com
Guy	ghimuro@systechusa.com
Karen	kcolmenares@systechusa.com
Matthew	mweiss@systechusa.com
Adam	afripp@systechusa.com
Payam	pkaufling@systechusa.com
Shanta	svollman@systechusa.com
Kevin	kmourgos@systechusa.com
Julia	jnayer@systechusa.com
Irene	imikkilineni@systechusa.com
James	jlandry@systechusa.com
Steven	smarkle@systechusa.com
Laura	lbissot@systechusa.com
Mozhe	matkinson@systechusa.com
James	jmarlow@systechusa.com
TJ	tolson@systechusa.com
Jason	jmallin@systechusa.com
Michael	mrogers@systechusa.com
Ki	kgee@systechusa.com
Hazel	hphiltanker@systechusa.com
Renske	rladwig@systechusa.com

first_name	email_address
Stephen	sstiles@systechusa.com
John	jseo@systechusa.com
Joshua	jpatel@systechusa.com
Trenna	trajs@systechusa.com
Curtis	cdavies@systechusa.com
Randall	rmatos@systechusa.com
Peter	pargas@systechusa.com
John	jrussell@systechusa.com
Karen	kpartners@systechusa.com
Alberto	aerrazuriz@systechusa.com
Gerald	gcambault@systechusa.com
Eleni	ezlotkey@systechusa.com
Peter	ptucker@systechusa.com
David	dbernstein@systechusa.com
Peter	phall@systechusa.com
Christopher	colsen@systechusa.com
Nanette	ncambault@systechusa.com
Oliver	otuvault@systechusa.com
Janette	jking@systechusa.com
Patrick	psully@systechusa.com
Allan	amcewen@systechusa.com
Lindsey	lsmith@systechusa.com
Louise	ldoran@systechusa.com
Sarath	ssewall@systechusa.com
Clara	cvishney@systechusa.com
Danielle	dgreene@systechusa.com
Mattea	mmarvins@systechusa.com
David	dlee@systechusa.com
Sundar	sande@systechusa.com
Amit	abanda@systechusa.com
Lisa	lozer@systechusa.com
Harrison	hbloom@systechusa.com
Taylor	tfox@systechusa.com
William	wsmith@systechusa.com
Elizabeth	ebates@systechusa.com
Sundita	skumar@systechusa.com
Ellen	eabel@systechusa.com
Alyssa	ahutton@systechusa.com

Jonathon	jtaylor@systechusa.com
Jack	jlivingston@systechusa.com
Kimberely	kgrant@systechusa.com
Charles	cjohnson@systechusa.com
Winston	wtaylor@systechusa.com
Jean	jfleur@systechusa.com
Martha	msullivan@systechusa.com
Girard	ggeoni@systechusa.com
Nandita	nsarchand@systechusa.com
Alexis	abull@systechusa.com
Julia	jdellinger@systechusa.com
Anthony	acabrio@systechusa.com
Kelly	kchung@systechusa.com
Jennifer	jdilly@systechusa.com
Timothy	tgates@systechusa.com
Randall	rperkins@systechusa.com
Sarah	sbell@systechusa.com
Britney	beverett@systechusa.com
Samuel	smccain@systechusa.com
Vance	vjones@systechusa.com
Alana	awalsh@systechusa.com
Kevin	kfeeney@systechusa.com
Donald	doconnell@systechusa.com
Douglas	dgrant@systechusa.com
Jennifer	jwhalen@systechusa.com
Michael	mhartstein@systechusa.com
Pat	pfay@systechusa.com
Susan	smavris@systechusa.com
Hermann	hbaer@systechusa.com
Shelley	shiggins@systechusa.com
William	wgietz@systechusa.com



37. Display the names and job titles of all employees with the same job as Trenna.

```
select first_name, last_name, job_title
from employees
where job_title = (select job_title from employees where first_name = 'Trenna');
```

first_name	last_name	job_id
Julia	Nayer	ST_CLERK
Irene	Mikkilineni	ST_CLERK
James	Landry	ST_CLERK
Steven	Markle	ST_CLERK
Laura	Bissot	ST_CLERK
Mozhe	Atkinson	ST_CLERK
James	Marlow	ST_CLERK
TJ	Olson	ST_CLERK
Jason	Mallin	ST_CLERK
Michael	Rogers	ST_CLERK
Ki	Gee	ST_CLERK
Hazel	Philtanker	ST_CLERK
Renske	Ladwig	ST_CLERK
Stephen	Stiles	ST_CLERK
John	Seo	ST_CLERK
Joshua	Patel	ST_CLERK
Trenna	Rajs	ST_CLERK
Curtis	Davies	ST_CLERK
Randall	Matos	ST_CLERK
Peter	Vargas	ST_CLERK

38. Display the names and department name of all employees working in the same city as Trena.

```
select e.first_name, e.last_name, d.department_name
from employees e
join departments d on e.department_id = d.department_id
join locations l on d.location_id = l.location_id
where l.city = (select l.city
                from employees e
                join departments d on e.department_id = d.department_id
                join locations l on d.location_id = l.location_id
                where e.first_name = 'Trena');
```

first_name	last_name	department_name			
Matthew	Weiss	Shipping			
Adam	Fripp	Shipping			
Payam	Kauffing	Shipping			
Shanta	Vollman	Shipping			
Kevin	Mourgos	Shipping			
Julia	Nayer	Shipping			
Irene	Mikkilineni	Shipping			
James	Landry	Shipping			
Steven	Markle	Shipping			
Laura	Bissot	Shipping			
Mozhe	Atkinson	Shipping			
James	Marlow	Shipping			
TJ	Olson	Shipping			
Jason	Mallin	Shipping			
Michael	Rogers	Shipping			
Ki	Gee	Shipping			
Hazel	Philtanker	Shipping			
Renske	Ladwig	Shipping			
Stephen	Stiles	Shipping			
John	Seo	Shipping			
Joshua	Patel	Shipping			
Trena	Rajs	Shipping			
Curtis	Davies	Shipping			
Randall	Matos	Shipping			
Peter	Vargas	Shipping			
Winston	Taylor	Shipping			
Jean	Fleaur	Shipping			
Martha	Sullivan	Shipping			
Girard	Geoni	Shipping			
Nandita	Sarchand	Shipping			
Alexis	Bull	Shipping			
Julia	Dellinger	Shipping	Britney	Everett	Shipping
Anthony	Cabrio	Shipping	Samuel	McCain	Shipping
Kelly	Chung	Shipping	Vance	Jones	Shipping
Jennifer	Dilly	Shipping	Alana	Walsh	Shipping
Timothy	Gates	Shipping	Kevin	Feeney	Shipping
Randall	Perkins	Shipping	Donald	OConnell	Shipping
Sarah	Bell	Shipping	Douglas	Grant	Shipping

39. Display the name of the employee whose salary is the lowest.

```
select first_name, last_name, salary
from employees
where salary = (select min(salary) from employees);
```

first_name	last_name	salary
TJ	Olson	2100.00

40. Display the names of all employees except the lowest paid.

```
select first_name, last_name, salary
from employees
where salary > (select min(salary) from employees);
```

first_name	last_name	salary	first_name	last_name	salary
Steven	King	24000.00	John	Seo	2700.00
Neena	Kochhar	17000.00	Joshua	Patel	2500.00
Alex	De Haan	17000.00	Trenna	Rajs	3500.00
Alexander	Hunold	9000.00	Curtis	Davies	3100.00
Bruce	Ernst	6000.00	Randall	Matos	2600.00
David	Austin	4800.00	Peter	Vargas	2500.00
Valli	Pataballa	4800.00	John	Russell	14000.00
Diana	Lorentz	4200.00	Karen	Partners	13500.00
Nancy	Greenberg	12000.00	Alberto	Errazuriz	12000.00
Daniel	Faviet	9000.00	Gerald	Cambrault	11000.00
John	Chen	8200.00	Eleni	Zlotkey	10500.00
Ismael	Sciarra	7700.00	Peter	Tucker	10000.00
Jose Manuel	Urman	7800.00	David	Bernstein	9500.00
Luis	Popp	6900.00	Peter	Hall	9000.00
Den	Raphaely	11000.00	Christopher	Olsen	8000.00
Alexander	Khoo	3100.00	Nanette	Cambrault	7500.00
Shelli	Baida	2900.00	Oliver	Tuvault	7000.00
Sigal	Tobias	2800.00	Janette	King	10000.00
Guy	Himuro	2600.00	Patrick	Sully	9500.00
Karen	Colmenares	2500.00	Allan	McEwen	9000.00
Matthew	Weiss	8000.00	Lindsey	Smith	8000.00
Adam	Fripp	8200.00	Louise	Doran	7500.00
Payam	Kaufing	7900.00	Sarath	Sewall	7000.00
Shanta	Vollman	6500.00	Clara	Vishney	10500.00
Kevin	Mourgos	5800.00	Danielle	Greene	9500.00
Julia	Nayer	3200.00	Mattea	Marvins	7200.00
Irene	Mikkilineni	2700.00	David	Lee	6800.00
James	Landry	2400.00	Sundar	Ande	6400.00
Steven	Markle	2200.00	Amit	Banda	6200.00
Laura	Bissot	3300.00	Lisa	Ozer	11500.00
Mozhe	Atkinson	2800.00	Harrison	Bloom	10000.00
James	Marlow	2500.00	Taylor	Fox	9600.00
Jason	Mallin	3300.00	William	Smith	7400.00
Michael	Rogers	2900.00	Elizabeth	Bates	7300.00
Gi	Gee	2400.00	Sundita	Kumar	6100.00
Hazel	Philtanker	2200.00	Ellen	Abel	11000.00
Renske	Ladwig	3600.00	Alyssa	Hutton	8800.00
Stephen	Stiles	3200.00	Jonathon	Taylor	8600.00
Jack	Livingston	8400.00	Kimberely	Grant	7000.00
Charles	Johnson	6200.00	Winston	Taylor	3200.00
Jean	Fleur	3100.00	Martha	Sullivan	2500.00
Girard	Geoni	2800.00	Nandita	Sarchand	4200.00
Alexis	Bull	4100.00	Julia	Dellinger	3400.00
Anthony	Cabrio	3000.00	Kelly	Chung	3800.00
Jennifer	Dilly	3600.00	Timothy	Gates	2900.00
Randall	Perkins	2500.00	Sarah	Bell	4000.00
Britney	Everett	3900.00	Samuel	McCain	3200.00
Vance	Jones	2800.00	Alana	Walsh	3100.00
Kevin	Feeney	3000.00	Donald	OConnell	2600.00
Douglas	Grant	2600.00	Jennifer	Whalen	4400.00
Michael	Hartstein	13000.00	Pat	Fay	6000.00
Susan	Mavris	6500.00	Hermann	Baer	10000.00
Shelley	Higgins	12000.00	William	Gietz	8300.00

41. Write a query to display the last name, department number, department name for all employees.

```
select e.last_name, e.department_id, d.department_name
from employees e
join departments d on e.department_id = d.department_id;
```

last_name	department_id	department_name	last_name	department_id	department_name
Whalen	10	Administration	Geoni	50	Shipping
Hartstein	20	Marketing	Sarchand	50	Shipping
Fay	20	Marketing	Bull	50	Shipping
Raphaely	30	Purchasing	Dellinger	50	Shipping
Khoo	30	Purchasing	Cabrio	50	Shipping
Baida	30	Purchasing	Chung	50	Shipping
Tobias	30	Purchasing	Dilly	50	Shipping
Himuro	30	Purchasing	Gates	50	Shipping
Colmenares	30	Purchasing	Perkins	50	Shipping
Mavris	40	Human Resources	Bell	50	Shipping
Weiss	50	Shipping	Everett	50	Shipping
Fripp	50	Shipping	McCain	50	Shipping
Kaufing	50	Shipping	Jones	50	Shipping
Vollman	50	Shipping	Walsh	50	Shipping
Mourgos	50	Shipping	Feeney	50	Shipping
Nayer	50	Shipping	OConnell	50	Shipping
Mikkilineni	50	Shipping	Grant	50	Shipping
Landry	50	Shipping	Hunold	60	IT
Markle	50	Shipping	Ernst	60	IT
Bissot	50	Shipping	Austin	60	IT
Atkinson	50	Shipping	Pataballa	60	IT
Marlow	50	Shipping	Lorentz	60	IT
Olson	50	Shipping	Baer	70	Public Relations
Mallin	50	Shipping	Russell	80	Sales
Rogers	50	Shipping	Partners	80	Sales
Gee	50	Shipping	Errazuriz	80	Sales
Philtanker	50	Shipping	Cambrault	80	Sales
Ladwig	50	Shipping	Zlotkey	80	Sales
Stiles	50	Shipping	Tucker	80	Sales
Seo	50	Shipping	Bernstein	80	Sales
Patel	50	Shipping	Hall	80	Sales
Rajs	50	Shipping	Olsen	80	Sales
Davies	50	Shipping	Cambrault	80	Sales
Matos	50	Shipping	Tuvault	80	Sales
Vargas	50	Shipping	King	80	Sales
Taylor	50	Shipping	Sully	80	Sales
Fleaur	50	Shipping	McEwen	80	Sales
Sullivan	50	Shipping	Smith	80	Sales

Doran	80	Sales
Sewall	80	Sales
Vishney	80	Sales
Greene	80	Sales
Marvins	80	Sales
Lee	80	Sales
Ande	80	Sales
Banda	80	Sales
Ozer	80	Sales
Bloom	80	Sales
Fox	80	Sales
Smith	80	Sales
Bates	80	Sales
Kumar	80	Sales
Abel	80	Sales
Hutton	80	Sales
Taylor	80	Sales
Livingston	80	Sales
Johnson	80	Sales
King	90	Executive
Kochhar	90	Executive
De Haan	90	Executive
Greenberg	100	Finance
Faviet	100	Finance
Chen	100	Finance
Sciarra	100	Finance
Urman	100	Finance
Popp	100	Finance
Higgins	110	Accounting
Gietz	110	Accounting

42. Create a unique list of all jobs that are in department 4. Include the location of the department in the output.

```
select distinct e.job_title, l.city as location
from employees e
join departments d on e.department_id = d.department_id
join locations l on d.location_id = l.location_id
where e.department_id = 4;
```

	job_id	location
--	--------	----------

43. Write a query to display the employee last name, department name, location id and city of all employees who earn commission.

```
select e.last_name, d.department_name, d.location_id, l.city
from employees e
join departments d on e.department_id = d.department_id
join locations l on d.location_id = l.location_id
where e.commission_pct is not null;
```

last_name	department_name	location_id	city
Russell	Sales	2500	Oxford
Partners	Sales	2500	Oxford
Errazuriz	Sales	2500	Oxford
Cambrault	Sales	2500	Oxford
Zlotkey	Sales	2500	Oxford
Tucker	Sales	2500	Oxford
Bernstein	Sales	2500	Oxford
Hall	Sales	2500	Oxford
Olsen	Sales	2500	Oxford
Cambrault	Sales	2500	Oxford
Tuvault	Sales	2500	Oxford
King	Sales	2500	Oxford
Sully	Sales	2500	Oxford
McEwen	Sales	2500	Oxford
Smith	Sales	2500	Oxford
Doran	Sales	2500	Oxford
Sewall	Sales	2500	Oxford
Vishney	Sales	2500	Oxford
Greene	Sales	2500	Oxford
Marvins	Sales	2500	Oxford
Lee	Sales	2500	Oxford
Ande	Sales	2500	Oxford
Banda	Sales	2500	Oxford
Ozer	Sales	2500	Oxford
Bloom	Sales	2500	Oxford
Fox	Sales	2500	Oxford
Smith	Sales	2500	Oxford
Bates	Sales	2500	Oxford
Kumar	Sales	2500	Oxford
Abel	Sales	2500	Oxford
Hutton	Sales	2500	Oxford
Taylor	Sales	2500	Oxford
Livingston	Sales	2500	Oxford
Johnson	Sales	2500	Oxford



44. Display the employee last name and department name of all employees who have an 'a' in their last name

last_name	department_name		
Kochhar	Executive		
De Haan	Executive		
Austin	IT		
Pataballa	IT		
Faviet	Finance		
Sciarra	Finance		
Urman	Finance		
Raphaely	Purchasing		
Baida	Purchasing		
Tobias	Purchasing		
Colmenares	Purchasing		
Kaufing	Shipping		
Vollman	Shipping		
Nayer	Shipping		
Landry	Shipping		
Markle	Shipping		
Atkinson	Shipping		
Marlow	Shipping		
Mallin	Shipping		
Philtanker	Shipping		
Ladwig	Shipping		
Patel	Shipping	Kumar	Sales
Rajs	Shipping	Abel	Sales
Davies	Shipping	Taylor	Sales
Matos	Shipping	Taylor	Shipping
Vargas	Shipping	Fleaur	Shipping
Partners	Sales	Sullivan	Shipping
Errazuriz	Sales	Sarchand	Shipping
Cambrault	Sales	Cabrio	Shipping
Hall	Sales	Gates	Shipping
Cambrault	Sales	McCain	Shipping
Tuvault	Sales	Walsh	Shipping
Doran	Sales	Grant	Shipping
Sewall	Sales	Whalen	Administration
Marvins	Sales	Hartstein	Marketing
Ande	Sales	Fay	Marketing
Banda	Sales	Mavris	Human Resources
Bates	Sales	Baer	Public Relations

45. Write a query to display the last name, job, department number and department name for all employees who work in ATLANTA.

```
select e.last_name, e.job_title, e.department_id, d.department_name
from employees e
join departments d on e.department_id = d.department_id
join locations l on d.location_id = l.location_id
where l.city = 'Atlanta';
```

	last_name	job_id	department_id	department_name
--	-----------	--------	---------------	-----------------

46. Display the employee last name and employee number along with their manager's last name and manager number.

```
select e.last_name as employee_last_name,
       e.employee_id as employee_number,
       m.last_name as manager_last_name,
       m.employee_id as manager_number
from employees e
left join employees m on e.manager_id = m.employee_id;
```

employee_last_name	employee_number	manager_last_name	manager_number
King	100	NULL	NULL
Kochhar	101	King	100
De Haan	102	King	100
Hunold	103	De Haan	102
Ernst	104	Hunold	103
Austin	105	Hunold	103
Pataballa	106	Hunold	103
Lorentz	107	Hunold	103
Greenberg	108	Kochhar	101
Faviet	109	Greenberg	108
Chen	110	Greenberg	108
Sciarra	111	Greenberg	108
Urman	112	Greenberg	108
Popp	113	Greenberg	108
Raphaely	114	King	100
Khoo	115	Raphaely	114
Baida	116	Raphaely	114
Tobias	117	Raphaely	114
Himuro	118	Raphaely	114
Colmenares	119	Raphaely	114
Weiss	120	King	100
Fripp	121	King	100
Kaufling	122	King	100
Vollman	123	King	100
Mourgos	124	King	100
Nayer	125	Weiss	120
Mikkilineni	126	Weiss	120
Landry	127	Weiss	120
Markle	128	Weiss	120
Bissot	129	Fripp	121
Atkinson	130	Fripp	121
Marlow	131	Fripp	121
Olson	132	Fripp	121
Mallin	133	Kaufling	122
Rogers	134	Kaufling	122

employee_last_name	employee_number	manager_last_name	manager_number
Gee	135	Kaufling	122
Philtanker	136	Kaufling	122
Ladwig	137	Vollman	123
Stiles	138	Vollman	123
Seo	139	Vollman	123
Patel	140	Vollman	123
Rajs	141	Mourgos	124
Davies	142	Mourgos	124
Matos	143	Mourgos	124
Vargas	144	Mourgos	124
Russell	145	King	100
Partners	146	King	100
Errazuriz	147	King	100
Cambrault	148	King	100
Zlotkey	149	King	100
Tucker	150	Russell	145
Bernstein	151	Russell	145
Hall	152	Russell	145
Olsen	153	Russell	145
Cambrault	154	Russell	145
Tuvault	155	Russell	145
King	156	Partners	146
Sully	157	Partners	146
McEwen	158	Partners	146
Smith	159	Partners	146
Doran	160	Partners	146
Sewall	161	Partners	146
Vishney	162	Errazuriz	147
Greene	163	Errazuriz	147
Marvins	164	Errazuriz	147
Lee	165	Errazuriz	147
Ande	166	Errazuriz	147
Banda	167	Errazuriz	147
Ozer	168	Cambrault	148
Bloom	169	Cambrault	148

employee_last_name	employee_number	manager_last_name	manager_number
Fox	170	Cambrault	148
Smith	171	Cambrault	148
Bates	172	Cambrault	148
Kumar	173	Cambrault	148
Abel	174	Zlotkey	149
Hutton	175	Zlotkey	149
Taylor	176	Zlotkey	149
Livingston	177	Zlotkey	149
Grant	178	Zlotkey	149
Johnson	179	Zlotkey	149
Taylor	180	Weiss	120
Fleaur	181	Weiss	120
Sullivan	182	Weiss	120
Geoni	183	Weiss	120
Sarchand	184	Fripp	121
Bull	185	Fripp	121
Dellinger	186	Fripp	121
Cabrio	187	Fripp	121
Chung	188	Kaufling	122
Dilly	189	Kaufling	122
Gates	190	Kaufling	122
Perkins	191	Kaufling	122
Bell	192	Vollman	123
Everett	193	Vollman	123
McCain	194	Vollman	123
Jones	195	Vollman	123
Walsh	196	Mourgos	124
Feeney	197	Mourgos	124
OConnell	198	Mourgos	124
Grant	199	Mourgos	124
Whalen	200	Kochhar	101
Hartstein	201	King	100
Fay	202	Hartstein	201
Mavris	203	Kochhar	101
Baer	204	Kochhar	101

47. Display the employee last name and employee number along with their manager's last name and manager number (including the employees who have no manager).

```
select e.last_name as employee_last_name,
       e.employee_id as employee_number,
       coalesce(m.last_name, 'No Manager') as manager_last_name,
       coalesce(m.employee_id, 'No Manager') as manager_number
from employees e
left join employees m on e.manager_id = m.employee_id;
```

employee_last_name	employee_number	manager_last_name	manager_number
King	100	NULL	NULL
Kochhar	101	King	100
De Haan	102	King	100
Hunold	103	De Haan	102
Ernst	104	Hunold	103
Austin	105	Hunold	103
Pataballa	106	Hunold	103
Lorentz	107	Hunold	103
Greenberg	108	Kochhar	101
Faviet	109	Greenberg	108
Chen	110	Greenberg	108
Sciarra	111	Greenberg	108
Urman	112	Greenberg	108
Popp	113	Greenberg	108
Raphaely	114	King	100
Khoo	115	Raphaely	114
Baida	116	Raphaely	114
Tobias	117	Raphaely	114
Himuro	118	Raphaely	114
Colmenares	119	Raphaely	114
Weiss	120	King	100
Fripp	121	King	100
Kaufling	122	King	100
Vollman	123	King	100
Mourgos	124	King	100
Nayer	125	Weiss	120
Mikkilineni	126	Weiss	120
Landry	127	Weiss	120
Markle	128	Weiss	120
Bissot	129	Fripp	121
Atkinson	130	Fripp	121
Marlow	131	Fripp	121
Olson	132	Fripp	121
Mallin	133	Kaufling	122
Rogers	134	Kaufling	122
Gee	135	Kaufling	122
Philtanker	136	Kaufling	122
Ladwig	137	Vollman	123

employee_last_name	employee_number	manager_last_name	manager_number
Stiles	138	Vollman	123
Seo	139	Vollman	123
Patel	140	Vollman	123
Rajs	141	Mourgos	124
Davies	142	Mourgos	124
Matos	143	Mourgos	124
Vargas	144	Mourgos	124
Russell	145	King	100
Partners	146	King	100
Errazuriz	147	King	100
Cambrault	148	King	100
Zlotkey	149	King	100
Tucker	150	Russell	145
Bernstein	151	Russell	145
Hall	152	Russell	145
Olsen	153	Russell	145
Cambrault	154	Russell	145
Tuvault	155	Russell	145
King	156	Partners	146
Sully	157	Partners	146
McEwen	158	Partners	146
Smith	159	Partners	146
Doran	160	Partners	146
Sewall	161	Partners	146
Vishney	162	Errazuriz	147
Greene	163	Errazuriz	147
Marvins	164	Errazuriz	147
Lee	165	Errazuriz	147
Ande	166	Errazuriz	147
Banda	167	Errazuriz	147
Ozer	168	Cambrault	148
Bloom	169	Cambrault	148
Fox	170	Cambrault	148
Smith	171	Cambrault	148
Bates	172	Cambrault	148
Kumar	173	Cambrault	148
Abel	174	Zlotkey	149
Hutton	175	Zlotkey	149

48. Create a query that displays employees last name, department number, and all the employees who work in the same department as a given employee.

```
select e1.last_name as employee_last_name,  
       e1.department_id,  
       e2.last_name as coworker_last_name  
from employees e1  
join employees e2 on e1.department_id = e2.department_id  
where e1.last_name = 'Given_Employee_LastName'  
and e1.employee_id <> e2.employee_id;
```

	employee_last_name	department_id	coworker_last_name
--	--------------------	---------------	--------------------



49. Create a query that displays the name, job, department name, salary, grade for all employees. Derive grade based on salary(>=50000=A, >=30000=B, <30000=C)

```
select e.first_name || ' ' || e.last_name as employee_name,
       e.job_title,
       d.department_name,
       e.salary,
       case
         when e.salary >= 50000 then 'A'
         when e.salary >= 30000 then 'B'
         else 'C'
       end as grade
from employees e
join departments d on e.department_id = d.department_id;
```

employee_name	job_id	department_name	salary	grade
0	AD_ASST	Administration	4400.00	C
0	MK_MAN	Marketing	13000.00	C
0	MK_REP	Marketing	6000.00	C
0	PU_MAN	Purchasing	11000.00	C
0	PU_CLERK	Purchasing	3100.00	C
0	PU_CLERK	Purchasing	2900.00	C
0	PU_CLERK	Purchasing	2800.00	C
0	PU_CLERK	Purchasing	2600.00	C
0	PU_CLERK	Purchasing	2500.00	C
0	HR_REP	Human Resources	6500.00	C
0	ST_MAN	Shipping	8000.00	C
0	ST_MAN	Shipping	8200.00	C
0	ST_MAN	Shipping	7900.00	C
0	ST_MAN	Shipping	6500.00	C
0	ST_MAN	Shipping	5800.00	C
0	ST_CLERK	Shipping	3200.00	C
0	ST_CLERK	Shipping	2700.00	C
0	ST_CLERK	Shipping	2400.00	C
0	ST_CLERK	Shipping	2200.00	C
0	ST_CLERK	Shipping	3300.00	C
0	ST_CLERK	Shipping	2800.00	C
0	ST_CLERK	Shipping	2500.00	C
0	ST_CLERK	Shipping	2100.00	C
0	ST_CLERK	Shipping	3300.00	C
0	ST_CLERK	Shipping	2900.00	C
0	ST_CLERK	Shipping	2400.00	C
0	ST_CLERK	Shipping	2200.00	C
0	ST_CLERK	Shipping	3600.00	C
0	ST_CLERK	Shipping	3200.00	C
0	ST_CLERK	Shipping	2700.00	C
0	ST_CLERK	Shipping	2500.00	C
0	ST_CLERK	Shipping	3500.00	C
0	ST_CLERK	Shipping	3100.00	C
0	ST_CLERK	Shipping	2600.00	C
0	ST_CLERK	Shipping	2500.00	C
0	SH_CLERK	Shipping	3200.00	C
0	SH_CLERK	Shipping	3100.00	C
0	SH_CLERK	Shipping	2500.00	C

employee_name	job_id	department_name	salary	grade
0	SH_CLERK	Shipping	2800.00	C
0	SH_CLERK	Shipping	4200.00	C
0	SH_CLERK	Shipping	4100.00	C
0	SH_CLERK	Shipping	3400.00	C
0	SH_CLERK	Shipping	3000.00	C
0	SH_CLERK	Shipping	3800.00	C
0	SH_CLERK	Shipping	3600.00	C
0	SH_CLERK	Shipping	2900.00	C
0	SH_CLERK	Shipping	2500.00	C
0	SH_CLERK	Shipping	4000.00	C
0	SH_CLERK	Shipping	3900.00	C
0	SH_CLERK	Shipping	3200.00	C
0	SH_CLERK	Shipping	2800.00	C
0	SH_CLERK	Shipping	3100.00	C
0	SH_CLERK	Shipping	3000.00	C
0	SH_CLERK	Shipping	2600.00	C
0	SH_CLERK	Shipping	2600.00	C
0	IT_PROG	IT	9000.00	C
0	IT_PROG	IT	6000.00	C
0	IT_PROG	IT	4800.00	C
0	IT_PROG	IT	4800.00	C
0	IT_PROG	IT	4200.00	C
0	PR_REP	Public Relations	10000.00	C
0	SA_MAN	Sales	14000.00	C
0	SA_MAN	Sales	13500.00	C
0	SA_MAN	Sales	12000.00	C
0	SA_MAN	Sales	11000.00	C
0	SA_MAN	Sales	10500.00	C
0	SA_REP	Sales	10000.00	C
0	SA_REP	Sales	9500.00	C
0	SA_REP	Sales	9000.00	C
0	SA_REP	Sales	8000.00	C
0	SA_REP	Sales	7500.00	C
0	SA_REP	Sales	7000.00	C
0	SA_REP	Sales	10000.00	C
0	SA_REP	Sales	9500.00	C
0	SA_REP	Sales	9000.00	C
0	SA_REP	Sales	8000.00	C

0	SA_REP	Sales	7500.00	C
0	SA_REP	Sales	7000.00	C
0	SA_REP	Sales	10500.00	C
0	SA_REP	Sales	9500.00	C
0	SA_REP	Sales	7200.00	C
0	SA_REP	Sales	6800.00	C
0	SA_REP	Sales	6400.00	C
0	SA_REP	Sales	6200.00	C
0	SA_REP	Sales	11500.00	C
0	SA_REP	Sales	10000.00	C
0	SA_REP	Sales	9600.00	C
0	SA_REP	Sales	7400.00	C
0	SA_REP	Sales	7300.00	C
0	SA_REP	Sales	6100.00	C
0	SA_REP	Sales	11000.00	C
0	SA_REP	Sales	8800.00	C
0	SA_REP	Sales	8600.00	C
0	SA_REP	Sales	8400.00	C
0	SA_REP	Sales	6200.00	C
0	AD_PRES	Executive	24000.00	C
0	AD_VP	Executive	17000.00	C
0	AD_VP	Executive	17000.00	C
0	FI_MGR	Finance	12000.00	C
0	FI_ACCO...	Finance	9000.00	C
0	FI_ACCO...	Finance	8200.00	C
0	FI_ACCO...	Finance	7700.00	C
0	FI_ACCO...	Finance	7800.00	C
0	FI_ACCO...	Finance	6900.00	C
0	AC_MGR	Accounting	12000.00	C
0	AC_ACC...	Accounting	8300.00	C

50. Display the names and hire date for all employees who were hired before their managers along with their manager names and hire date. Label the columns as Employee name, emp\_hire\_date, manager name, man\_hire\_date  
Database Name: AdventureWorks

```
SELECT
    e.first_name || ' ' || e.last_name AS employee_name,
    e.hire_date AS emp_hire_date,
    m.first_name || ' ' || m.last_name AS manager_name,
    m.hire_date AS man_hire_date
FROM employees e
JOIN employees m
    ON e.manager_id = m.employee_id
WHERE e.hire_date < m.hire_date;
```

employee_name	emp_hire_date	manager_name	man_hire_date
0	1990-01-03	0	1993-01-13
0	1994-08-16	0	1994-08-17
0	1997-02-16	0	1997-04-10
0	1995-07-14	0	1997-10-10
0	1995-10-17	0	1999-11-16
0	1997-01-29	0	1999-11-16
0	1998-03-15	0	1999-11-16
0	1998-07-09	0	1999-11-16
0	1996-01-30	0	1997-01-05
0	1996-03-04	0	1997-01-05
0	1996-08-01	0	1997-01-05
0	1997-03-11	0	1999-10-15
0	1998-03-23	0	1999-10-15
0	1998-01-24	0	1999-10-15
0	1999-02-23	0	1999-10-15
0	1999-03-24	0	1999-10-15
0	1996-05-11	0	2000-01-29
0	1997-03-19	0	2000-01-29
0	1998-03-24	0	2000-01-29
0	1998-04-23	0	2000-01-29
0	1999-05-24	0	2000-01-29
0	2000-01-04	0	2000-01-29
0	1996-01-27	0	1997-04-10
0	1997-02-20	0	1997-04-10
0	1996-02-04	0	1997-10-10
0	1997-03-03	0	1997-10-10
0	1998-04-24	0	1999-11-16
0	1998-05-23	0	1999-11-16
0	1999-06-21	0	1999-11-16
0	1987-09-17	0	1989-09-21

51. Write a query to display employee numbers and employee name (first name, last name) of all the sales employees who received an amount of 2000 in bonus.

```
select e.emp_idno, e.emp_fname, e.emp_lname  
from emp e  
join salesman s on e.emp_idno = s.salesman_id  
where s.commission = 2000;
```

	emp_idno	emp_fname	emp_lname
--	----------	-----------	-----------

52. Fetch address details of employees belonging to the state CA. If address is null, provide default value N/A.

```
select
  e.employee_id,
  e.first_name || ' ' || e.last_name as employee_name,
  coalesce(l.street_address, 'n/a') as street_address,
  coalesce(l.city, 'n/a') as city,
  coalesce(l.state_province, 'n/a') as state_province,
  coalesce(l.postal_code, 'n/a') as postal_code,
  coalesce(l.country_id, 'n/a') as country
from employees e
join departments d on e.department_id = d.department_id
join locations l on d.location_id = l.location_id
where l.state_province = 'CA';
```

	employee_id	employee_name	street_address	city	state_province	postal_code	country

53. Write a query that displays all the products along with the Sales OrderID even if an order has never been placed for that product.
54. Find the subcategories that have at least two different prices less than \$15.
55. A. Write a query to display employees and their manager details. Fetch employee id, employee first name, and manager id, manager name. B. Display the employee id and employee name of employees who do not have manager.
56. A. Display the names of all products of a particular subcategory 15 and the names of their vendors. B. Find the products that have more than one vendor.
57. Find all the customers who do not belong to any store.
58. Find sales prices of product 718 that are less than the list price recommended for that product.
59. Display product number, description and sales of each product in the year 2001.
60. Build the logic on the above question to extract sales for each category by year. Fetch Product Name, Sales\_2001, Sales\_2002, Sales\_2003. Hint: For questions 9 & 10 (From Sales.SalesOrderHeader, sales. SalesOrderDetail, Production. Product. Use ShipDate of SalesOrderHeader to extract shipped year. Calculate sales using QTY and unitprice from Sales OrderDetail.)
- Database Name: HR

61. Write a query to display the last name and hire date of any employee in the same department as SALES.

```
select e.last_name, e.hire_date
from employees e
where e.department_id = (
    select department_id
    from departments
    where department_name = 'SALES'
);
```

last_name	hire_date
Russell	1996-10-01
Partners	1997-01-05
Errazuriz	1997-03-10
Cambrault	1999-10-15
Zlotkey	2000-01-29
Tucker	1997-01-30
Bernstein	1997-03-24
Hall	1997-08-20
Olsen	1998-03-30
Cambrault	1998-12-09
Tuvault	1999-11-23
King	1996-01-30
Sully	1996-03-04
McEwen	1996-08-01
Smith	1997-03-10
Doran	1997-12-15
Sewall	1998-11-03
Vishney	1997-11-11
Greene	1999-03-19
Marvins	2000-01-24
Lee	2000-02-23
Ande	2000-03-24
Banda	2000-04-21
Ozer	1997-03-11
Bloom	1998-03-23
Fox	1998-01-24
Smith	1999-02-23
Bates	1999-03-24
Kumar	2000-04-21
Abel	1996-05-11
Hutton	1997-03-19
Taylor	1998-03-24
Livingston	1998-04-23
Johnson	2000-01-04



62. Create a query to display the employee numbers and last names of all employees who earn more than the average salary. Sort the results in ascending order of salary.

```
select employee_id, last_name, salary
from employees
where salary > (select avg(salary) from employees)
order by salary asc;
```

employee_id	last_name	salary			
203	Mavris	6500.00			
123	Vollman	6500.00			
165	Lee	6800.00			
113	Popp	6900.00			
155	Tuvault	7000.00			
161	Sewall	7000.00			
178	Grant	7000.00			
164	Marvins	7200.00			
172	Bates	7300.00			
171	Smith	7400.00			
160	Doran	7500.00			
154	Cambrault	7500.00			
111	Sciarra	7700.00			
112	Urman	7800.00			
122	Kaufing	7900.00			
159	Smith	8000.00			
153	Olsen	8000.00			
120	Weiss	8000.00			
110	Chen	8200.00			
121	Fripp	8200.00			
206	Gietz	8300.00			
177	Livingston	8400.00			
176	Taylor	8600.00			
175	Hutton	8800.00			
103	Hunold	9000.00	114	Raphaely	11000.00
109	Faviet	9000.00	148	Cambrault	11000.00
158	McEwen	9000.00	174	Abel	11000.00
152	Hall	9000.00	168	Ozer	11500.00
157	Sully	9500.00	108	Greenberg	12000.00
151	Bernstein	9500.00	147	Errazuriz	12000.00
163	Greene	9500.00	205	Higgins	12000.00
170	Fox	9600.00	201	Hartstein	13000.00
156	King	10000.00	146	Partners	13500.00
150	Tucker	10000.00	145	Russell	14000.00
169	Bloom	10000.00	101	Kochhar	17000.00
204	Baer	10000.00	102	De Haan	17000.00
149	Zlotkey	10500.00	100	King	24000.00
162	Vishney	10500.00	NULL	NULL	NULL

63. Write a query that displays the employee numbers and last names of all employees who work in a department with any employee whose last name contains a 'u'

```
select employee_id, last_name
from employees
where department_id in (
    select distinct department_id
    from employees
    where last_name like '%u%'
);
```

employee_id	last_name		
103	Hunold		
104	Ernst	141	Rajs
105	Austin	142	Davies
106	Pataballa	143	Matos
107	Lorentz	144	Vargas
108	Greenberg	180	Taylor
109	Faviet	181	Fleaur
110	Chen	182	Sullivan
111	Sciarra	183	Geoni
112	Urman	184	Sarchand
113	Popp	185	Bull
114	Raphaely	186	Dellinger
115	Khoo	187	Cabrio
116	Baida	188	Chung
117	Tobias	189	Dilly
118	Himuro	190	Gates
119	Colmenares	191	Perkins
120	Weiss	192	Bell
121	Fripp	193	Everett
122	Kaufing	194	McCain
123	Vollman	195	Jones
124	Mourgos	196	Walsh
125	Nayer	197	Feeney
126	Mikkilineni	198	OConnell
127	Landry	199	Grant
128	Markle	145	Russell
129	Bissot	146	Partners
130	Atkinson	147	Errazuriz
131	Marlow	148	Cambrault
132	Olson	149	Zlotkey
133	Mallin	150	Tucker
134	Rogers	151	Bernstein
135	Gee	152	Hall
136	Philtanker	153	Olsen
137	Ladwig	154	Cambrault
138	Stiles	155	Tuvault
139	Seo	156	King
140	Patel	157	Sully
		158	McEwen
		159	Smith
		160	Doran
		161	Sewall
		162	Vishney
		163	Greene
		164	Marvins
		165	Lee
		166	Ande
		167	Banda
		168	Ozer
		169	Bloom
		170	Fox
		171	Smith
		172	Bates
		173	Kumar
		174	Abel
		175	Hutton
		176	Taylor
		177	Livingston
		179	Johnson
		NULL	NULL

64. Display the last name, department number, and job ID of all employees whose department location is ATLANTA.

```
select e.last_name, e.department_id, e.job_id
from employees e
join departments d on e.department_id = d.department_id
join locations l on d.location_id = l.location_id
where l.city = 'ATLANTA';
```

	last_name	department_id	job_id

65. Display the last name and salary of every employee who reports to FILLMORE.

```
select e.department_id, e.last_name, e.job_id
from employees e
join departments d
on e.department_id = d.department_id
where d.department_name = 'OPERATIONS';
```

66. Display the department number, last name, and job ID for every employee in the OPERATIONS department.

```
select department_id, last_name, job_id
from employees
where department_id in (
    select department_id
    from departments
    where department_name = 'OPERATIONS'
);
```

67. Modify the above query to display the employee numbers, last names, and salaries of all employees who earn more than the average salary and who work in a department with any employee with a 'u' in their name.

```
select employee_id, last_name, salary
from employee
where salary > (select avg(salary) from employee)
and department_id in (
    select distinct department_id
    from employee
    where first_name like '%u%' or last_name like '%u%'
);
```

Description :

- The **first condition** (`salary > (select avg(salary) from employee)`) ensures that we only get employees who earn more than the average salary.
- The **second condition** filters employees who work in a department where at least one employee has a 'u' in their first or last name. This is done using a subquery that finds all such departments.

68. Display the names of all employees whose job title is the same as anyone in the sales dept.

```
select first_name, last_name
from employee
where job_id in (
    select distinct job_id
    from employee
    where department_id = (select department_id from department where
lower(department_name) = 'sales')
);
```

1. The **subquery** finds the `department_id` of the Sales department. Another **subquery** selects all `job_ids` assigned to employees in the Sales Department.
2. The **outer query** retrieves the names of employees who have any of these job titles.

69. Write a compound query to produce a list of employees showing raise percentages, employee IDs, and salaries. Employees in department 1 and 3 are given a 5% raise, employees in department 2 are given a 10% raise, employees in departments 4 and 5 are given a 15% raise, and employees in department 6 are not given a raise.

```
select employee_id, salary, 5 as raise_percentage  
from employee  
where department_id in (1, 3)
```

union all

```
select employee_id, salary, 10  
from employee  
where department_id = 2
```

union all

```
select employee_id, salary, 15  
from employee  
where department_id in (4, 5)
```

union all

```
select employee_id, salary, 0  
from employee  
where department_id = 6;
```

The query categorizes employees based on their `department_id` and assigns the corresponding raise percentage.

- **union all** is used instead of **union** to preserve duplicate salaries or IDs (if any).
- The raise percentage is hardcoded for each department group.

70. Write a query to display the top three earners in the EMPLOYEES table. Display their last names and salaries.

71. Display the names of all employees with their salary and commission earned. Employees with a null commission should have 0 in the commission column

1. Top Three Earners

```
select last_name, salary
from employee
order by salary desc
limit 3;
```

2. Employees with Salary and Commission Earned

```
select first_name, last_name, salary, coalesce(commission_pct, 0) as
commission_earned
from employee;
```

72. Display the Managers (name) with top three salaries along with their salaries and department information.

```
select e.first_name, e.last_name, e.salary, e.department_id
from employee e
where e.employee_id in (select distinct manager_id from employee where
manager_id is not null)
order by e.salary desc
limit 3;
```

Identifies Managers:

- The `where e.employee_id in (select distinct manager_id from employee where manager_id is not null)` condition ensures we only consider employees who are managers.

Orders by Salary:

- `order by e.salary desc` sorts employees by salary in descending order.

Limits to Top 3:

- `limit 3` ensures only the top three earning managers are displayed.

## Date Function

1) Find the date difference between the hire date and resignation\_date for all the employees. Display in no. of days, months and year(1 year 3 months 5 days).

Emp\_ID Hire Date Resignation\_Date  
1 1/1/2000 7/10/2013  
2 4/12/2003 3/8/2017  
3 22/9/2012 21/6/2015  
4 13/4/2015 NULL  
5 03/06/2016 NULL  
6 08/08/2017 NULL  
7 13/11/2016 NULL

```
select emp_id,
       hire_date,
       ifnull(resignation_date, current_date) as end_date,
       concat(
         timestampdiff(year, hire_date, ifnull(resignation_date, current_date)), '
years ',
         timestampdiff(month, hire_date, ifnull(resignation_date, current_date))
         - timestampdiff(year, hire_date, ifnull(resignation_date, current_date)) *
12, ' months ',
         datediff(ifnull(resignation_date, current_date),
         date_add(hire_date, interval timestampdiff(month, hire_date,
ifnull(resignation_date, current_date)) month)), ' days'
       ) as duration
from employees;
```

	emp_id	hire_date	end_date	duration
►	1	2000-01-01	2013-10-07	13 years 9 months 6 days
	2	2003-12-04	2017-08-03	13 years 7 months 30 days
	3	2012-09-22	2015-06-21	2 years 8 months 30 days
	4	2015-04-13	2025-02-28	9 years 10 months 15 days
	5	2016-06-03	2025-02-28	8 years 8 months 25 days
	6	2017-08-08	2025-02-28	7 years 6 months 20 days
	7	2016-11-13	2025-02-28	8 years 3 months 15 days

2) Format the hire date as mm/dd/yyyy(09/22/2003) and resignation\_date as mon dd, yyyy(Aug 12th, 2004). Display the null as (DEC, 01th 1900)

```
select emp_id,  
       date_format(hire_date, '%m/%d/%Y') as formatted_hire_date,  
  
       -- Formatting resignation date or replacing NULL with DEC 01th, 1900  
       case  
         when resignation_date is null then 'Dec 01th, 1900'  
         else concat(  
           date_format(resignation_date, '%b '),  
           day(resignation_date),  
           case  
             when day(resignation_date) in (1, 21, 31) then 'st'  
             when day(resignation_date) in (2, 22) then 'nd'  
             when day(resignation_date) in (3, 23) then 'rd'  
             else 'th'  
           end,  
           date_format(resignation_date, ', %Y')  
         )  
       end as formatted_resignation_date  
from employees;
```

	emp_id	formatted_hire_date	formatted_resignation_date
►	1	01/01/2000	Oct 7th, 2013
	2	12/04/2003	Aug 3rd, 2017
	3	09/22/2012	Jun 21st, 2015
	4	04/13/2015	Dec 01th, 1900
	5	06/03/2016	Dec 01th, 1900
	6	08/08/2017	Dec 01th, 1900
	7	11/13/2016	Dec 01th, 1900



3) Calculate experience of the employee till date in Years and months(example 1 year and 3 months) Use Getdate() as input date for the below three questions.

	emp_id	hire_date	resignation_date	years	months	experience
▶	1	2000-01-01	2013-10-07	13	9	13 year(s) and 9 month(s)
	2	2003-12-04	2017-08-03	13	8	13 year(s) and 8 month(s)
	3	2012-09-22	2015-06-21	2	9	2 year(s) and 9 month(s)
	4	2015-04-13	NULL	9	10	9 year(s) and 10 month(s)
	5	2016-06-03	NULL	8	8	8 year(s) and 8 month(s)
	6	2017-08-08	NULL	7	6	7 year(s) and 6 month(s)
	7	2016-11-13	NULL	8	3	8 year(s) and 3 month(s)

4) Display the count of days in the previous quarter

```
select datediff(
    case
        when quarter(curdate()) = 1 then last_day(date_sub(curdate(), interval 1
year))
        else last_day(date_sub(curdate(), interval quarter(curdate()) - 2 quarter))
    end,
    case
        when quarter(curdate()) = 1 then date_sub(last_day(date_sub(curdate(),
interval 1 year)), interval day(last_day(date_sub(curdate(), interval 1 year))) - 1 day)
        else date_sub(last_day(date_sub(curdate(), interval quarter(curdate()) - 2
quarter)), interval day(last_day(date_sub(curdate(), interval quarter(curdate()) - 2
quarter))) - 1 day)
    end
) + 1 as days_in_previous_quarter;
```

Identify the Previous Quarter:

- If the current quarter is **Q1**, the previous quarter is **Q4** of the previous year.
- Otherwise, the previous quarter is (current quarter - 1) of the same year.

Get the First and Last Day of the Previous Quarter:

- `last_day(date_sub(curdate(), interval quarter(curdate()) - 2 quarter))` → Gets the last day of the previous quarter.
- `date_sub(last_day(...), interval day(last_day(...)) - 1 day)` → Gets the first day of the previous quarter.

Calculate the Difference:

- `datediff(last_day_of_previous_quarter, first_day_of_previous_quarter) + 1` → Gives the total number of days in that quarter.

5) Fetch the previous Quarter's second week's first day's date

```
select date_add(
  case
    when quarter(curdate()) = 1 then date_sub(curdate(), interval 1 year)
    else date_sub(curdate(), interval quarter(curdate()) - 2 quarter)
  end,
  interval 7 day
) as second_week_first_day;
```

Find the Previous Quarter's Start Date:

- If the current quarter is **Q1**, the previous quarter is **Q4 of the previous year**.
- Otherwise, subtract **(quarter - 2)** quarters from the current date.

Add 7 Days to Get the Second Week's First Day:

- The first week starts on the first day of the quarter.
- Adding **7 days** gives the first day of the second week.

6) Fetch the financial year's 15th week's dates (Format: Mon DD YYYY)

select

```
date_format(date_add(str_to_date(concat(year(curdate()), '-04-01'), '%Y-%m-%d'),
interval 14 week), '%b %d %Y') as start_date,
date_format(date_add(str_to_date(concat(year(curdate()), '-04-01'), '%Y-%m-%d'),
interval 14 week + 6 day), '%b %d %Y') as end_date;
```

Assumes the financial year starts on April 1st.

Finds the start date of the 15th week:

- Week 1 starts from April 1st.
- Week 15 starts after 14 full weeks → **interval 14 week**.

Finds the end date:

- Adds 6 days to the start date.

Formats the output as **Mon DD YYYY** (e.g., Apr 08 2024)

7) Find out the date that corresponds to the last Saturday of January, 2015 using with clause. Use Airport database for the below two question:

with january\_dates as (

```
select str_to_date('2015-01-31', '%Y-%m-%d') as last_day
```

)

```
select date_sub(last_day, interval weekday(last_day) + 1 day) as last_saturday
from january_dates;
```

**with january\_dates:**

- Defines a CTE (Common Table Expression) with the last day of January 2015 (**2015-01-31**).

Calculates the last Saturday:

- **weekday(last\_day)**: Returns the index of the weekday (Monday = 0, Sunday = 6).
- **interval weekday(last\_day) + 1 day**: Subtracts the necessary days to land on Saturday.

8) Find the number of days elapsed between first and last flights of a passenger.

```
select passenger_id,  
       datediff(max(flight_date), min(flight_date)) as days_elapsed  
from flights  
group by passenger_id;
```

Finds the first and last flight dates of each passenger:

- `min(flight_date)`: Gets the earliest flight date (first flight).
- `max(flight_date)`: Gets the latest flight date (last flight).

Calculates the difference in days using `datediff()`.

Groups by `passenger_id` to calculate the elapsed days for each passenger.

9) Find the total duration in minutes and in seconds of the flight from Rostov to Moscow.

```
select flight_id,  
       timediff(arrival_time, departure_time) as duration_hhmmss,  
       timestampdiff(minute, departure_time, arrival_time) as duration_minutes,  
       timestampdiff(second, departure_time, arrival_time) as duration_seconds  
from flights  
where source_airport = 'Rostov' and destination_airport = 'Moscow';
```

Filters flights where `source_airport = 'Rostov'` and `destination_airport = 'Moscow'`.

Calculates the flight duration in different formats:

- `timediff(arrival_time, departure_time)`: Returns HH:MM:SS format.
- `timestampdiff(minute, departure_time, arrival_time)`: Returns total duration in minutes.
- `timestampdiff(second, departure_time, arrival_time)`: Returns total duration in seconds.