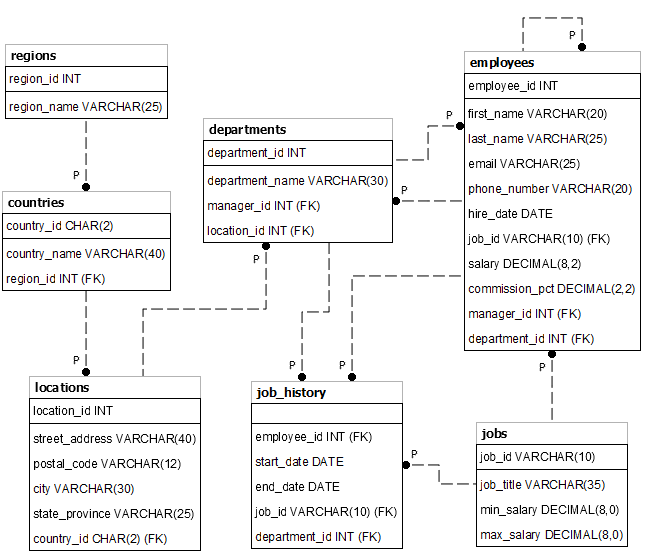
1. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) қызметкердің толық аты-жөнін , идентификаторын және қазіргі уақытта жұмыс істейтін елдің атын қайтаратын SQL сұрауын жазыңыз.



Сурет - 1. HR – диаграммасы

SELECT employees.first\_name || ' ' || employees.last\_name AS full\_name,

employees.employee\_id,

countries.country\_name

FROM employees

JOIN job\_history ON employees.employee\_id = job\_history.employee\_id

JOIN departments ON job\_history.department\_id = departments.department\_id

JOIN locations ON departments.location\_id = locations.location\_id

JOIN countries ON locations.country\_id = countries.country\_id

WHERE job\_history.end\_date IS NULL;

 employees.first\_name || ' ' || employees.last\_name AS full\_name: Қызметкердің толық аты-жөні.

 employees.employee\_id: Қызметкердің идентификаторы.

 countries.country\_name: Қызметкер жұмыс істейтін елдің аты.

 JOIN job\_history ON employees.employee\_id = job\_history.employee\_id: Қызметкерлер мен жұмыс тарихы кестелерін employee\_id арқылы байланыстырады.

 JOIN departments ON job\_history.department\_id = departments.department\_id: Жұмыс тарихы мен бөлімдер кестелерін department\_id арқылы байланыстырады.

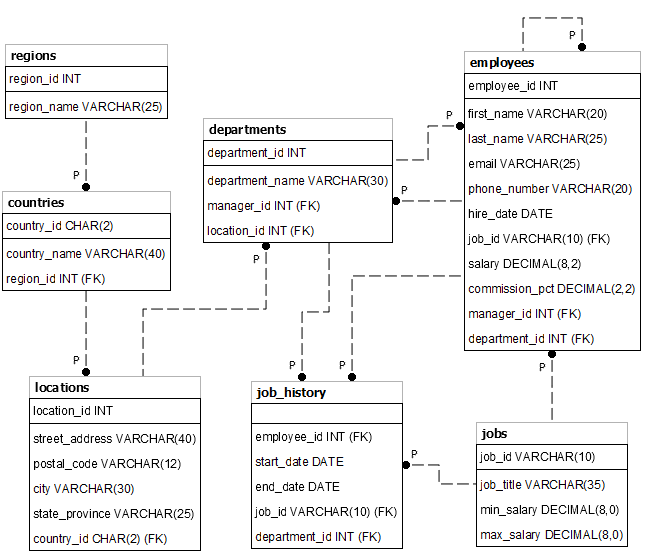
 JOIN locations ON departments.location\_id = locations.location\_id: Бөлімдер мен орындар кестелерін location\_id арқылы байланыстырады.

 JOIN countries ON locations.country\_id = countries.country\_id: Орындар мен елдер кестелерін country\_id арқылы байланыстырады.

 WHERE job\_history.end\_date IS NULL: Қызметкердің қазіргі уақытта жұмыс істейтін орнын алу үшін, аяқталу күні жоқ жазбаларды алады.

2. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) бөлімнің атауы, орташа жалақысы және осы бөлімде жұмыс істейтін және комиссия алатын қызметкерлер саны туралы ақпаратты шығару үшін SQL сұрауын жазыңыз.



Сурет - 1. HR – диаграммасы

SELECT departments.department\_name,

AVG(employees.salary) AS average\_salary,

COUNT(employees.employee\_id) AS num\_employees\_with\_commission

FROM departments

JOIN employees ON departments.department\_id = employees.department\_id

WHERE employees.commission\_pct IS NOT NULL

GROUP BY departments.department\_name;

 departments.department\_name: Бөлімнің атауы.

 AVG(employees.salary) AS average\_salary: Бөлімдегі қызметкерлердің орташа жалақысын есептейді.

 COUNT(employees.employee\_id) AS num\_employees\_with\_commission: Комиссия алатын қызметкерлер санын есептейді.

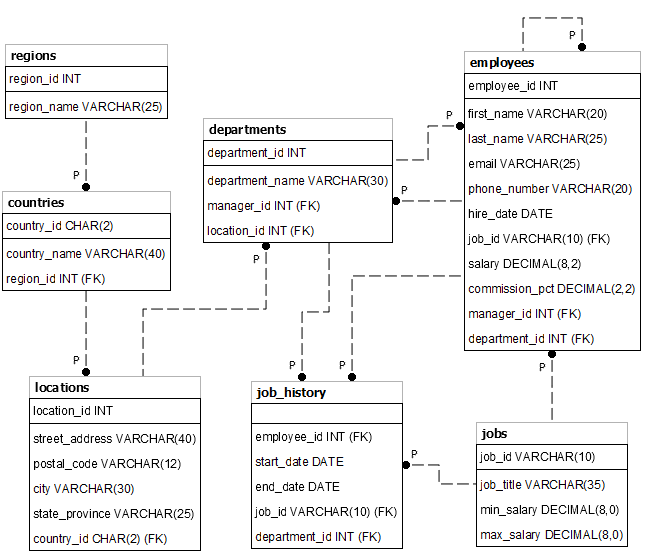
 JOIN employees ON departments.department\_id = employees.department\_id: Бөлімдер мен қызметкерлер кестелерін department\_id арқылы байланыстырады.

 WHERE employees.commission\_pct IS NOT NULL: Комиссия алатын қызметкерлерді ғана таңдайды.

 GROUP BY departments.department\_name: Нәтижелерді бөлімнің аты бойынша топтастырады.

3. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) Лондонда жұмыс істейтін барлық қызметкерлердің атын, тегін және жалақысын алу үшін SQL сұрауын жазыңыз.



Сурет - 1. HR – диаграммасы

SELECT employees.first\_name,

employees.last\_name,

employees.salary

FROM employees

JOIN departments ON employees.department\_id = departments.department\_id

JOIN locations ON departments.location\_id = locations.location\_id

WHERE locations.city = 'London';

 employees.first\_name: Қызметкердің аты.

 employees.last\_name: Қызметкердің тегі.

 employees.salary: Қызметкердің жалақысы.

 JOIN departments ON employees.department\_id = departments.department\_id: Қызметкерлер мен бөлімдер кестелерін department\_id арқылы байланыстырады.

 JOIN locations ON departments.location\_id = locations.location\_id: Бөлімдер мен орындар кестелерін location\_id арқылы байланыстырады.

 WHERE locations.city = 'London': Қызметкерлердің жұмыс істейтін қаласы Лондон болуын тексереді.

4. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) идентификаторларды, жұмыс атауын және 80 идентификаторы бар бөлімде жұмыс істеген күндер санын қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT employees.employee\_id,

jobs.job\_title,

job\_history.end\_date - job\_history.start\_date AS days\_worked

FROM employees

JOIN job\_history ON employees.employee\_id = job\_history.employee\_id

JOIN jobs ON job\_history.job\_id = jobs.job\_id

WHERE job\_history.department\_id = 80;

 employees.employee\_id: Қызметкердің идентификаторы.

 jobs.job\_title: Қызметкердің жұмыс атауы.

 job\_history.end\_date - job\_history.start\_date AS days\_worked: Қызметкердің бөлімде жұмыс істеген күндер саны.

 JOIN job\_history ON employees.employee\_id = job\_history.employee\_id: Қызметкерлер мен жұмыс тарихы кестелерін employee\_id арқылы байланыстырады.

 JOIN jobs ON job\_history.job\_id = jobs.job\_id: Жұмыс тарихы мен жұмыс кестелерін job\_id арқылы байланыстырады.

 WHERE job\_history.department\_id = 80: 80 идентификаторы бар бөлімде жұмыс істеген жазбаларды таңдайды.

5. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) бөлім атауын, атын, тегін және менеджер қаласын қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT departments.department\_name,

employees.first\_name,

employees.last\_name,

manager\_locations.city AS manager\_city

FROM employees

JOIN departments ON employees.department\_id = departments.department\_id

JOIN employees AS managers ON departments.manager\_id = managers.employee\_id

JOIN locations AS manager\_locations ON managers.location\_id = manager\_locations.location\_id;

 departments.department\_name: Бөлім атауы.

 employees.first\_name: Қызметкердің аты.

 employees.last\_name: Қызметкердің тегі.

 manager\_locations.city AS manager\_city: Менеджердің қаласы.

 JOIN departments ON employees.department\_id = departments.department\_id: Қызметкерлер мен бөлімдер кестелерін department\_id арқылы байланыстырады.

 JOIN employees AS managers ON departments.manager\_id = managers.employee\_id: Бөлімдер мен менеджерлер кестелерін manager\_id арқылы байланыстырады. Бұл жерде employees кестесі менеджерлерді алу үшін managers деген псевдониммен қолданылған.

 JOIN locations AS manager\_locations ON managers.location\_id = manager\_locations.location\_id: Менеджерлер мен орындар кестелерін location\_id арқылы байланыстырады. Бұл жерде locations кестесі менеджерлердің орындарын алу үшін manager\_locations деген псевдониммен қолданылған.

6. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) 1800 доллар немесе одан да көп жалақы алатын қызметкерлерді тізімдеу үшін SQL сұрауын жазыңыз. Қызметкердің идентификаторын, басталу күнін, аяқталу күнін, жұмыс идентификаторын және бөлім идентификаторын қайтарады.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT employees.employee\_id,

job\_history.start\_date,

job\_history.end\_date,

job\_history.job\_id,

job\_history.department\_id

FROM employees

JOIN job\_history ON employees.employee\_id = job\_history.employee\_id

WHERE employees.salary >= 1800;

 employees.employee\_id: Қызметкердің идентификаторы.

 job\_history.start\_date: Жұмыс басталу күні.

 job\_history.end\_date: Жұмыс аяқталу күні.

 job\_history.job\_id: Жұмыс идентификаторы.

 job\_history.department\_id: Бөлім идентификаторы.

 JOIN job\_history ON employees.employee\_id = job\_history.employee\_id: Қызметкерлер мен жұмыс тарихы кестелерін employee\_id арқылы байланыстырады.

 WHERE employees.salary >= 1800: Жалақысы 1800 доллар немесе одан көп қызметкерлерді таңдайды.

7. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) әр лауазым бойынша қызметкерлердің орташа жалақысын есептеу үшін SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT jobs.job\_title,

AVG(employees.salary) AS average\_salary

FROM employees

JOIN jobs ON employees.job\_id = jobs.job\_id

GROUP BY jobs.job\_title;

 jobs.job\_title: Лауазым атауы.

 AVG(employees.salary) AS average\_salary: Әр лауазым бойынша қызметкерлердің орташа жалақысын есептейді.

 JOIN jobs ON employees.job\_id = jobs.job\_id: Қызметкерлер мен лауазымдар кестелерін job\_id арқылы байланыстырады.

 GROUP BY jobs.job\_title: Нәтижелерді лауазым атауы бойынша топтастырады.

8. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) 110 идентификаторы бар бөлімде жұмыс істейтін барлық қызметкерлердің жалақысы мен жалақысының арасындағы айырмашылықты есептеу үшін SQL сұрауын жазыңыз. Лауазымның атын, қызметкердің атын және жалақы айырмашылығын қайтарады.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT jobs.job\_title,

employees.first\_name,

employees.salary - dept\_avg\_salaries.avg\_salary AS salary\_difference

FROM employees

JOIN jobs ON employees.job\_id = jobs.job\_id

JOIN (SELECT department\_id, AVG(salary) AS avg\_salary

FROM employees

WHERE department\_id = 110

GROUP BY department\_id) AS dept\_avg\_salaries

ON employees.department\_id = dept\_avg\_salaries.department\_id

WHERE employees.department\_id = 110;

 jobs.job\_title: Лауазым атауы.

 employees.first\_name: Қызметкердің аты.

 employees.salary - dept\_avg\_salaries.avg\_salary AS salary\_difference: Қызметкердің жалақысы мен орташа жалақының арасындағы айырмашылық.

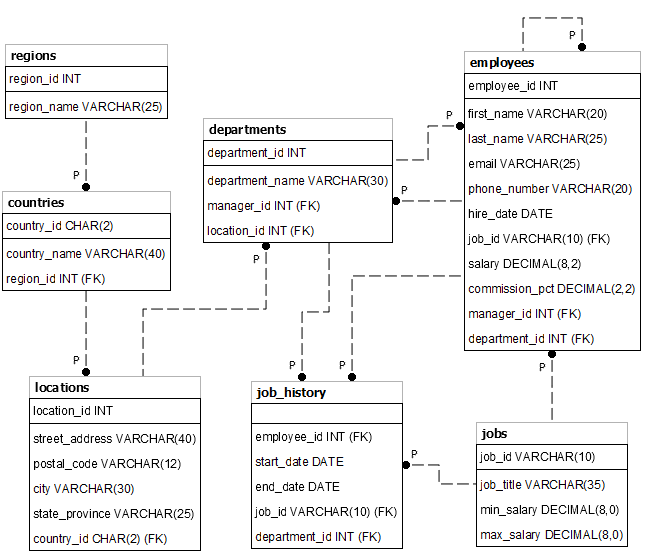
 JOIN jobs ON employees.job\_id = jobs.job\_id: Қызметкерлер мен лауазымдар кестелерін job\_id арқылы байланыстырады.

 JOIN (SELECT department\_id, AVG(salary) AS avg\_salary FROM employees WHERE department\_id = 110 GROUP BY department\_id) AS dept\_avg\_salaries ON employees.department\_id = dept\_avg\_salaries.department\_id: 110 идентификаторы бар бөлімдегі орташа жалақыны есептейтін ішкі сұрауды қосады.

 WHERE employees.department\_id = 110: 110 идентификаторы бар бөлімдегі қызметкерлерді таңдайды.

9. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) 124 идентификаторы бар қызметкерге қарағанда жоғары жалақы алатын қызметкерлерді (тегі, аты) тізімдеу үшін SQL сұрауын жазыңыз.



Сурет - 1. HR – диаграммасы

SELECT e.first\_name, e.last\_name

FROM employees e

WHERE e.salary > (

SELECT salary

FROM employees

WHERE employee\_id = 124

);

 e.first\_name: Қызметкердің аты.

 e.last\_name: Қызметкердің тегі.

 WHERE e.salary > (SELECT salary FROM employees WHERE employee\_id = 124): Ішкі сұрау арқылы 124 идентификаторы бар қызметкердің жалақысын алады және сол жалақыдан жоғары жалақы алатын қызметкерлерді таңдайды.

10. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) қызметкерлердің саны 20-дан асатын бөлімдердің аттарын, идентификаторларын қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT departments.department\_name, departments.department\_id

FROM departments

JOIN (

SELECT department\_id, COUNT(\*) AS num\_employees

FROM employees

GROUP BY department\_id

HAVING num\_employees > 20

) AS emp\_counts ON departments.department\_id = emp\_counts.department\_id;

 departments.department\_name: Бөлімнің атауы.

 departments.department\_id: Бөлімнің идентификаторы.

 JOIN (SELECT department\_id, COUNT(\*) AS num\_employees FROM employees GROUP BY department\_id HAVING num\_employees > 20) AS emp\_counts ON departments.department\_id = emp\_counts.department\_id: 20-дан асатын қызметкерлердің барлық бөлімдерге тікелей қосымша сұрауды қосады.

11. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) елдің, қаланың атауын және кемінде 2 қызметкері бар бөлімдердің санын қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT locations.country\_name, locations.city, COUNT(departments.department\_id) AS num\_departments

FROM locations

JOIN departments ON locations.location\_id = departments.location\_id

GROUP BY locations.country\_name, locations.city

HAVING COUNT(departments.department\_id) >= 2;

1. locations.country\_name: Елдің атауы.
2. locations.city: Қаланың атауы.
3. COUNT(departments.department\_id) AS num\_departments: Бөлімдердің санын есептеу.
4. JOIN departments ON locations.location\_id = departments.location\_id: Мекенжай кестесі мен бөлімдер кестесінің location\_id арқылы қосымшаға байланыстырады.
5. GROUP BY locations.country\_name, locations.city: Нәтижелерді елдің атауы мен қаланың атауы бойынша топтастырады.
6. HAVING COUNT(departments.department\_id) >= 2: Бөлімдердің саны 2-ден көп болуын тексереді.

12. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) Hunold тегі бар менеджерге есеп беретін қызметкерлердің аттарын, фамилияларын, идентификаторларын және жалақысын қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT employees.first\_name, employees.last\_name, employees.employee\_id, employees.salary

FROM employees

JOIN employees AS managers ON employees.manager\_id = managers.employee\_id

WHERE managers.last\_name = 'Hunold';

 employees.first\_name: Қызметкердің аты.

 employees.last\_name: Қызметкердің тегі.

 employees.employee\_id: Қызметкердің идентификаторы.

 employees.salary: Қызметкердің жалақысы.

 JOIN employees AS managers ON employees.manager\_id = managers.employee\_id: Қызметкерлер мен менеджерлер кестелерін байланыстырады. Бұл жерде менеджерге қосымша сұрау қолданылған.

 WHERE managers.last\_name = 'Hunold': Тек Hunold тегі бар менеджерге есеп беретін қызметкерлерді таңдайды.

13. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) жалақысы ең төменгі жалақыдан 2500-ге дейінгі қызметкерлер туралы толық ақпаратты алу үшін SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT \*

FROM employees

WHERE salary BETWEEN (SELECT MIN(salary) FROM employees) AND 2500;

 \*: Барлық қызметкерлердің барлық ақпараттарын алу үшін.

 WHERE salary BETWEEN (SELECT MIN(salary) FROM employees) AND 2500: Жалақысы ең төменгі жалақыдан 2500-ге дейінгі қызметкерлерді таңдайды.

14. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) "Hall"бөлімінде жұмыс істейтін қызметкерлердің аттарын, фамилияларын және жалдау күнін қайтаратын SQL сұрауын жазыңыз. "Hall"тегі бар жазбаларды алып тастаңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT first\_name, last\_name, hire\_date

FROM employees

WHERE department\_id = (

SELECT department\_id

FROM departments

WHERE department\_name = 'Hall'

);

 first\_name: Қызметкердің аты.

 last\_name: Қызметкердің тегі.

 hire\_date: Жұмысқа орналасу күні.

 WHERE department\_id = (SELECT department\_id FROM departments WHERE department\_name = 'Hall'): "Hall" бөлімінің department\_id-сін анықтап, осы идентификаторға ие болатын қызметкерлерді таңдайды.

15. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) атауы "S"белгісінен басталатын бөлімде жұмыс істейтін қызметкерлердің тізімін қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT \*

FROM employees

WHERE department\_id IN (

SELECT department\_id

FROM departments

WHERE department\_name LIKE 'S%'

);

 \*: Барлық қызметкерлердің барлық ақпараттарын алу үшін.

 WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE 'S%'): "S" белгісінен басталатын бөлімдердің department\_id-лерін таңдай алуға болады.

16. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) "SH\_CLERK"лауазымында жұмыс істейтін қызметкерлерге қарағанда жалақысы төмен қызметкерлер туралы ақпаратты шығару үшін SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT \*

FROM employees

WHERE job\_id = 'SH\_CLERK'

AND salary < (

SELECT MIN(salary)

FROM employees

WHERE job\_id = 'SH\_CLERK'

);

 \*: Барлық қызметкерлердің барлық ақпараттарын алу үшін.

 WHERE job\_id = 'SH\_CLERK': "SH\_CLERK" лауазымында жұмыс істейтін қызметкерлерді таңдайды.

 AND salary < (SELECT MIN(salary) FROM employees WHERE job\_id = 'SH\_CLERK'): Жалақысы төмен қызметкерлер туралы ақпаратты шығару үшін, осы лауазымдың құрамына кірісті жұмысшылардың жалақыларын салыстыруға болады.

17. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) барлық бөлімдер бойынша жалақысы орташа деңгейден жоғары қызметкерлердің тізімін қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT employees.first\_name, employees.last\_name, AVG(employees.salary) AS average\_salary

FROM employees

JOIN departments ON employees.department\_id = departments.department\_id

GROUP BY departments.department\_id

HAVING employees.salary > AVG(employees.salary);

 employees.first\_name: Қызметкердің аты.

 employees.last\_name: Қызметкердің тегі.

 AVG(employees.salary) AS average\_salary: Бөлім бойынша орташа жалақы деңгейін есептеу.

 JOIN departments ON employees.department\_id = departments.department\_id: Қызметкерлер мен бөлімдер кестелерін байланыстырады.

 GROUP BY departments.department\_id: Бөлім бойынша топтау.

 HAVING employees.salary > AVG(employees.salary): Жалақысы орташа деңгейден жоғары қызметкерлерді таңдау.

18. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) орташа жалақыны және осы бөлімде комиссионные алатын қызметкерлердің санын есептеу үшін SQL сұрауын жазыңыз. Бөлімнің атын, орташа жалақысын және қызметкерлер санын қайтарады.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT departments.department\_name, AVG(employees.salary) AS average\_salary, COUNT(\*) AS commission\_workers\_count

FROM employees

JOIN departments ON employees.department\_id = departments.department\_id

WHERE employees.commission\_pct IS NOT NULL

GROUP BY departments.department\_name;

 departments.department\_name: Бөлімнің атауы.

 AVG(employees.salary) AS average\_salary: Орташа жалақы.

 COUNT(\*) AS commission\_workers\_count: Бөлімде комиссионные алатын қызметкерлер саны.

 JOIN departments ON employees.department\_id = departments.department\_id: Қызметкерлер мен бөлімдер кестелерін байланыстырады.

 WHERE employees.commission\_pct IS NOT NULL: Комиссиялық табыс бар қызметкерлерді таңдайды.

 GROUP BY departments.department\_name: Бөлім бойынша топтау.

19. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) 1993 жылдың 1 қаңтарында немесе осы күннен кейін және 1997 жылдың 31 тамызында немесе одан бұрын жұмысқа қабылданған қызметкерлердің тізімін қайтаратын SQL сұрауын жазыңыз. Лауазымның атын, бөлімнің атын, қызметкердің атын, тегін және қызметке кіріскен күнін қайтарады.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT job\_history.employee\_id, employees.first\_name, employees.last\_name, job\_history.start\_date

FROM job\_history

JOIN employees ON job\_history.employee\_id = employees.employee\_id

WHERE job\_history.start\_date >= '1993-01-01' AND job\_history.start\_date <= '1997-07-31';

 job\_history.employee\_id: Қызметкердің идентификаторы.

 employees.first\_name: Қызметкердің аты.

 employees.last\_name: Қызметкердің тегі.

 job\_history.start\_date: Жұмысқа қабылдану күні.

 JOIN employees ON job\_history.employee\_id = employees.employee\_id: Қызметкерлер мен жұмыс тарихы кестелерін байланыстырады.

 WHERE job\_history.start\_date >= '1993-01-01' AND job\_history.start\_date <= '1997-07-31': Сұрау жасалған күндер арасында болатын жұмысқа қабылдану күндерін тексереді.

20. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) бағыныстағы алты қызметкерден көп менеджерлерді тізімдеу үшін SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT m.employee\_id, m.first\_name, m.last\_name, COUNT(e.employee\_id) AS number\_of\_subordinates

FROM employees m

JOIN employees e ON m.employee\_id = e.manager\_id

GROUP BY m.employee\_id, m.first\_name, m.last\_name

HAVING COUNT(e.employee\_id) > 6;

 employees кестесінен m (менеджерлер) және e (бағыныстылар) деген алиас пайдаланып, кестелерді өз-өзімен біріктіреді. Мұнда менеджер employee\_id - m алиасы, ал бағыныстының manager\_id - e алиасы.

 GROUP BY клаузасында менеджерлердің employee\_id, first\_name, және last\_name бойынша топтастырады.

 HAVING клаузасында бағыныстылардың саны алтыдан көп менеджерлерді ғана таңдайды.

21. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) ең жоғары жалақы 10000-нан асатын бөлімдердің тізімін қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT DISTINCT d.department\_id, d.department\_name

FROM departments d

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

WHERE e.salary > 10000;

 departments кестесінен d алиасын және employees кестесінен e алиасын пайдаланып, кестелерді department\_id бойынша біріктіреді.

 WHERE клаузасында salary 10000-нан асатын қызметкерлерді ғана таңдайды.

 DISTINCT кілт сөзін пайдаланып, қайталануларды жояды және тек бөлімдердің айрықша тізімін қайтарады.

22. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) қызметкерлері жоқ бөлімдерді қоса алғанда, бөлімдердің тізімін қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT d.department\_id, d.department\_name, e.employee\_id

FROM departments d

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

GROUP BY d.department\_id, d.department\_name

ORDER BY d.department\_id;

 departments кестесінен d алиасын және employees кестесінен e алиасын пайдаланып, кестелерді LEFT JOIN арқылы біріктіреді. Бұл оператор барлық бөлімдерді қосады, тіпті олардың ешқандай қызметкері болмаса да.

 GROUP BY клаузасында бөлімдердің department\_id және department\_name бағандары бойынша топтастырады.

 ORDER BY клаузасында нәтижелерді department\_id бойынша реттейді.

23. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) 100-бөлімдегі кез келген адамнан көп жалақы алатын 80 – бөлімдегі

қызметкерлерінің тізімін алу үшін SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT e80.employee\_id, e80.first\_name, e80.last\_name, e80.salary

FROM employees e80

WHERE e80.department\_id = 80

AND e80.salary > (

SELECT MAX(e100.salary)

FROM employees e100

WHERE e100.department\_id = 100

);

 employees кестесінен e80 алиасын пайдаланып, 80-бөлімдегі қызметкерлерді таңдайды.

 WHERE клаузасында department\_id 80-ге тең екеніне көз жеткізеді.

 AND клаузасында ішкі сұрау (subquery) арқылы 100-бөлімдегі ең жоғары жалақыны (MAX) алады.

 80-бөлімдегі қызметкерлердің жалақысын осы ішкі сұрауда алынған ең жоғары жалақымен салыстырады және осыдан көп жалақы алатын қызметкерлерді таңдайды.

24. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) идентификаторы 105 болатын қызметкердің қала атауын қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

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.employee\_id = 105;

 employees кестесінен e алиасын пайдаланып, 105 идентификаторы бар қызметкерді таңдайды.

 departments кестесінен d алиасын пайдаланып, қызметкердің бөлімін анықтайды (department\_id бойынша қосылады).

 locations кестесінен l алиасын пайдаланып, бөлімнің орналасқан жерін анықтайды (location\_id бойынша қосылады).

 Нәтиже ретінде сол орналасқан жердің city бағанының мәнін қайтарады.

25. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) бұрын ИТ- бағдарламашылары (IT\_PROG) болып жұмыс істеген қызметкерлер үшін ағымдағы жұмыс туралы ақпаратты шығару үшін SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT e.employee\_id, e.first\_name, e.last\_name, e.job\_id, j.job\_title, e.department\_id, e.salary

FROM employees e

JOIN job\_history jh ON e.employee\_id = jh.employee\_id

JOIN jobs j ON e.job\_id = j.job\_id

WHERE jh.job\_id = 'IT\_PROG';

 employees кестесінен e алиасын пайдаланып, қызметкерлерді таңдайды.

 job\_history кестесінен jh алиасын пайдаланып, employee\_id бойынша қызметкерлерді біріктіреді және бұрын ИТ-бағдарламашысы болып жұмыс істегендерін анықтайды.

 jobs кестесінен j алиасын пайдаланып, қызметкерлердің ағымдағы жұмысын анықтайды (job\_id бойынша қосылады).

 WHERE клаузасында бұрын ИТ-бағдарламашысы (IT\_PROG) болып жұмыс істеген қызметкерлерді ғана таңдайды.

26. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

а) 80 немесе 40 бөлімдерде жұмыс істейтін барлық қызметкерлер үшін бөлімнің атын, тегін, нөмірін және атын қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT d.department\_name, e.last\_name, e.phone\_number, e.first\_name

FROM employees e

JOIN departments d ON e.department\_id = d.department\_id

WHERE e.department\_id IN (80, 40);

 employees кестесінен e алиасын пайдаланып, қызметкерлерді таңдайды.

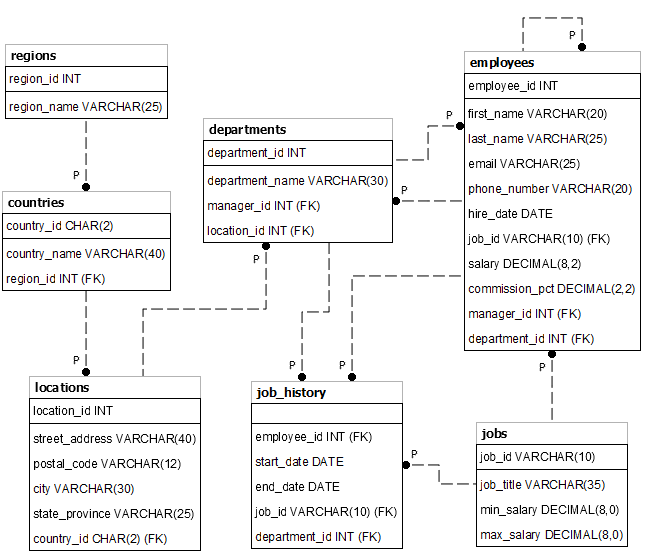
 departments кестесінен d алиасын пайдаланып, қызметкерлердің бөлімін анықтайды (department\_id бойынша қосылады).

 WHERE клаузасында қызметкерлердің department\_id мәндері 80 немесе 40-қа тең екендігіне көз жеткізеді.

 Нәтиже ретінде бөлімнің атауы (department\_name), қызметкердің тегі (last\_name), нөмірі (phone\_number), және аты (first\_name) алынады.

27. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

a) барлық қызметкерлердің аттарын, соның ішінде олардың басшысының (менеджерінің) атын қайтаратын SQL сұрауын жазыңыз.



Сурет - 1. HR – диаграммасы

SELECT e.employee\_id, e.first\_name AS employee\_first\_name, e.last\_name AS employee\_last\_name,

m.first\_name AS manager\_first\_name, m.last\_name AS manager\_last\_name

FROM employees e

LEFT JOIN employees m ON e.manager\_id = m.employee\_id

ORDER BY e.employee\_id;

 employees кестесінен e алиасын пайдаланып, барлық қызметкерлерді таңдайды.

 employees кестесінен m алиасын пайдаланып, қызметкерлердің менеджерлерін анықтайды (manager\_id мәні employee\_id мәніне сәйкес келеді).

 LEFT JOIN операторын пайдалану арқылы барлық қызметкерлерді қосады, тіпті олардың менеджері болмаса да (соңғысы негізгі менеджерлер үшін).

 Нәтиже ретінде қызметкерлердің идентификаторын, олардың аттарын (employee\_first\_name, employee\_last\_name), және олардың менеджерлерінің аттарын (manager\_first\_name, manager\_last\_name) қайтарады.

 ORDER BY клаузасында нәтижелерді қызметкердің идентификаторы бойынша реттейді.

28. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

a) бес қызметкерден астам комиссиялық пайызы (commission\_pct) бар бөлімдерді тізімдеу үшін SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT d.department\_id, d.department\_name, COUNT(e.employee\_id) AS num\_of\_employees\_with\_commission

FROM departments d

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

WHERE e.commission\_pct IS NOT NULL

GROUP BY d.department\_id, d.department\_name

HAVING COUNT(e.employee\_id) > 5;

 departments кестесінен d алиасын және employees кестесінен e алиасын пайдаланып, кестелерді department\_id бойынша біріктіреді.

 WHERE клаузасында комиссиялық пайызы (commission\_pct) бар қызметкерлерді ғана таңдайды (IS NOT NULL).

 GROUP BY клаузасында бөлімнің идентификаторы (department\_id) және бөлімнің атауы (department\_name) бойынша топтастырады.

 HAVING клаузасында комиссиялық пайызы бар қызметкерлердің саны бес немесе одан көп екенін тексереді.

 COUNT функциясын пайдаланып, комиссиялық пайызы бар қызметкерлердің санын есептейді (num\_of\_employees\_with\_commission).

29. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

a) компанияға 10-нан астам қызметкер қабылданған жылдарды қайтаратын SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT YEAR(hire\_date) AS hire\_year, COUNT(\*) AS num\_of\_employees

FROM employees

GROUP BY YEAR(hire\_date)

HAVING COUNT(\*) > 10;

Бұл сұрау атқарылған кезде, қызметкердің еңбекке орналасу күнінің жылын есептеу үшін hire\_date катарындан жылды алып, оны YEAR функциясы арқылы алып тастайды. Содан кейін, employees кестесінде кез келген жылға сәйкес келетін қызметкерлер санын есептеу үшін COUNT функциясы қолданылады. Ағымда, HAVING мәліметтер базасы кестесінде анықталған критерийлерді қолдана отырып, 10-нан астам қызметкердің кестеде қабылдану күндерін шолу үшін шартты тексереді.

30. Бірінші суреттегі диаграмманы пайдаланып келесі есепті шешіңіз.

a) басшысының (менеджердің) жалақысынан көп жалақысы бар қызметкерлердің тізімін шығару үшін SQL сұрауын жазыңыз.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

Сурет - 1. HR – диаграммасы

SELECT m.employee\_id, m.first\_name, m.last\_name, COUNT(e.employee\_id) AS num\_of\_subordinates

FROM employees m

JOIN employees e ON m.employee\_id = e.manager\_id

GROUP BY m.employee\_id, m.first\_name, m.last\_name

HAVING COUNT(e.employee\_id) > 1;

Бұл сұрау атқарылған кезде, басшысының (менеджердің) жалақысынан көп жалақысы бар қызметкерлерді табу үшін employees кестесінен m алиасын пайдаланамыз. Содан кейін, басшысының (менеджердің) employee\_id кескіні қолдана отырып, басшысының жалақысынан артық жалақысы бар қызметкерлерді табу үшін олардың жалақысынан (manager\_id) қолданамыз. GROUP BY клаузасы арқылы, басшысының (менеджердің) идентификаторы (employee\_id) және аты-жөні (first\_name, last\_name) бойынша топтастыруымыз керек. Содан кейін, HAVING мәліметтер базасы кестесінде анықталған критерийлерді қолдана отырып, басшысының (менеджердің) жалақысынан артық жалақысы бар қызметкерлердің санын тексереміз.