

Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Інститут прикладного системного аналізу  
Кафедра математичних методів системного аналізу

## **ЗВІТ**

про виконання комп'ютерного практикуму №3  
з дисципліни «Системи баз даних»

Виконав:

Студент III курсу

Групи КА-96

Вітковський Д.О.

Перевірила:

Афанасьєва І.В.

**Тема:** Мова маніпулювання даними (DML)

**Завдання:** Відповідно до обраної теми курсового проекту, створити представлення, перелічені у порядку виконання роботи.

### **Порядок виконання роботи**

#### **Робота з однією таблицею:**

1. Представлення, яке повертає усі дані таблиці.
2. Представлення, яке повертає рядки, що задовольняють наперед заданій умові.
3. Представлення, яке повертає рядки, що задовольняють наперед заданим умовам.
4. Представлення, яке повертає конкретні стовпці.
5. Представлення, яке повертає конкретні стовпці, які мають зрозумілі та зручні для читання імена (псевдоніми).
6. Представлення, яке використовує конкатенацію стовпців.
7. Представлення, яке використовує вираз CASE.
8. Представлення, яке повертає обмежену кількість рядків.
9. Представлення, яке повертає n випадкових рядків таблиці.
10. Представлення з пошуком значень NULL
11. Представлення з пошуком за шаблоном.
12. Представлення, в якому рядки відсортовано за деяким полем.
13. Представлення, в якому рядки відсортовано за більш ніж одним полем.
14. Представлення, в якому рядки відсортовано за під рядком (функція substring).
15. Представлення, в якому обробляються NULL значення при сортуванні.
16. Представлення, в якому рядки відсортовано за залежністю даних від ключа.

#### **Робота з кількома таблицями:**

17. Представлення, яке розміщує один набір рядків під іншим.
18. Представлення, яке поєднує пов'язані рядки

## JOIN ~ EQUI-JOIN ~ INNER JOIN

- 19. Представлення, яке знаходить однакові рядки в двох таблицях[1, стор.66].
- 20. Представлення, яке повертає записи із значеннями, яких немає в іншій таблиці

## ANTI-JOIN ~ LEFT (RIGHT) OUTER JOIN ~ LEFT (RIGHT) JOIN

- 21. Представлення, яке повертає записи із значеннями, для яких немає відповідності в іншій таблиці.
- 22. Представлення, в якому реалізовано незалежне додавання об'єднань у запит.
- 23. Представлення, в якому реалізовано об'єднання з використанням агрегованих функцій.
- 24. Представлення, в якому реалізовано зовнішнє об'єднання з використанням агрегованих функцій.
- 25. Представлення, в якому відсутні дані в кількох таблицях

## FULL OUTER JOIN

- 26. Представлення, в якому містяться NULL-значення в операціях та порівняннях.

## Робота з масивами символів – рядками:

- 27. Представлення з проходом рядка.
- 28. Представлення, в якому виводяться одиночні лапки.
- 29. Представлення, в якому видаляються всі непотрібні символи.
- 30. (\*) Представлення, в якому розділяються числові та символічні дані.
- 31. (\*) Представлення, в якому вибираються ініціали з імені.
- 32. (\*\*) Представлення, в якому перетворити рядок у список оператора IN із множиною значень.

## Робота з числами:

- 33. Представлення, в якому вивести мінімальне/максимальне значення у стовпці.
- 34. Представлення, яке повертає кількість записів у таблиці.

35. Представлення, яке повертає кількість визначених (не-NULL) значень.
36. Представлення, в якому обчислити поточну суму.

### Робота з датами:

37. Представлення, в якому обчислити кількість днів між двома датами.
38. (\*) Представлення, в якому обчислити кількість повторів днів тижня
39. протягом року.
40. (\*) Представлення, в якому визначити, чи рік високосний.
41. (\*) Представлення, в якому визначити перший та останній день місяця.
42. (\*\*) Представлення, в якому створити календар.
43. (\*\*) Представлення, в якому доповнити дати, що відсутні.
44. (\*\*) Представлення, в якому виявити накладення діапазонів дат.

Реалізувати скрипт, у якому передбачити: демонстрацію роботи кожного з заданих пунктів.

## Лістинг скрипту

### Робота з однією таблицею

```
-- 1
select * from plugins;
```

	id	title	description	category_id	icon	price
1	1	Test1	Test description	2	ojfpoqiwncau...	0.00
2	2	Waystones	Smoothly teleport you from one place to...	4	<null>	3.50
3	3	KatanaNiEr	Get yourself a katana that will follow ...	4	<null>	0.00
4	4	Structure plugin	Lets you to build multi-block structure...	5	<null>	1.50
5	5	Just guns	Uses Structure plugin API to add severa...	6	<null>	0.00

```
-- 2
select * from plugins where category_id = 2;
```

	id	title	description	category_id	icon	price
1	1	Test1	Test description	2	ojfpoqiwncaus...	0.00

```
-- 3
select * from plugins where category_id = 2 and icon IS NULL;
```

	id	title	description	category_id	icon	price
1	2	Waystones	Smoothly teleport you from on...	4	<null>	3.50
2	3	KatanaNiEr	Get yourself a katana that wi...	4	<null>	0.00

```
-- 4
select id, title, category_id from plugins;
```

	id	title	category_id
1	1	Test1	2
2	2	Waystones	4
3	3	KatanaNiEr	4
4	4	Structure plugin	5
5	5	Just guns	6

```
-- 5
select id plugin_id, title plugin_title, category_id plugin_category_id from
plugins;
```

	plugin_id	plugin_title	plugin_category_id
1	1	Test1	2
2	2	Waystones	4
3	3	KatanaNiEr	4
4	4	Structure plugin	5
5	5	Just guns	6

```
-- 6
select concat('(', id, ') ', title) as plugin from plugins;
```

	plugin
1	(1) Test1
2	(2) Waystones
3	(3) KatanaNiEr
4	(4) Structure plugin
5	(5) Just guns

```
-- 7
select id plugin_id,
       title plugin_title,
       (case
        when category_id = 1 then 'utils'
        when category_id = 2 then 'misc'
        when category_id = 3 then 'mini-games'
        when category_id = 4 then 'adventure'
        when category_id = 5 then 'tools'
        when category_id = 6 then 'mechanisms'
       end) as plugin_category
from plugins;
```

	plugin_id	plugin_title	plugin_category
1	1	Test1	misc
2	2	Waystones	adventure
3	3	KatanaNiEr	adventure
4	4	Structure plugin	tools
5	5	Just guns	mechanisms

```
-- 8
select * from plugins limit 2;
```

	id	title	description	category_id	icon	price
1	1	Test1	Test description	2	ojfpoqiwncaus...	0.00
2	2	Waystones	Smoothly teleport you from o...	4	<null>	3.50

```
-- 9
select * from plugins
order by rand()
limit 5;
```

	id	title	description	category_id	icon	price
1	4	Structure plugin	Lets you to build...	5	<null>	1.50
2	5	Just guns	Uses Structure pl...	6	<null>	0.00
3	3	KatanaNiEr	Get yourself a ka...	4	<null>	0.00
4	1	Test1	Test description	2	ojfpoqiwncaush...	0.00
5	2	Waystones	Smoothly teleport...	4	<null>	3.50

	id	title	description	category_id	icon	price
1	5	Just guns	Uses Structure plugin ...	6	<null>	0.00
2	2	Waystones	Smoothly teleport you ...	4	<null>	3.50
3	1	Test1	Test description	2	ojfpoqiwncaus...	0.00
4	3	KatanaNiEr	Get yourself a katana ...	4	<null>	0.00
5	4	Structure plugin	Lets you to build mult...	5	<null>	1.50

```
-- 10
select * from plugins where icon is null;
```

	id	title	description	category_id	icon	price
1	2	Waystones	Smoothly teleport you from ...	4	<null>	3.50
2	3	KatanaNiEr	Get yourself a katana that ...	4	<null>	0.00
3	4	Structure plugin	Lets you to build multi-blo...	5	<null>	1.50
4	5	Just guns	Uses Structure plugin API t...	6	<null>	0.00

```
-- 11
select * from plugins where title like '%n%';
```

	id	title	description	category_id	icon	price
1	2	Waystones	Smoothly teleport ...	4	<null>	3.50
2	3	KatanaNiEr	Get yourself a kat...	4	<null>	0.00
3	4	Structure plugin	Lets you to build ...	5	<null>	1.50
4	5	Just guns	Uses Structure plu...	6	<null>	0.00

```
-- 12
select * from plugins
order by title;
```

	id	title	description	category_id	icon	price
1	5	Just guns	Uses Structure plugin API...	6	<null>	0.00
2	3	KatanaNiEr	Get yourself a katana tha...	4	<null>	0.00
3	4	Structure plugin	Lets you to build multi-b...	5	<null>	1.50
4	1	Test1	Test description	2	ojfpoqiwnca...	0.00
5	2	Waystones	Smoothly teleport you fro...	4	<null>	3.50

```
-- 13
select * from plugins
order by category_id, title;
```

	id	title	description	category_id	icon	price
1	1	Test1	Test description	2	ojfpoqiwncaus...	0.00
2	3	KatanaNiEr	Get yourself a katana tha...	4	<null>	0.00
3	2	Waystones	Smoothly teleport you fro...	4	<null>	3.50
4	4	Structure plugin	Lets you to build multi-b...	5	<null>	1.50
5	5	Just guns	Uses Structure plugin API...	6	<null>	0.00

```
-- 14
select * from plugins
order by substr(title, 2);
```

	id	title	description	category_id	icon	price
1	3	KatanaNiEr	Get yourself a katana ...	4	<null>	0.00
2	2	Waystones	Smoothly teleport you ...	4	<null>	3.50
3	1	Test1	Test description	2	ojfpoqiwnca...	0.00
4	4	Structure plugin	Lets you to build mult...	5	<null>	1.50
5	5	Just guns	Uses Structure plugin ...	6	<null>	0.00

```
-- 15
select * from plugins
order by if(icon is null, 0, 1);
```

	id	title	description	category_id	icon	price
1	2	Waystones	Smoothly teleport you fro...	4	<null>	3.50
2	3	KatanaNiEr	Get yourself a katana tha...	4	<null>	0.00
3	4	Structure plugin	Lets you to build multi-b...	5	<null>	1.50
4	5	Just guns	Uses Structure plugin API...	6	<null>	0.00
5	1	Test1	Test description	2	ojfpoqiwncau...	0.00

```
-- 16
select * from plugins
order by if(category_id = id * 2, 1, 0);
```

	id	title	description	category_id	icon	price
1	3	KatanaNiEr	Get yourself a katana ...	4	<null>	0.00
2	4	Structure plugin	Lets you to build mult...	5	<null>	1.50
3	5	Just guns	Uses Structure plugin ...	6	<null>	0.00
4	1	Test1	Test description	2	ojfpoqiwncau...	0.00
5	2	Waystones	Smoothly teleport you ...	4	<null>	3.50

## Робота з кількома таблицями

```
-- 17
select nickname user_nickname, id
  from users
  where id = 1
union all
select '-----', null
union all
select distinct title plugin_title, plugins.id
  from plugins
  join plugin_authors pa on plugins.id = pa.plugin_id and pa.user_id = 1;
```

	user_nickname	id
1	Danik_Vitek	1
2	-----	<null>
3	Test1	1
4	Waystones	2
5	Structure plugin	4

```
-- 18
select u.nickname `Author Nickname`, p.title `Plugin Title`
from users u join plugin_authors pa on u.id = pa.user_id join plugins p on
pa.plugin_id = p.id
order by p.title desc, u.nickname;
```

	`Author Nickname`	`Plugin Title`
1	Danik_Vitek	Waystones
2	Danik_Vitek	Test1
3	SapeNeCo	Test1
4	Danik_Vitek	Structure plugin
5	SapeNeCo	Structure plugin
6	Ventor_	KatanaNiEr
7	SapeNeCo	Just guns

```
-- 19.1
create view if not exists V as
  select p.title, p.description, p.category_id from plugins p
  where p.category_id = 4;
```

```
-- 19.2
select p.title, p.description, p.category_id
from plugins p, v
where p.title = v.title
  and p.description = v.description
  and p.category_id = v.category_id;
```

	title	description	category_id
1	Waystones	Smoothly teleport you from one place to another; Yo...	4
2	KatanaNiEr	Get yourself a katana that will follow you, floatin...	4

```
-- 20
select u.nickname, u.first_name, u.last_name
from users u
where u.id not in (select pa.user_id from plugin_authors pa);
```

	nickname	first_name	last_name
1	Nobody_butAnyone	Nobody	Knows
2	NitkaN	Nikita	Nitka



```
-- 21
select u.nickname, u.first_name, u.last_name
from users u
left join plugin_authors pa on u.id = pa.user_id
where pa.plugin_id is null;
```

	🔍 nickname	🔍 first_name	🔍 last_name
1	Nobody_but_Anyone	Nobody	Knows
2	NitkaN	Nikita	Nitka

```
-- 22
select u.nickname, p.title
from users u
left join plugin_authors pa on u.id = pa.user_id
left join plugins p on pa.plugin_id = p.id;
```

	🔍 nickname	🔍 title
1	Danik_Vitek	Test1
2	Danik_Vitek	Waystones
3	Danik_Vitek	Structure plugin
4	NitkaN	<null>
5	Nobody_but_Anyone	<null>
6	SapeNeCo	Test1
7	SapeNeCo	Structure plugin
8	SapeNeCo	Just guns
9	Ventor_	KatanaNiEr

```
-- 23
with svg_by_categiry as
  (select c.id id,
         avg(p.price) avg_price
   from categories c
   join plugins p on p.category_id = c.id
   group by c.id)
select p.title,
       p.price,
       c.title category,
       sbc.avg_price `average price`
from plugins p
join categories c on p.category_id = c.id
join svg_by_categiry sbc on c.id = sbc.id;
```

	🔍 title	🔍 price	🔍 category	🔍 `average price`
1	Test1	0.00	misc	0.000000
2	Waystones	3.50	adventure	1.750000
3	KatanaNiEr	0.00	adventure	1.750000
4	Structure plugin	1.50	tools	1.500000
5	Just guns	0.00	mechanisms	0.000000

```
-- 24
with count_purch_by_user as
(select pp.user_id,
       count(pp.plugin_id) `count`
  from purchased_plugins pp
 group by pp.user_id)
select u.nickname `user nickname`,
       coalesce(cpbu.`count`, 0) `number of purchased plugins`
  from users u
 left join count_purch_by_user cpbu on cpbu.user_id = u.id
 order by `number of purchased plugins` desc;
```

	🔍 `user nickname`	🔍 `number of purchased plugins`
1	Ventor_	4
2	Nobody_but_Anyone	2
3	NitkaN	0
4	Danik_Vitek	0
5	SapeNeCo	0

```
-- 25
select u.nickname, p.title
  from users u
 left join purchased_plugins pa on u.id = pa.user_id
 left join plugins p on pa.plugin_id = p.id
 union
select u.nickname, p.title
  from users u
 right join purchased_plugins pa on u.id = pa.user_id
 right join plugins p on pa.plugin_id = p.id;
```

	🔍 nickname	🔍 title
1	Danik_Vitek	<null>
2	NitkaN	<null>
3	Nobody_but_Anyone	Waystones
4	Nobody_but_Anyone	Structure plugin
5	SapeNeCo	<null>
6	Ventor_	Waystones
7	Ventor_	KatanaNiEr
8	Ventor_	Structure plugin
9	Ventor_	Just guns
10	<null>	Test1

```
-- 26
with count_made_by_user as
(select pa.user_id,
       count(pa.plugin_id) `count`
  from plugin_authors pa
 group by pa.user_id)
select u.nickname `user nickname`,
       coalesce(cmbu.`count`, 0) `number of created plugins`
  from users u
   left join count_made_by_user cmbu on cmbu.user_id = u.id
 having `number of created plugins` < 2
 order by `number of created plugins` desc;
```

	🔍 `user nickname`	🔍 `number of created plugins`
1	Ventor_	1
2	NitkaN	0
3	Nobody_but_Anyone	0

## Робота з масивами символів – рядками

```
-- 27
select substr(p.title, iter.pos, 1) as C
from (select title from plugins where id = 2) p,
     (select id as pos from t10) iter
where iter.pos <= length(p.title);
```

	C
1	W
2	a
3	y
4	s
5	t
6	o
7	n
8	e
9	s

```
-- 28
select 'you can write quotas using \\' or \''';
```

	you can write quotas using \' or \''
1	you can write quotas using \' or \''

```
-- 29
select u.nickname `user nickname`,
       regexp_replace(u.nickname, '[eyuioa]', '') `stripped nickname`,
       p.title `plugin title`,
       regexp_replace(p.title, '[eyuioa]', '') `stripped title`
from users u
join plugin_authors pa on u.id = pa.user_id
join plugins p on pa.plugin_id = p.id;
```

	`user nickname`	`stripped nickname`	`plugin title`	`stripped title`
1	Danik_Vitek	Dnk_Vtk	Test1	Tst1
2	Danik_Vitek	Dnk_Vtk	Waystones	Wstns
3	Danik_Vitek	Dnk_Vtk	Structure plugin	Strctr plgn
4	SapeNeCo	SpNC	Test1	Tst1
5	SapeNeCo	SpNC	Structure plugin	Strctr plgn
6	SapeNeCo	SpNC	Just guns	Jst gns
7	Ventor_	Vntr_	KatanaNiEr	KtnNr

```
-- 30
with artificial_table as
  (select concat(p.title, p.price) str from plugins p)
select regexp_replace(`at`.str, '[0-9.]', '') `text`,
       regexp_replace(`at`.str, '[a-zA-Z _]', '') `nums`
from artificial_table as `at`;
```

	text	nums
1	Test	10.00
2	Waystones	3.50
3	KatanaNiEr	0.00
4	Structure plugin	1.50
5	Just guns	0.00

```
-- 31
with full_names as
  (select concat(u.first_name, ' ', u.last_name) full_name from users u)
select concat(case when cnt = 2 then
  concat_ws(
    '.',
    substr(substring_index(full_name, ' ', 1), 1, 1),
    substr(full_name,
      length(substring_index(full_name, ' ', 1))+ 2, 1),
    substr(substring_index(full_name, ' ', -1), 1, 1))
  else
    concat_ws(
      '.',
      substr(substring_index(full_name, ' ', 1), 1, 1),
      substr(substring_index(full_name, ' ', -1), 1, 1))
  end, '.') as initials
from (select full_name,
  length(full_name)-length(replace(full_name, ' ', '')) as cnt
  from (select replace(fns.full_name, '.', '') as full_name
    from full_names fns) y
) x;
```

	initials
1	D.V.
2	V.S.
3	S.N.
4	N.K.
5	N.N.

```
-- 32
with str_list as (select '3,4,5' vals)
select u.nickname
from users u
where u.id in (
  select substring_index(substring_index(str_list.vals, ',', iter.pos), ',', -
1) u_id
  from (select id pos from t10) iter,
  str_list
  where iter.pos <= length(str_list.vals) - length(replace(str_list.vals, ',',
')) + 1
);
```

	nickname
1	SapeNeCo
2	Nobody_but_Anyone
3	NitkaN

## Робота з числами

```
-- 33
select min(p.price) min_price from plugins p;
```

	min_price
1	0.00

```
-- 34
select count(*) total_count from plugins;
```

	total_count
1	5

```
-- 35
select count(p.icon) present_icons
from plugins p
where p.icon is not null;
```

	present_icons
1	1

```
-- 36
select p.title,
       p.price,
       (select sum(_p.price) from plugins _p
        where _p.id <= p.id) total_price
from plugins p
order by total_price;
```

	title	price	total_price
1	Test1	0.00	0.00
2	Waystones	3.50	3.50
3	KatanaNiEr	0.00	3.50
4	Structure plugin	1.50	5.00
5	Just guns	0.00	5.00

## Робота з датами

```
-- 37
select datediff(
    max(u.registration_time),
    min(u.registration_time)
) max_registration_differecne
from users u;
```

	max_registration_differecne
1	42

```
-- 38
select date_format(
    date_add(
        cast(concat(year(current_date), '-01-01') as date),
        interval t500.id-1 day),
    '%W') day,
count(*) `count`
from t500
where t500.id <= datediff(
    cast(concat(year(current_date)+1, '-01-01') as date),
    cast(concat(year(current_date), '-01-01') as date))
group by date_format(
    date_add(cast(concat(year(current_date), '-01-01') as date),
    interval t500.id-1 day),
    '%W');
```

	day	count
1	Friday	53
2	Monday	52
3	Saturday	52
4	Sunday	52
5	Thursday	52
6	Tuesday	52
7	Wednesday	52

```
-- 39
select if(day(
    last_day(
        date_add(
            date_add(
                date_add(
                    current_date,
                    interval -dayofyear(current_date) day
                ),
                interval 1 day
            ),
            interval 1 month
        )
    )
) = 28, 'Не високосний', 'Високосний') leap_year;
```

	leap_year
1	Не високосний

```
-- 40
select date_add(
    current_date,
    interval -day(current_date)+1 day) first_day,
last_day(current_date) last_day;
```

	first_day	last_day
1	2021-11-01	2021-11-30

```
-- 41
select max(case dw when 2 then dm end) as Mo,
       max(case dw when 3 then dm end) as Tu,
       max(case dw when 4 then dm end) as We,
       max(case dw when 5 then dm end) as Th,
       max(case dw when 6 then dm end) as Fr,
       max(case dw when 7 then dm end) as Sa,
       max(case dw when 1 then dm end) as Su
from (select date_format(dy,'%u') wk,
           date_format(dy,'%d') dm,
           date_format(dy,'%w')+1 dw
      from (select adddate(x.dy, t500.id-1) dy
            from (select adddate(current_date,-dayofmonth(current_date)+1) dy,
                  date_format(
                     adddate(current_date,
                              -dayofmonth(current_date)+1),
                              '%m') mth
                 ) x,
            t500
      where t500.id <= 31
            and date_format(adddate(x.dy,t500.id-1),'%m') = x.mth
      ) y
) z
group by wk
order by wk;
```

	Mo	Tu	We	Th	Fr	Sa	Su
1	01	02	03	04	05	06	07
2	08	09	10	11	12	13	14
3	15	16	17	18	19	20	21
4	22	23	24	25	26	27	28
5	29	30	<null>	<null>	<null>	<null>	<null>

```

-- 42
with users_months as (
    with reg_time_to_month as (
        select adddate(date(u.registration_time), -
dayofmonth(u.registration_time)+1) month_start
        from users u)
    select rttm.month_start month_start,
        count(rttm.month_start) num_reg
    from reg_time_to_month rttm
    group by rttm.month_start)
select um.month_start, um.num_reg
from users_months um
union
select w.month_start, w.num_reg
from (
    select z.mnth month_start, 0 num_reg
    from (
        select date(date_add(min_rd, interval t500.id - 1 month)) mnth
        from (
            select min_rd,
                date_add(max_rd, interval 12 month) max_rd
            from (
                select adddate(min(u.registration_time), -
dayofyear(min(u.registration_time)) + 1) min_rd,
                    adddate(max(u.registration_time), -
dayofyear(max(u.registration_time)) + 1) max_rd
                from users u
            ) x
        ) y,
        t500
        where date_add(min_rd, interval t500.id - 1 month) <= max_rd
    ) z
) w
order by 1;

```

	month_start	num_reg
1	2021-01-01	0
2	2021-02-01	0
3	2021-03-01	0
4	2021-04-01	0
5	2021-05-01	0
6	2021-06-01	0
7	2021-07-01	0
8	2021-08-01	0
9	2021-09-01	0
10	2021-10-01	0
11	2021-11-01	4
12	2021-11-01	0
13	2021-12-01	0
14	2022-01-01	0
15	2022-01-01	1
16	2022-02-01	0
17	2022-03-01	0
18	2022-04-01	0
19	2022-05-01	0
20	2022-06-01	0
21	2022-07-01	0
22	2022-08-01	0
23	2022-09-01	0
24	2022-10-01	0
25	2022-11-01	0
26	2022-12-01	0



```
-- 43
create table if not exists overlapping_time_periods(
    emp_mane varchar(50) not null,
    proj_id int not null,
    start_date date not null,
    end_date date not null
);
insert into overlapping_time_periods
values ('employee1', 1, '2021-01-01', '2021-02-03'),
       ('employee3', 1, '2021-01-01', '2021-02-03'),
       ('employee3', 2, '2021-01-01', '2021-01-31'),
       ('employee4', 2, '2021-01-01', '2021-02-03'),
       ('employee3', 3, '2021-01-29', '2021-02-10'),
       ('employee5', 3, '2021-01-29', '2021-02-10'),
       ('employee1', 4, '2021-01-31', '2021-02-28'),
       ('employee2', 4, '2021-01-31', '2021-02-28'),
       ('employee5', 5, '2021-02-08', '2021-03-03'),
       ('employee6', 5, '2021-02-08', '2021-03-03'),
       ('employee5', 6, '2021-03-01', '2021-03-14');
select a.emp_mane,
concat('project ',b.proj_id,
' overlaps project ',a.proj_id) as msg
from overlapping_time_periods a,
     overlapping_time_periods b
where a.emp_mane = b.emp_mane
and b.start_date >= a.start_date
and b.start_date <= a.end_date
and a.proj_id != b.proj_id;
```

	emp_mane	msg
1	employee3	project 1 overlaps project 2
2	employee3	project 2 overlaps project 1
3	employee3	project 3 overlaps project 1
4	employee3	project 3 overlaps project 2
5	employee1	project 4 overlaps project 1
6	employee5	project 5 overlaps project 3
7	employee5	project 6 overlaps project 5

## Висновки

Під час виконання комп'ютерного практикуму було створено та виконано 43 запити. Протягом цього було вивчено оператор with та отримано навички роботи зі строками, списками та датами.