

Бази даних та інформаційні системи

ЛАБОРАТОРНА РОБОТА №5

Нормалізація відношень бази даних

Виконав:

Студент Процьків Назарій

Група ПМі-21

Оцінка - _____

Перевірила:

доц. Малець Р.Б.

Тема: Нормалізація відношень бази даних.

Мета роботи: Ознайомлення з поняттям нормалізації відношень бази даних та самим процесом нормалізації.

Завдання (Варіант 17):

Розробити базу даних для сайту соціальної мережі. Соціальна мережа підтримує реєстрацію користувачів зі збереженням усіх їхніх деталей (ім'я, прізвище, дата народження, місце проживання, телефони, сайти/е-mail/skype іт.д.), місця і періоди перебування, місця і періоди навчання, роботи, служби, приєднані файли (зображення, фільми, аудіо), які можна пов'язувати із місцями з деталей. Крім того, кожен користувач має можливість розміщувати свої повідомлення на власній сторінці, отримувати на повідомлення «лайки» та коментарі, а також додавати інших користувачів в друзі, або в «чорний список». Додатково користувач повинен мати змогу шукати нових друзів за довільними критеріями.

Хід роботи

1. Опрацював теоретичний матеріал.
2. Проаналізував створені відношення (таблиці) бази даних на відповідність нормальним формам:

Таблиця Users:

Data Output Messages Notifications												
	id [PK] bigint	firstname text	lastname text	birthdate date	phonenum number text	password text	address text	workPlace text	studyPlace text	servicePlace text	beingPlace text	
1	1	Nazarii	Protskiv	2003-12-29	+380963566929	qwerty123456	Lviv, Chervonoi Kalyny 58B	work1	study1	service1	being1	
2	2	Oleksandr	Zhenchenko	2003-12-28	+380683838806	sanya123456	Lviv, Hnata Hotkevycha 1	work2	study2	service2	being2	
3	3	Olena	Hatala	2004-07-23	+380993740427	olena123456	Velykyi Lubin 1	work3	study3	service3	being3	
4	4	Sofia	Hoshko	2005-12-03	+380685435516	sofia123456	Dubliany 1	work4	study4	service4	being4	
5	5	Yarema	Tymchyshyn	2004-08-20	+380959388072	yarema123456	Luts 1	work5	study5	service5	being5	
6	6	bad	user	2002-10-21	+380954358972	baduser123456	Lviv, Shevchenka 1	work6	study6	service6	being6	
7	7	7	7	2001-08-01	+380421505772	password7	address7	work7	study7	service7	being7	
8	8	8	8	2007-01-14	+380625648721	password8	address8	work8	study8	service8	being8	
9	9	9	9	2001-08-10	+380761049381	password9	address9	work9	study9	service9	being9	
10	10	10	10	2001-12-16	+380716615141	password10	address10	work10	study10	service10	being10	
11	11	11	11	2007-09-17	+380651103558	password11	address11	work11	study11	service11	being11	
12	12	12	12	2008-02-26	+380196316776	password12	address12	work12	study12	service12	being12	
13	13	13	13	2008-01-03	+380179661808	password13	address13	work13	study13	service13	being13	
14	14	14	14	2007-10-06	+380938340923	password14	address14	work14	study14	service14	being14	
15	15	15	15	2001-05-06	+380955056237	password15	address15	work15	study15	service15	being15	
16	16	16	16	2002-05-13	+380177610829	password16	address16	work16	study16	service16	being16	
17	17	17	17	2004-01-25	+380351489499	password17	address17	work17	study17	service17	being17	

Таблиця має унікальний ключовий атрибут – id, у таблиці немає рядків, які повторюються, а також всі значення атрибутів є елементарними, тому ця таблиця є у 1NF.

В таблиці первинний ключ є простим (id), тому всі інші значення функціонально повно залежать від нього. З цього випливає, що таблиця users є у 2NF.

В таблиці всі неключові атрибути (firstname, lastname, birthdate, phonenum,

password, address, workPlace, studyPlace, servicePlace, beingPlace) взаємно незалежні між собою і повністю залежать лише від первинного ключа id, тобто кожний неключовий атрибут не транзитивно залежить від ключа, тому дана таблиця є у 3NF.

В цій таблиці функціональні залежності можна описати як
id -> firstname, lastname, birthdate, phonenumber, password, address, workPlace, studyPlace, servicePlace, beingPlace. Цей ланцюг залежностей можна розбити на 10.

1. id -> firstname
2. id -> lastname
3. id -> birthdate
4. id -> phonenumber
5. id -> password
6. id -> address
7. id -> workPlace
8. id -> studyPlace
9. id -> servicePlace
10. id -> beingPlace

Оскільки детермінант кожної з цих залежностей є ключем, то дана таблиця є у BCNF.

У таблиці немає багатозначних залежностей, які не є функціональними залежностями, тому ця таблиця є у 4NF.

Таблиця не має залежних сполучень, що не визначаються потенційними ключами, отже ця таблиця є у 5NF.

З цих пояснень випливає, що дану таблицю не потрібно оптимізовувати.

Таблиця page:

Query

Query History

1

select * from page

Data Output

Messages

Notifications

pageid

bigint

message_id

[PK] text

1

2

#23

2

3

#33

3

4

#43

4

5

#53

5

6

#63

Таблиця має унікальний ключовий атрибут – pageid, у таблиці немає рядків, які повторюються, а також всі значення атрибутів є елементарними, тому ця таблиця є у 1NF.

В таблиці первинний ключ є простим (pageid), тому всі інші значення функціонально повно залежать від нього. З цього випливає, що таблиця page є у 2NF.

В таблиці всі неключові атрибути (message_id) взаємно незалежні між собою і залежать лише від первинного ключа pageid, тобто кожний неключовий атрибут не транзитивно залежить від ключа, тому дана таблиця є у 3NF.

В цій таблиці функціональні залежності можна описати як id -> message_id. Це одинарний ланцюг, тому його не потрібно розбивати на менші. Оскільки детермінант кожної з цих залежностей є ключем, то дана таблиця є у BCNF.

У таблиці немає багатозначних залежностей, які не є функціональними залежностями, тому ця таблиця є у 4NF.

Таблиця не має залежних сполучень, що не визначаються потенційними ключами, отже ця таблиця є у 5NF.

З цих пояснень випливає, що дану таблицю не потрібно оптимізувати.

Таблиця message:

Query		Query History	
1		select * from message	
Data Output		Messages	
message_id [PK] text		message_text text	
1	#13	I am Nazarii3	
2	#14	I am Nazarii4	
3	#15	I am Nazarii5	
4	#16	I am Nazarii6	
5	#17	I am Nazarii7	
6	#18	I am Nazarii8	
7	#19	I am Nazarii9	
8	#110	I am Nazarii10	
9	#111	I am Nazarii11	
10	#112	I am Nazarii12	
12	#114	I am Nazarii14	
13	#115	I am Nazarii15	
14	#116	I am Nazarii16	
15	#117	I am Nazarii17	
16	#118	I am Nazarii18	
17	#119	I am Nazarii19	
18	#120	I am Nazarii20	
19	#121	I am Nazarii21	
20	#122	I am Nazarii22	
21	#123	I am Nazarii23	
22	#124	I am Nazarii24	

Таблиця має унікальний ключовий атрибут – message_id, у таблиці немає рядків, які повторюються, а також всі значення атрибутів є елементарними, тому ця таблиця є у 1NF.

В таблиці первинний ключ є простим (message_id), тому всі інші значення функціонально повно залежать від нього. З цього випливає, що таблиця page є у 2NF.

В таблиці всі неключові атрибути (message_text) взаємно незалежні між собою і залежать лише від первинного ключа message_id, тобто кожний неключовий атрибут не транзитивно залежить від ключа, тому дана таблиця є у 3NF.

В цій таблиці функціональні залежності можна описати як message_id -> message_text. Це одинарний ланцюг, тому його не потрібно розбивати на менші. Оскільки детермінант кожної з цих залежностей є ключем, то дана таблиця є у BCNF.

У таблиці немає багатозначних залежностей, які не є функціональними залежностями, тому ця таблиця є у 4NF.

Таблиця не має залежних сполучень, що не визначаються потенційними ключами, отже ця таблиця є у 5NF.

З цих пояснень випливає, що дану таблицю не потрібно оптимізувати.

Таблиця commentary:

Query

Query History


1


select * from commentary


Data Output


Messages


Notifications




















	commentary_id text	userid bigint	message_id text	commentary_text text
1	&21	2	#11	Wow! Cool! 21
2	&22	2	#11	Wow! Cool! 22
3	&23	2	#11	Wow! Cool! 23
4	&41	4	#31	Wow! Cool! 41
5	&42	4	#31	Wow! Cool! 42
6	&43	4	#51	Wow! Cool! 43
7	&51	5	#11	Wow! Cool! 51
8	&52	5	#11	Wow! Cool! 52
9	&53	5	#11	Wow! Cool! 53

Таблиця має унікальний ключовий атрибут – commentary_id, у таблиці немає рядків, які повторюються, а також всі значення атрибутів є елементарними, тому ця таблиця є у 1NF.

В таблиці первинний ключ є простим (commentary_id), тому всі інші значення

функціонально повно залежать від нього. З цього випливає, що таблиця page є у 2NF.

В таблиці всі неключові атрибути (userid, message_text, commentary_text) взаємно незалежні між собою і залежать лише від первинного ключа commentary_id, тобто кожний неключовий атрибут не транзитивно залежить від ключа, тому дана таблиця є у 3NF.

В цій таблиці функціональні залежності можна описати як commentary_id -> userid, message_id, message_text.

Цей ланцюг залежностей можна розбити на 3.

1. commentary_id -> userid.
2. commentary_id -> message_id.
3. commentary_id -> message_text.

Оскільки детермінант кожної з цих залежностей є ключем, то дана таблиця є у BCNF.

У таблиці немає багатозначних залежностей, які не є функціональними залежностями, тому ця таблиця є у 4NF.

Таблиця не має залежних сполучень, що не визначаються потенційними ключами, отже ця таблиця є у 5NF.

З цих пояснень випливає, що дану таблицю не потрібно оптимізувати.

Таблиця friendslist:

Query

Query History


1


select * from friendslist


Data Output


Messages


Notifications






















	userid bigint		friendid bigint	
1	1		2	
2	1		3	
3	1		4	
4	1		5	
5	2		3	
6	2		4	
7	2		5	
8	2		1	
9	3		1	
10	3		5	
11	4		1	
12	4		2	
13	4		3	
14	5		1	
15	5		3	

Ця таблиця має лише два атрибути: ключ userid і залежний від нього friendid. Очевидно, що ця таблиця також є у 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.

Дану таблицю не потрібно оптимізовувати.

Таблиця blacklist:

Query

Query History

1

select * from blacklist

Data Output

Messages

Notifications

	userid bigint		enemyid bigint	
1		1		6
2		2		6
3		3		6
4		4		6
5		5		6
6		6		1
7		6		2
8		6		3
9		6		4
10		6		5
11		3		5
12		3		4

Ця таблиця за будовою є такою самою як і таблиця friendslist, тому вона також є у 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.

Таблиця likes:

Query

Query History

1

select * from Likes

Data Output

Messages

Notifications

<

Ця таблиця також має два атрибути: userid – ключ, message_id – залежний від ключа. Ця таблиця також є у 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.

Останніх 4 таблиці мають однакову будову, тому я їх всіх буду розглядати одночасно. Таблиці beingplaces, studyplaces, serviceplaces, workplaces:

Beingplaces:

Data Output					Messages					Notifications				
	beingid		city		startdate		enddate							
	text		text		date		date							
1	being1		Lisabon		2015-12-12		2015-12-27							
2	being1		Lviv		2015-12-28		2016-02-21							
3	being1		Ankara		2016-02-12		2016-06-17							
4	being1		Lviv		2016-06-18		2023-03-19							
5	being2		Lviv		2016-12-12		2016-12-27							
6	being2		Kyiv		2016-12-28		2017-02-21							
7	being2		Cherkasy		2017-02-12		2017-07-17							
8	being2		Lubin		2017-07-18		2023-03-19							
9	being3		Lviv		2016-12-12		2016-12-27							
10	being3		Kyiv		2016-12-28		2017-02-21							
11	being3		Cherkasy		2017-02-12		2017-07-17							
12	being3		Lubin		2017-07-18		2023-03-19							

Studyplaces:

Data Output					Messages					Notifications				
	studyid		study		startdate		enddate							
	text		text		date		date							
1	study1		LPML		2018-09-01		2021-05-31							
2	study1		LNU		2021-09-01		2024-05-31							
3	study2		ViytivkaSchool		2010-09-01		2021-05-31							
4	study2		LNU		2021-09-01		2024-05-31							
5	study3		LubinSchool		2010-09-01		2021-05-31							
6	study3		LNU		2021-09-01		2024-05-31							
7	study4		DublianySchool		2018-09-01		2021-05-31							
8	study4		LNU		2021-09-01		2024-05-31							
9	study5		YaremaSchool		2010-09-01		2021-05-31							
10	study5		LNU		2021-09-01		2024-05-31							

Serviceplaces:


Data Output					Messages					Notifications				
	serviceid		service		startdate		enddate							
	text		text		date		date							
1	service1		LisabonService		2015-12-12		2015-12-27							
2	service1		LvivService		2015-12-28		2016-02-21							
3	service1		AnkaraService		2016-02-12		2016-06-17							
4	service1		LvivService		2016-06-18		2023-03-19							
5	service2		LvivService		2016-12-12		2016-12-27							
6	service2		KyivService		2016-12-28		2017-02-21							
7	service2		CherkasyService		2017-02-12		2017-07-17							
8	service2		LubinService		2017-07-18		2023-03-19							
9	service3		LvivService		2016-12-12		2016-12-27							
10	service3		KyivService		2016-12-28		2017-02-21							
11	service3		CherkasyService		2017-02-12		2017-07-17							


Workplaces:


Data Output


Messages


Notifications




















	<div>workid</div> <div>text</div> <div></div>	<div>company</div> <div>text</div> <div></div>	<div>startdate</div> <div>date</div> <div></div>	<div>enddate</div> <div>date</div> <div></div>
1	work1	Google	2015-12-12	2015-12-27
2	work1	Amazon	2015-12-28	2016-02-21
3	work1	Starlink	2016-02-12	2016-06-17
4	work1	Google	2016-06-18	2023-03-19
5	work2	Amazon	2016-12-12	2016-12-27
6	work2	Facebook	2016-12-28	2017-02-21
7	work2	Starlink	2017-02-12	2017-07-17
8	work2	Glovo	2017-07-18	2023-03-19
9	work3	Bolt	2016-12-12	2016-12-27

Ці таблиці мають 4 атрибути з першою колонкою як ключ і іншими трьома як залежними, в цих таблицях немає повторень кортежів, а також всі значення атрибутів є елементарними, тому ці таблиці є у 1NF.

З попередніх пояснень очевидно, що ці таблиці також є у 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.

Висновок: моя база даних не потребувала нормалізації жодної з таблиць, тому у звіті до цієї лабораторної роботи я просто пояснив чому. Під час виконання лабораторної роботи я ознайомився з поняттям нормалізації відношень бази даних та власне самим процесом нормалізації.