

ПРОЕКТ ПО БАЗИ ДАННИ

на тема:

Електронен дневник

Съдържание

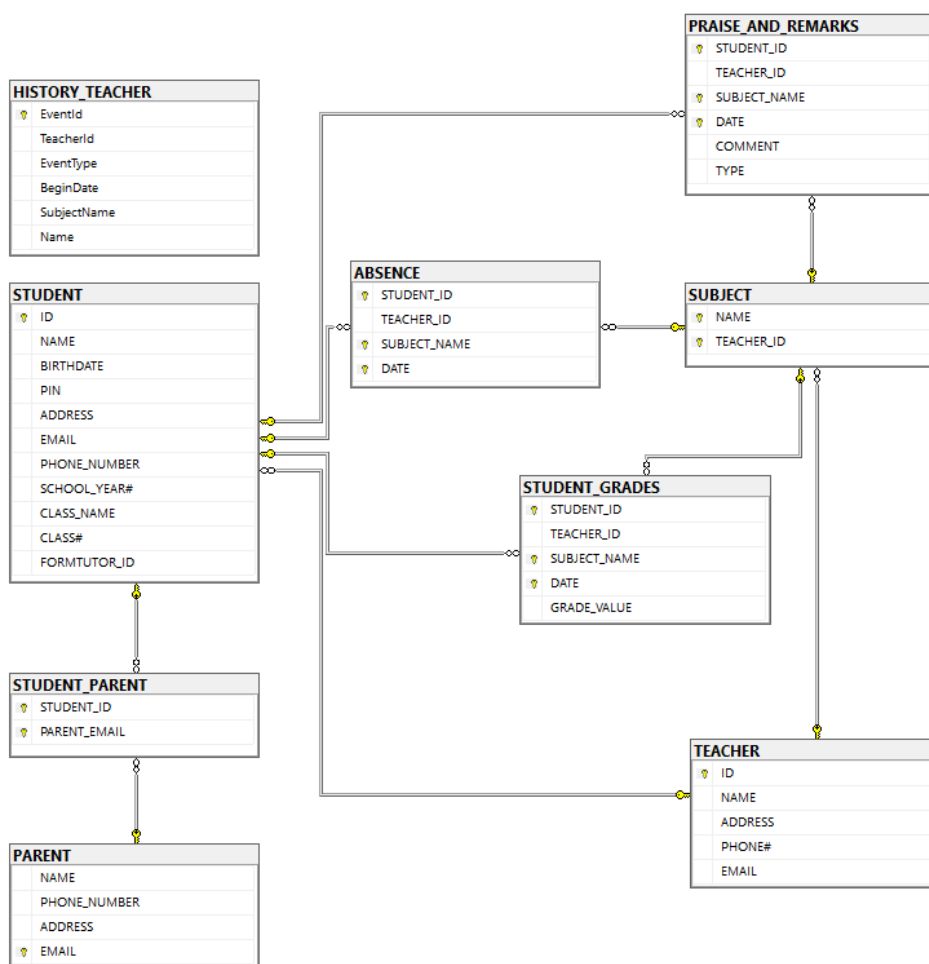
1. Описание на предметната област и на заданието
2. Схема на релациите
3. Примерно съдържание
4. Примерни заявки с резултата им разделени на:
 - 4.1. Прости заявки
 - 4.2. Заявки върху две и повече релации
 - 4.3. Подзаявки
 - 4.4. Съединения
 - 4.5. Групиране и агрегация
5. Заключение върху:
 - 5.1. Подобрения с добавяне на ограничения на ниво таблица
 - 5.2. Доразвиване на системата с добавяне на изгледи – индекси и тригери

1. Описание на предметната област и на заданието

Базата данни за електронен дневник е предназначена за използване от учебните заведения. Съдържа информация за резултатите от текущия контрол на учениците, получени по време на учебната година.

В дневника е поместена, както информация за оценките, заедно с личните данни на ученик, така и лични данни на преподавателите и родителите на учениците. Съдържа се информация за конкретен учебен предмет и за набора от оценки на съответен ученик. Поместени са, също така, данни за отсъствията по дата на даден ученик и такива за похвалите и забележките, които е получил.

2. Схема на релациите



Таблицата **STUDENT** съдържа информация за конкретен ученик. Първичният ключ е атрибутът **ID**.

- **ID** - пореден номер на ученика;

- NAME - две имена;
- BIRTHDATE - рождена дата;
- PIN - ЕГН;
- ADDRESS - адрес на местоживеене;
- EMAIL - личен e-mail на ученика;
- PHONE_NUMBER - телефонен номер на ученика;
- SCHOOL_YEAR# - номер на випуска, към който принадлежи ученикът;
- CLASS_NAME - име на паралелката;
- CLASS# - номер на ученика в класа;
- FORMTUTOR_ID - идентификационен номер на класния ръководител; външен ключ към TEACHER.ID;

Таблицата **PARENT** съдържа информация за съответен родител на ученик.

Първичният ключ е атрибутът *EMAIL*.

- NAME - две имена на родител;
- PHONE_NUMBER - телефонен номер на родител;
- ADDRESS - адрес на родител;
- EMAIL - e-mail на родител;

Таблицата **STUDENT_PARENT** е помощна таблица, обуславяща връзката между таблиците **STUDENT** и **PARENT**. Двата ѝ атрибута *STUDENT_ID* и *PARENT_EMAIL* заедно формират първичния ключ.

- STUDENT_ID - идентификационен номер на даден ученик; външен ключ към STUDENT.ID;
- PARENT_EMAIL - имейл на родител; външен ключ към PARENT.EMAIL;

Таблицата **TEACHER** съдържа информация за конкретен преподавател. Първичният ключ е атрибутът *ID*.

- ID - пореден номер на преподавателя;
- NAME - две имена;
- ADDRESS - адрес на местоживеене;
- PHONE# - личен телефонен номер;
- EMAIL - личен e-mail на преподавателя;

Таблицата **SUBJECT** съдържа информация за учебните предмети, преподавани в конкретното училище. Първичният ключ е формиран заедно от атрибутите *NAME* и *TEACHER_ID*.

- NAME - име на учебния предмет;
- TEACHER_ID - идентификационен номер на преподавателя, преподаващ съответния предмет; външен ключ към TEACHER.ID;

Таблицата **ABSENCE** съдържа информация за отсъствията на конкретен ученик. Първичният и ключ се формира заедно от атрибутите *STUDENT_ID*, *SUBJECT_NAME* и *DATE*.

- STUDENT_ID - идентификационен номер на даден ученик; външен ключ към STUDENT.ID;

- TEACHER_ID - идентификационен номер на даден преподавател;
- SUBJECT_NAME - наименование на учебен предмет;
- DATE - дата на отсъствие;

Таблицата **STUDENT_GRADES** съдържа информация за оценките на конкретен ученик. Първичният ѝ ключ се формира заедно от атрибутите *STUDENT_ID*, *SUBJECT_NAME* и *DATE*.

- STUDENT_ID - идентификационен номер на даден ученик; външен ключ към STUDENT.ID;
- TEACHER_ID - идентификационен номер на даден преподавател;
- SUBJECT_NAME - наименование на учебен предмет;
- DATE - дата на получаване на оценката;
- GRADE_VALUE - стойност на оценката;

Таблицата **PRAISE_AND_REMARKS** съдържа информация за забележките и похвалите на конкретен ученик. Първичният ѝ ключ се формира заедно от атрибутите *STUDENT_ID*, *SUBJECT_NAME* и *DATE*.

- STUDENT_ID - идентификационен номер на даден ученик; външен ключ към STUDENT.ID;
- TEACHER_ID - идентификационен номер на даден преподавател;
- SUBJECT_NAME - наименование на учебен предмет;
- DATE - дата на получаване на забележката или похвалата;
- COMMENT - текст с контекста на забележката или похвалата;
- TYPE - вид, 'p' за похвала, 'r' за забележка;

Таблицата **HISTORY_TEACHER** е помощна таблица, влизаща в употреба при добавянето на тригери. Първичният ѝ ключ е атрибутът *EventId*.

- EventId - пореден номер на събитието;
- TeacherId - идентификационен номер на учителят;
- EventType - вид на събитието, 'update', 'insert', 'delete' ;
- BeginDate - начална дата;
- SubjectName - име на учебен предмет;
- Name - име на преподавател;

3. Примерно съдържание

- Съдържание на таблицата STUDENT

ID	NAME	BIRTHDATE	PIN	ADDRESS	EMAIL	PHONE_NUMBER	SCHOOL_YEAR#	CLASS_NAME	CLASS#	FORMTUTOR_ID
1	Vivien Crane	2004-07-30 00:00:00.000	447302515	Ap #784-6500 Nulla. Rd.	amet.lorem@hotmail.org	895632569	11	b	7	9
2	Madonna Osborne	2006-01-13 00:00:00.000	641132617	Ap #896-2184 Nullam. Av.	in@aol.ca	898456235	9	c	8	3
3	Tatiana Gomez	2003-01-23 00:00:00.000	341234628	Ap #325-3327 Velit. Rd.	curabitur.vel@aol.net	988256356	12	a	6	12
4	Bruce James	2005-01-24 00:00:00.000	541248312	545-3900 Leo Avenue	ut.nulla@outlook.edu	878412563	10	d	24	14
5	Garth Payne	2003-01-17 00:00:00.000	341171625	Ap #913-9585 Magna Ave	mauris@google.com	888756532	12	a	9	12
6	Courtney Kline	2005-08-01 00:00:00.000	548013658	482-7750 Mauris Road	ultrices@aol.org	898456123	10	d	11	14
7	Kiona Manning	2007-08-16 00:00:00.000	748153214	758-2849 Bibendum St.	omare.sagittis@google.com	886259636	8	g	23	5
8	Caldwell Mclean	2005-06-11 00:00:00.000	546117465	P.O. Box 503, 2540 Porttitor Street	diam.sed.diam@icloud.org	887142536	10	d	28	14
9	Lacota Sweeney	2004-09-15 00:00:00.000	449159582	Ap #448-2296 Luctus Rd.	iaculis@outlook.net	888788288	11	b	23	9
10	Mona Hall	2008-01-23 00:00:00.000	841237214	195-917 Dui. St.	cras.convallis.convallis@hotmail.net	878388256	7	e	1	4
11	Libby Benson	2005-08-27 00:00:00.000	548270036	711-7319 Nascetur Av.	justo@hotmail.couk	882456936	10	d	4	14
12	Hedwig Mann	2004-11-18 00:00:00.000	411185695	Ap #555-2914 Magna. St	erat@icloud.couk	886262636	11	b	10	9
13	Anastasia Pearson	2005-01-06 00:00:00.000	541063695	Ap #763-1691 Tincidunt Rd.	lobortis.nisi.nibh@aol.edu	882626396	10	f	1	8
14	Germane Macdonald	2005-09-29 00:00:00.000	549298793	Ap #966-7316 Egestas Avenue	pellentesque@google.ca	885625126	10	d	22	14
15	Gay Herman	2004-12-13 00:00:00.000	412135263	Ap #217-2853 Eget Avenue	omare.lectus@aol.ca	883462634	11	b	11	9
16	Zephania Davidson	2005-11-21 00:00:00.000	511214785	Ap #727-9127 Odio Ave	cras.interdum.nunc@hotmail.ca	882707577	10	d	19	14
17	Hanase Maxwell	2004-09-23 00:00:00.000	449236598	P.O. Box 291, 9079 Eros. St.	mollis@aol.net	8854023156	11	b	15	9
18	Zelda Bolton	2004-09-09 00:00:00.000	449095214	P.O. Box 330, 6911 Volutpat. Rd.	risus.varius@icloud.org	8896220901	11	b	30	9
19	Gannon Collier	2004-12-05 00:00:00.000	412056589	4032 Malesuada Av.	donec.egestas.aliquam@yahoo.edu	8741256398	11	b	18	9
20	Cleo Merritt	2005-02-09 00:00:00.000	542095231	Ap #374-4028 Malesuada Av.	molestie@outlook.org	884290503	10	f	10	8
21	Steven Chambers	2010-05-08 00:00:00.000	1045087459	4032 Malesuada Av.	tempor.lorem@google.couk	88888858	5	a	9	1
22	Ivana Huffman	2011-02-01 00:00:00.000	1142015691	P.O. Box 330, 6911 Volutpat. Rd.	at@icloud.com	877777737	4	d	21	7
23	Tanya Fulton	2010-12-09 00:00:00.000	1012096548	P.O. Box 291, 9079 Eros. St.	ligula.aenean.gravida@yahoo.couk	881236598	5	a	24	1
24	Chiquita Ward	2011-04-17 00:00:00.000	1144172567	Ap #727-9127 Odio Ave	at.fringilla@outlook.org	988878828	4	d	24	7
25	Zoe Home	2009-11-26 00:00:00.000	911265976	Ap #966-7316 Egestas Avenue	felis@aol.net	889789286	6	b	12	6
26	Tamara Melendez	2012-01-11 00:00:00.000	1241115261	Ap #784-6500 Nulla. Rd.	nam.ac.nulla@google.couk	883607525	3	c	16	10
27	Naida Duffy	2012-05-06 00:00:00.000	1245063974	545-3900 Leo Avenue	nec.malesuada.ut@protonmail.com	882804384	3	c	4	10
28	Xander Guy	2012-01-06 00:00:00.000	1241062597	Ap #913-9585 Magna Ave	nunc.mauris.sapien@hotmail.org	885905265	3	c	12	10
29	Tatum Mills	2011-02-01 00:00:00.000	1142015324	Ap #325-3327 Velit. Rd.	ut.pharetra.sed@aol.org	789369258	4	d	6	7
30	Emerson Lyons	2010-05-10 00:00:00.000	1045106215	Ap #784-6500 Nulla. Rd.	cursus@icloud.net	885275325	5	a	11	1

- Съдържание на таблицата TEACHER

ID	NAME	ADDRESS	PHONE#	EMAIL
1	Griffith Summers	P.O. Box 552, 5089 Suspendisse Av.	881823283	nunc@aol.org
2	Aileen Good	P.O. Box 227, 8412 Iaculis Rd.	984884656	pharetra@protonmail.org
3	Hashim Hall	Ap #265-4559 Faucibus St.	873550573	egestas.lacinia@hotmail.ca
4	Hall Ortiz	Ap #475-4594 Arcu. Ave	783076118	eleifend@outlook.ca
5	Rana Hess	Ap #755-6463 Tincidunt Avenue	989484578	libero.morbi@protonmail.com
6	Igor Mayo	P.O. Box 413, 4801 Elit Avenue	881312315	neque.sed.sem@yahoo.couk
7	Aileen Lott	6500 Tempus Rd.	874261432	lobortis.risus.in@google.com
8	Aaron Knight	247-2062 Pede Rd.	881953728	interdum.sed@icloud.org
9	Daniel Harvey	1799 Quisque Rd.	890159493	consectetuer.adipiscing@outlook.org
10	Len Chan	872-9640 A Road	882975453	id.libero@icloud.ca
11	Kim Powers	Ap #104-3812 Aliquam Road	886537821	ridiculus.mus.proin@icloud.edu
12	Virginia Noble	P.O. Box 110, 310 Scelerisque Rd.	895168662	proin.ultrices@aol.edu
13	Ezra Whitley	Ap #313-9598 Pede. Road	982342347	amet.luctus@aol.edu
14	Veda Farrell	Ap #540-3702 Sapien. St.	886648305	lectus.a@yahoo.couk
15	Dana Moreno	Ap #172-1599 Donec Avenue	871100606	rutrum.lorem.ac@hotmail.org

- Съдържание на таблицата **PARENT**

NAME	PHONE_NUMBER	ADDRESS	EMAIL
Cameron Delacruz	896891151	758-2849 Bibendum St.	auctor.nunc@icloud.com
Athena Neal	8885446221	P.O. Box 291, 9079 Eros. St.	et.ultrices@protonmail.ca
Bert Mccall	887311667	Ap #966-7316 Egestas Avenue	eusmod.mauris@google.ca
Jordan Burnett	984742536	P.O. Box 330, 6911 Volutpat. Rd.	facilisis.eget@outlook.org
Brenna Wagner	886715777	4032 Malesuada Av.	feugiat.non.lobortis@protonmail.com
Althea Alexander	874730883	Ap #763-1691 Tincidunt Rd.	in.faucibus@icloud.net
Nicholas Skinner	871574569	Ap #555-2914 Magna. St.	lacus.quisque@google.org
Hyacinth Greene	988213427	Ap #448-2296 Luctus Rd.	lobortis.mauris@icloud.ca
Linus Everett	983311903	Ap #325-3327 Velit. Rd.	nam.ligula.elit@hotmail.org
Rebecca Newton	883567406	Ap #374-4028 Malesuada Av.	nibh@protonmail.edu
Mercedes Ratliff	981550726	Ap #727-9127 Odio Ave	nulla.tempor@protonmail.edu
Fulton Shannon	895582642	Ap #217-2853 Eget Avenue	nunc@outlook.couk
Scarlett Neal	888856236	Ap #784-6500 Nulla. Rd.	rhoncus@outlook.net
Kelly Mullen	896342212	545-3900 Leo Avenue	sed.leo@google.net
Elvis Kirkland	873125384	195-917 Dui. St.	tellus.aenean@google.net
Deirdre Buchanan	892431307	Ap #896-2184 Nullam Av.	tellus@icloud.org
Otto Rush	888537633	Ap #913-9585 Magna Ave	turpis.in@icloud.couk
Lawrence Hunt	879715123	482-7750 Mauris Road	turpis@outlook.org
Kibo Forbes	883412477	711-7319 Nascetur Av.	ultrices@icloud.org
Daphne Pittman	785724625	P.O. Box 503, 2540 Porttitor Street	ut@icloud.org

- Съдържание на таблицата **STUDENT_PARENT**

STUDENT_ID	PARENT_EMAIL
1	rhoncus@outlook.net
2	tellus@icloud.org
3	nam.ligula.elit@hotmail.org
4	sed.leo@google.net
5	turpis.in@icloud.couk
6	turpis@outlook.org
7	auctor.nunc@icloud.com
8	ut@icloud.org
9	lobortis.mauris@icloud.ca
10	tellus.aenean@google.net
11	ultrices@icloud.org
12	lacus.quisque@google.org
13	in.faucibus@icloud.net
14	euismod.mauris@google.ca
15	nunc@outlook.couk
16	nulla.tempor@protonmail.edu
17	et.ultrices@protonmail.ca
18	facilisis.eget@outlook.org
19	feugiat.non.lobortis@protonmail.com
20	nibh@protonmail.edu
21	feugiat.non.lobortis@protonmail.com
22	facilisis.eget@outlook.org
23	et.ultrices@protonmail.ca
24	nulla.tempor@protonmail.edu
25	euismod.mauris@google.ca
26	rhoncus@outlook.net
27	sed.leo@google.net
28	turpis.in@icloud.couk
29	nam.ligula.elit@hotmail.org
30	rhoncus@outlook.net

- Съдържание на таблицата **SUBJECT**

NAME	TEACHER_ID
Biology	7
Bulgarian language	3
Bulgarian language	13
Bulgarian language	15
Chemistry	9
English language	4
English language	5
English language	14
Geography	8
German language	5
History	6
History	7
IT	1
IT	12
Mathematics	1
Music	10
Physics	2
Physics	9
Sport	11

- Съдържание на таблицата **PRAISE_AND_REMARKS**

STUDENT_ID	TEACHER_ID	SUBJECT_NAME	DATE	COMMENT	TYPE
3	9	Chemistry	2022-03-02 00:00:00.000	The student assists experiments very well!	p
4	9	Chemistry	2021-12-18 00:00:00.000	The student have lack of knowledge in specific pa...	r
7	7	Biology	2021-10-22 00:00:00.000	The student sleeps in class!	r
13	13	Bulgarian language	2022-04-24 00:00:00.000	The student eats in class!	r
15	6	History	2022-02-05 00:00:00.000	The student have excelent skills in analysing time...	p
19	9	Chemistry	2022-06-12 00:00:00.000	The student behaves inappropriately!	r
22	13	Bulgarian language	2022-06-10 00:00:00.000	The student need to focus on improving its sema...	r
27	5	German language	2021-12-03 00:00:00.000	The student shows fluent skills in "listening" and "...	p
27	10	Music	2022-01-12 00:00:00.000	The student knows all of the learned songs perfe...	p
30	15	Bulgarian language	2022-01-04 00:00:00.000	The student have really good articulation!	p

- Съдържание на таблицата **STUDENT_GRADES**

STUDENT_ID	TEACHER_ID	SUBJECT_NAME	DATE	GRADE_VALUE
1	12	IT	2022-05-11 00:00:00.000	6.00
1	10	Music	2022-04-04 00:00:00.000	4.50
2	14	English language	2022-03-20 00:00:00.000	2.00
2	2	Physics	2021-12-15 00:00:00.000	5.00
3	4	English language	2021-11-10 00:00:00.000	6.00
3	12	IT	2021-11-04 00:00:00.000	4.00
4	11	Sport	2022-01-25 00:00:00.000	6.00
5	8	Geography	2022-06-12 00:00:00.000	3.00
6	8	Geography	2021-12-23 00:00:00.000	6.00
8	15	Bulgarian language	2021-10-20 00:00:00.000	6.00
8	2	Physics	2022-05-07 00:00:00.000	5.00
9	2	Physics	2021-12-28 00:00:00.000	6.00
10	7	Biology	2021-12-20 00:00:00.000	5.00
10	6	History	2022-05-10 00:00:00.000	3.00
10	12	IT	2022-05-16 00:00:00.000	6.00
10	9	Physics	2021-12-11 00:00:00.000	6.00
12	1	Mathematics	2022-06-18 00:00:00.000	5.00
12	9	Physics	2022-05-11 00:00:00.000	2.00
13	9	Chemistry	2022-04-14 00:00:00.000	3.00
13	7	History	2022-02-27 00:00:00.000	5.00
14	3	Bulgarian language	2021-10-14 00:00:00.000	5.00
14	11	Sport	2022-02-02 00:00:00.000	6.00
16	14	English language	2022-02-10 00:00:00.000	4.00
16	2	Physics	2022-03-01 00:00:00.000	5.00
17	4	English language	2021-12-12 00:00:00.000	4.00
18	9	Chemistry	2021-12-09 00:00:00.000	5.00
18	1	Mathematics	2022-06-28 00:00:00.000	6.00
20	7	Biology	2022-02-12 00:00:00.000	6.00

21	14	3	Bulgarian language	2021-10-14 00:00:00.000	5.00
22	14	11	Sport	2022-02-02 00:00:00.000	6.00
23	16	14	English language	2022-02-10 00:00:00.000	4.00
24	16	2	Physics	2022-03-01 00:00:00.000	5.00
25	17	4	English language	2021-12-12 00:00:00.000	4.00
26	18	9	Chemistry	2021-12-09 00:00:00.000	5.00
27	18	1	Mathematics	2022-06-28 00:00:00.000	6.00
28	20	7	Biology	2022-02-12 00:00:00.000	6.00
29	21	5	German language	2022-03-19 00:00:00.000	5.00
30	21	1	Mathematics	2021-10-22 00:00:00.000	5.00
31	22	14	English language	2022-03-25 00:00:00.000	5.00
32	23	4	English language	2021-10-12 00:00:00.000	6.00
33	23	5	German language	2021-12-18 00:00:00.000	6.00
34	23	6	History	2022-02-25 00:00:00.000	5.00
35	23	10	Music	2022-04-28 00:00:00.000	6.00
36	25	5	German language	2021-10-14 00:00:00.000	6.00
37	27	13	Bulgarian language	2022-04-17 00:00:00.000	6.00
38	28	4	English language	2021-10-05 00:00:00.000	5.00
39	29	4	English language	2021-12-26 00:00:00.000	6.00
40	29	6	History	2021-10-09 00:00:00.000	4.00

- Съдържание на таблицата **ABSENCE**

STUDENT_ID	TEACHER_ID	SUBJECT_NAME	DATE
2	2	Physics	2022-02-02 00:00:00.000
5	6	History	2021-11-28 00:00:00.000
6	3	Bulgarian language	2022-05-20 00:00:00.000
7	4	English language	2022-05-15 00:00:00.000
7	8	Geography	2022-05-25 00:00:00.000
8	8	Geography	2022-06-24 00:00:00.000
10	3	Bulgarian language	2021-12-18 00:00:00.000
10	10	Music	2022-03-23 00:00:00.000
11	2	Physics	2022-01-28 00:00:00.000
13	15	Bulgarian language	2022-01-25 00:00:00.000
16	5	English language	2021-12-12 00:00:00.000
16	14	English language	2022-02-17 00:00:00.000
18	2	Physics	2022-01-12 00:00:00.000
20	13	Bulgarian language	2022-03-21 00:00:00.000
23	9	Chemistry	2022-01-30 00:00:00.000
28	8	Geography	2022-03-17 00:00:00.000
28	9	Physics	2021-12-30 00:00:00.000
29	1	Mathematics	2022-02-25 00:00:00.000
29	5	German language	2022-04-03 00:00:00.000
30	5	German language	2022-02-01 00:00:00.000

4. Примерни заявки с резултата им

4.1. Прости заявки

-- 1. Заявка, която извежда имената на всички ученици, които са от ,b‘ клас и ги сортира наобратно.

```
SELECT s.NAME  
FROM STUDENT s  
WHERE CLASS_NAME = 'b'  
ORDER BY s.NAME DESC
```

	NAME
1	Zoe Horne
2	Zelda Bolton
3	Vivien Crane
4	Lacota Sweeney
5	Hedwig Mann
6	Hanae Maxwell
7	Gay Herman
8	Gannon Collier

-- 2. Заявка, която извежда името на предмета, датата и коментара на всички похвали и ги сортира по дата.

```
SELECT p.SUBJECT_NAME, p.DATE, p.COMMENT  
FROM PRAISE_AND_REMARKS p  
WHERE p.TYPE = 'p'  
ORDER BY p.DATE
```

	SUBJECT_NAME	DATE	COMMENT
1	German language	2021-12-03 00:00:00.000	The student shows fluent skills in "listening" and ...
2	Bulgarian language	2022-01-04 00:00:00.000	The student have really good articulation!
3	Music	2022-01-12 00:00:00.000	The student knows all of the learned songs perfe...
4	History	2022-02-05 00:00:00.000	The student have excelent skills in analysing tim...
5	Chemistry	2022-03-02 00:00:00.000	The student assists experiments very well!

-- 3. Заявка, която извежда данните на преподавателя с име Hall Ortiz.

```
SELECT *
FROM TEACHER t
WHERE t.NAME = 'Hall Ortiz'
```

	ID	NAME	ADDRESS	PHONE#	EMAIL
1	4	Hall Ortiz	Ap #475-4594 Arcu. Ave	783076118	eleifend@outlook.ca

-- 4. Заявка, която извежда номера и адреса на родител с име Daphne Pittman.

```
SELECT p.PHONE_NUMBER, p.ADDRESS
FROM PARENT p
WHERE p.NAME = 'Daphne Pittman'
```

	PHONE_NUMBER	ADDRESS
1	785724625	P.O. Box 503, 2540 Porttitor Street

-- 5. Заявка, която извежда данните на всички ученици от 10 клас и ги сортира по име.

```
SELECT *
FROM STUDENT s
WHERE s.SCHOOL_YEAR# = 10
ORDER BY s.NAME
```

	ID	NAME	BIRTHDATE	PIN	ADDRESS	EMAIL	PHONE_NUMBER	SCHOOL_YEAR#	CLASS_NAME	CLASS#	FORMTUTOR_ID
1	13	Anastasia Pearson	2005-01-06 00:00:00.000	541063695	Ap #763-1691 Tincidunt Rd.	lobortis.nisi.nibh@aol.edu	882626396	10	f	1	8
2	4	Bruce James	2005-01-24 00:00:00.000	541248312	545-3900 Leo Avenue	ut.nulla@outlook.edu	878412563	10	d	24	14
3	8	Caldwell Mclean	2005-06-11 00:00:00.000	546117465	P.O. Box 503, 2540 Porttitor Street	diam.sed.diam@icloud.org	887142536	10	d	28	14
4	20	Cleo Merritt	2005-02-09 00:00:00.000	542095231	Ap #374-4028 Malesuada Av.	molesie@outlook.org	884290503	10	f	10	8
5	6	Courtney Kline	2005-08-01 00:00:00.000	548013658	482-7750 Mauris Road	ultrices@aol.org	898456123	10	d	11	14
6	14	Germane Macdonald	2005-09-29 00:00:00.000	549298793	Ap #966-7316 Egestas Avenue	pellentesque@google.ca	885625126	10	d	22	14
7	11	Libby Benson	2005-08-27 00:00:00.000	548270036	711-7319 Nascetur Av.	justo@hotmail.co.uk	882456936	10	d	4	14
8	16	Zephania Davidson	2005-11-21 00:00:00.000	511214785	Ap #727-9127 Odio Ave	cras.interdum.nunc@hotmail.ca	882707577	10	d	19	14

4.2. Заявки върху две и повече релации

-- 1. Заявка, която извежда името на класния ръководител на Vivien Crane.

```
SELECT t.NAME  
FROM STUDENT s, TEACHER t  
WHERE s.NAME = 'Vivien Crane' AND s.FORMTUTOR_ID = t.ID
```

	NAME
1	Daniel Harvey

-- 2. Заявка, която извежда имената на родителите, клас и имена на учениците от всички класове без 12-ти, чието име на ученик съдържа 'а' и ги сортира първо по име на родител и след това по име на ученик.

```
SELECT p.NAME Parent_Name, s.NAME Student_Name, s.SCHOOL_YEAR#  
FROM STUDENT s, PARENT p, STUDENT_PARENT sp  
WHERE UPPER(s.NAME) LIKE UPPER('%a%') AND s.ID = sp.STUDENT_ID AND  
p.EMAIL = sp.PARENT_EMAIL  
EXCEPT  
SELECT p.NAME Parent_Name, s.NAME Student_Name, s.SCHOOL_YEAR#  
FROM STUDENT s, PARENT p, STUDENT_PARENT sp  
WHERE UPPER(s.NAME) LIKE UPPER('%a%') AND s.SCHOOL_YEAR# = 12 AND  
s.ID = sp.STUDENT_ID AND p.EMAIL = sp.PARENT_EMAIL  
ORDER BY p.NAME, s.NAME
```

	Parent_Name	Student_Name	SCHOOL_YEAR#
1	Althea Alexander	Anastasia Pearson	10
2	Athena Neal	Hanae Maxwell	11
3	Athena Neal	Tanya Fulton	5
4	Bert Mccall	Germane Macdonald	10
5	Brenna Wagner	Gannon Collier	11
6	Brenna Wagner	Steven Chambers	5
7	Cameron Delacruz	Kiona Manning	8
8	Daphne Pittman	Caldwell Mclean	10
9	Deirdre Buchanan	Madonna Osborne	9
10	Elvis Kirkland	Mona Hall	7
11	Fulton Shannon	Gay Herman	11
12	Hyacinth Greene	Lacota Sweeney	11
13	Jordan Burnett	Ivana Huffman	4
14	Jordan Burnett	Zelda Bolton	11
15	Kelly Mullen	Bruce James	10
16	Kelly Mullen	Naida Duffy	3
17	Linus Everett	Tatum Mills	4
18	Mercedes Ratliff	Chiquita Ward	4
19	Mercedes Ratliff	Zephania Davidson	10
20	Nicholas Skinner	Hedwig Mann	11
21	Otto Rush	Xander Guy	3
22	Scarlett Neal	Tamara Melendez	3
23	Scarlett Neal	Vivien Crane	11

-- 3. Заявка, която извежда имената на родителите, имената на учениците и клас на всички ученици от 11 и 12 клас и ги сортира по клас, име на ученик, име на родител.

```

SELECT p.NAME Parent_Name, s.NAME Student_Name, s.SCHOOL_YEAR#
FROM STUDENT s, PARENT p, STUDENT_PARENT sp
WHERE s.SCHOOL_YEAR# = 11 AND s.ID = sp.STUDENT_ID AND p.EMAIL =
sp.PARENT_EMAIL
UNION
SELECT p.NAME Parent_Name, s.NAME Student_Name , s.SCHOOL_YEAR#
FROM STUDENT s, PARENT p, STUDENT_PARENT sp
WHERE s.SCHOOL_YEAR# = 12 AND s.ID = sp.STUDENT_ID AND p.EMAIL =
sp.PARENT_EMAIL
ORDER BY s.SCHOOL_YEAR#, s.NAME, p.NAME

```

	Parent_Name	Student_Name	SCHOOL_YEAR#
1	Brenna Wagner	Gannon Collier	11
2	Fulton Shannon	Gay Herman	11
3	Athena Neal	Hanae Maxwell	11
4	Nicholas Skinner	Hedwig Mann	11
5	Hyacinth Greene	Lacota Sweeney	11
6	Scarlett Neal	Vivien Crane	11
7	Jordan Burnett	Zelda Bolton	11
8	Otto Rush	Garth Payne	12
9	Linus Everett	Tatiana Gomez	12

-- 4. Заявка, която извежда имената на класните ръководители на учениците от 10 клас и предметите, по които те преподават без повторения.

```
SELECT DISTINCT t.NAME, sb.NAME
```

```
FROM STUDENT s, TEACHER t, SUBJECT sb
```

```
WHERE s.SCHOOL_YEAR# = 10 AND s.FORMTUTOR_ID = t.ID AND t.ID = sb.TEACHER_ID
```

	NAME	NAME
1	Aaron Knight	Geography
2	Veda Farrell	English language

-- 5. Заявка, която извежда имената на учениците, имената на предметите, имената на преподавателите и датата на отсъствията на учениците от 10 клас, чието име съдържа 'а'.

```
SELECT s.NAME Student_Name, a.SUBJECT_NAME Subject_Name, t.NAME Teacher_Name, a.DATE
```

```
FROM STUDENT s, TEACHER t, SUBJECT sb, ABSENCE a
```

```
WHERE s.SCHOOL_YEAR# = 10 AND UPPER(s.NAME) LIKE UPPER('%a%') AND s.ID = a.STUDENT_ID AND a.SUBJECT_NAME = sb.NAME AND a.TEACHER_ID = sb.TEACHER_ID AND sb.TEACHER_ID = t.ID
```

	Student_Name	Subject_Name	Teacher_Name	DATE
1	Caldwell Mclean	Geography	Aaron Knight	2022-06-24 00:00:00.000
2	Anastasia Pearson	Bulgarian language	Dana Moreno	2022-01-25 00:00:00.000
3	Zephania Davidson	English language	Rana Hess	2021-12-12 00:00:00.000
4	Zephania Davidson	English language	Veda Farrell	2022-02-17 00:00:00.000

4.3. Подзаявки

-- 1. Заявка, която извежда имената и дата на раждане на всички ученици, родени по-рано от Gannon Collier и ги сортира по дата.

```
SELECT s.NAME, s.BIRTHDATE
FROM STUDENT s
WHERE s.BIRTHDATE < (SELECT s.BIRTHDATE FROM STUDENT s WHERE
s.NAME = 'Gannon Collier')
ORDER BY BIRTHDATE
```

	NAME	BIRTHDATE
1	Garth Payne	2003-01-17 00:00:00.000
2	Tatiana Gomez	2003-01-23 00:00:00.000
3	Vivien Crane	2004-07-30 00:00:00.000
4	Zelda Bolton	2004-09-09 00:00:00.000
5	Lacota Sweeney	2004-09-15 00:00:00.000
6	Hanae Maxwell	2004-09-23 00:00:00.000
7	Hedwig Mann	2004-11-18 00:00:00.000

-- 2. Заявка, която извежда данните за най-малкия ученик.

```
SELECT *
FROM STUDENT s
WHERE s.BIRTHDATE <= ALL(SELECT BIRTHDATE FROM STUDENT)
```

	ID	NAME	BIRTHDATE	PIN	ADDRESS	EMAIL	PHONE_NUMBER	SCHOOL_YEAR#	CLASS_NAME	CLASS#	FORMTUTOR_ID
1	5	Garth Payne	2003-01-17 00:00:00.000	341171625	Ap #913-9585 Magna Ave	mauris@google.com	888756532	12	a	9	12

-- 3. Заявка, която извежда името рождената дата и имейла на учениците, получили забележка след датата на забележката на Anastasia Pearson.

```
SELECT s.NAME, s.BIRTHDATE, s.EMAIL
FROM STUDENT s, PRAISE_AND_REMARKS pr
WHERE s.ID = pr.STUDENT_ID AND pr.DATE > (SELECT p.DATE FROM STUDENT
s1, PRAISE_AND_REMARKS p WHERE s1.ID = p.STUDENT_ID AND p.TYPE = 'r'
AND s1.NAME = 'Anastasia Pearson')
```

	NAME	BIRTHDATE	EMAIL
1	Gannon Collier	2004-12-05 00:00:00.000	donec.egestas.aliquam@yahoo.edu
2	Ivana Huffman	2011-02-01 00:00:00.000	at@icloud.com

-- 4. Заявка, която извежда имената на учениците, които имат поне една похвала и ги сортира

```
SELECT DISTINCT s.NAME
FROM STUDENT s
```

WHERE s.ID IN (SELECT p.STUDENT_ID FROM PRAISE_AND_REMARKS p WHERE p.TYPE = 'p')

ORDER BY s.NAME

	NAME
1	Emerson Lyons
2	Gay Herman
3	Naida Duffy
4	Tatiana Gomez

-- 5. Заявка, която извежда имената на класните ръководители на ученици, които имат поне една 6-ца без повторения

SELECT DISTINCT t.NAME

FROM TEACHER t, (select FORMTUTOR_ID FROM STUDENT s, STUDENT_GRADES g WHERE s.ID = g.STUDENT_ID AND g.GRADE_VALUE = 6) st

WHERE t.ID = st.FORMTUTOR_ID

	NAME
1	Aaron Knight
2	Aileen Lott
3	Daniel Harvey
4	Griffith Summers
5	Hall Ortiz
6	Igor Mayo
7	Len Chan
8	Veda Farrell
9	Virginia Noble

4.4. Съединения

--1. Заявка, която извежда имената и оценките по съответен предмет на всички ученици от 12А клас.

SELECT STUDENT.NAME AS 'STUDENT NAME',
STUDENT_GRADES.SUBJECT_NAME, STUDENT_GRADES.GRADE_VALUE

FROM STUDENT JOIN STUDENT_GRADES ON STUDENT.ID =
STUDENT_GRADES.STUDENT_ID

AND STUDENT.SCHOOL_YEAR# = 12 AND STUDENT.CLASS_NAME = 'a';

	STUDENT NAME	SUBJECT_NAME	GRADE_VALUE
1	Tatiana Gomez	English language	6.00
2	Tatiana Gomez	IT	4.00
3	Garth Payne	Geography	3.00

--2. Заявка, която извежда имената на учениците с похвала

```

SELECT STUDENT.NAME AS 'STUDENT NAME',
PRAISE_AND_REMARKS.COMMENT

FROM STUDENT JOIN PRAISE_AND_REMARKS ON STUDENT.ID =
PRAISE_AND_REMARKS.STUDENT_ID

AND PRAISE_AND_REMARKS.TYPE = 'p';

```

	STUDENT NAME	COMMENT
1	Tatiana Gomez	The student assists experiments very well!
2	Gay Herman	The student have excelent skills in analysing tim...
3	Naida Duffy	The student shows fluent skills in "listening" and ...
4	Naida Duffy	The student knows all of the learned songs perfe...
5	Emerson Lyons	The student have really good articulation!

--3. Заявка, която извежда имената и класа на учениците в гимназиален етап на обучение, както и имената на родителите им.

```

SELECT STUDENT.NAME AS 'STUDENT NAME', STUDENT.SCHOOL_YEAR#,
STUDENT.CLASS_NAME, PARENT.NAME AS 'PARENT NAME'

FROM STUDENT JOIN STUDENT_PARENT ON STUDENT.ID =
STUDENT_PARENT.STUDENT_ID AND STUDENT.SCHOOL_YEAR# > 7 JOIN
PARENT ON STUDENT_PARENT.PARENT_EMAIL = PARENT.EMAIL

ORDER BY STUDENT.SCHOOL_YEAR#, STUDENT.CLASS_NAME;

```

	STUDENT NAME	SCHOOL_YEAR#	CLASS_NAME	PARENT NAME
1	Kiona Manning	8	g	Cameron Delacruz
2	Madonna Osborne	9	c	Deirdre Buchanan
3	Bruce James	10	d	Kelly Mullen
4	Caldwell Mclean	10	d	Daphne Pittman
5	Courtney Kline	10	d	Lawrence Hunt
6	Libby Benson	10	d	Kibo Forbes
7	Germane Macdonald	10	d	Bert Mccall
8	Zephania Davidson	10	d	Mercedes Ratliff
9	Anastasia Pearson	10	f	Althea Alexander
10	Cleo Merritt	10	f	Rebecca Newton
11	Hanae Maxwell	11	b	Athena Neal
12	Zelda Bolton	11	b	Jordan Burnett
13	Gannon Collier	11	b	Brenna Wagner
14	Gay Herman	11	b	Fulton Shannon
15	Hedwig Mann	11	b	Nicholas Skinner
16	Vivien Crane	11	b	Scarlett Neal
17	Lacota Sweeney	11	b	Hyacinth Greene
18	Garth Payne	12	a	Otto Rush
19	Tatiana Gomez	12	a	Linus Everett

--4. Заявка, която извежда имената на учениците, изучаващи съответен предмет и неговото наименование.

```
SELECT DISTINCT STUDENT.NAME AS 'STUDENT NAME', SUBJECT.NAME AS  
'SUBJECT'
```

```
FROM STUDENT JOIN TEACHER ON STUDENT.FORMTUTOR_ID = TEACHER.ID  
JOIN SUBJECT ON TEACHER.ID = SUBJECT.TEACHER_ID
```

```
UNION
```

```
SELECT DISTINCT STUDENT.NAME AS 'STUDENT NAME',  
STUDENT_GRADES.SUBJECT_NAME
```

```
FROM STUDENT JOIN STUDENT_GRADES ON STUDENT.ID =  
STUDENT_GRADES.STUDENT_ID
```

```
UNION
```

```
SELECT DISTINCT STUDENT.NAME AS 'STUDENT NAME',  
ABSENCE.SUBJECT_NAME
```

```
FROM STUDENT JOIN ABSENCE ON STUDENT.ID = ABSENCE.STUDENT_ID
```

```
UNION
```

```
SELECT DISTINCT STUDENT.NAME AS 'STUDENT NAME',  
PRAISE_AND_REMARKS.SUBJECT_NAME
```

```
FROM STUDENT JOIN PRAISE_AND_REMARKS ON STUDENT.ID =  
PRAISE_AND_REMARKS.STUDENT_ID
```

```
ORDER BY STUDENT.NAME;
```

	STUDENT NAME	SUBJECT
1	Anastasia Pearson	Bulgarian language
2	Anastasia Pearson	Chemistry
3	Anastasia Pearson	Geography
4	Anastasia Pearson	History
5	Bruce James	Chemistry
6	Bruce James	English language
7	Bruce James	Sport
8	Caldwell Mclean	Bulgarian language
9	Caldwell Mclean	English language
10	Caldwell Mclean	Geography
11	Caldwell Mclean	Physics
12	Chiquita Ward	Biology
13	Chiquita Ward	History
14	Cleo Merritt	Biology
15	Cleo Merritt	Bulgarian language
16	Cleo Merritt	Geography
17	Courtney Kline	Bulgarian language
18	Courtney Kline	English language
19	Courtney Kline	Geography
20	Emerson Lyons	Bulgarian language
21	Emerson Lyons	German language
22	Emerson Lyons	IT
23	Emerson Lyons	Mathematics
24	Gannon Collier	Chemistry
25	Gannon Collier	Physics
26	Garth Payne	Geography
27	Garth Payne	History
28	Garth Payne	IT
29	Gay Herman	Chemistry
30	Gay Herman	History
31	Gay Herman	Physics
32	Germane Macdo...	Bulgarian language
33	Germane Macdo...	English language
34	Germane Macdo...	Sport
35	Hanae Maxwell	Chemistry

36	Hanae Maxwell	English language
37	Hanae Maxwell	Physics
38	Hedwig Mann	Chemistry
39	Hedwig Mann	Mathematics
40	Hedwig Mann	Physics
41	Ivana Huffman	Biology
42	Ivana Huffman	Bulgarian language
43	Ivana Huffman	English language
44	Ivana Huffman	History
45	Kiona Manning	Biology
46	Kiona Manning	English language
47	Kiona Manning	Geography
48	Kiona Manning	German language
49	Lacota Sweeney	Chemistry
50	Lacota Sweeney	Physics
51	Libby Benson	English language
52	Libby Benson	Physics
53	Madonna Osborne	Bulgarian language
54	Madonna Osborne	English language
55	Madonna Osborne	Physics
56	Mona Hall	Biology
57	Mona Hall	Bulgarian language
58	Mona Hall	English language
59	Mona Hall	History
60	Mona Hall	IT
61	Mona Hall	Music
62	Mona Hall	Physics
63	Naida Duffy	Bulgarian language
64	Naida Duffy	German language
65	Naida Duffy	Music
66	Steven Chambers	German language
67	Steven Chambers	IT
68	Steven Chambers	Mathematics
69	Tamara Melendez	Music
70	Tanya Fulton	Chemistry

70	Tanya Fulton	Chemistry
71	Tanya Fulton	English language
72	Tanya Fulton	German language
73	Tanya Fulton	History
74	Tanya Fulton	IT
75	Tanya Fulton	Mathematics
76	Tanya Fulton	Music
77	Tatiana Gomez	Chemistry
78	Tatiana Gomez	English language
79	Tatiana Gomez	IT
80	Tatum Mills	Biology
81	Tatum Mills	English language
82	Tatum Mills	German language
83	Tatum Mills	History
84	Tatum Mills	Mathematics
85	Vivien Crane	Chemistry
86	Vivien Crane	IT
87	Vivien Crane	Music
88	Vivien Crane	Physics
89	Xander Guy	English language
90	Xander Guy	Geography
91	Xander Guy	Music
92	Xander Guy	Physics
93	Zelda Bolton	Chemistry
94	Zelda Bolton	Mathematics
95	Zelda Bolton	Physics
96	Zephania Davids...	English language
97	Zephania Davids...	Physics
98	Zoe Horne	German language
99	Zoe Horne	History

--5. Заявка, която извежда имената на учителите и съответната паралелка, на която са класни ръководители в случай, че са такива.

```
SELECT DISTINCT TEACHER.NAME AS 'TEACHER NAME',  
STUDENT.SCHOOL_YEAR#, STUDENT.CLASS_NAME  
FROM TEACHER JOIN STUDENT ON STUDENT.FORMTUTOR_ID = TEACHER.ID  
ORDER BY SCHOOL_YEAR#;
```

	TEACHER NAME	SCHOOL_YEAR#	CLASS_NAME
1	Len Chan	3	c
2	Aileen Lott	4	d
3	Griffith Summers	5	a
4	Igor Mayo	6	b
5	Hall Ortiz	7	e
6	Rana Hess	8	g
7	Hashim Hall	9	c
8	Aaron Knight	10	f
9	Veda Farrell	10	d
10	Daniel Harvey	11	b
11	Virginia Noble	12	a

4.5. Групиране и агрегация

--1. Заявка, която извежда имената на родителите с повече от едно деца, посещаващи училището и броят на децата им.

```
SELECT PARENT.NAME AS 'PARENT NAME', COUNT(STUDENT.ID) AS 'NUMBER  
OF CHILDREN'
```

```
FROM PARENT JOIN STUDENT_PARENT ON PARENT.EMAIL =  
STUDENT_PARENT.PARENT_EMAIL JOIN STUDENT ON  
STUDENT_PARENT.STUDENT_ID = STUDENT.ID
```

```
GROUP BY PARENT.NAME
```

```
HAVING COUNT(STUDENT.ID) > 1
```

```
ORDER BY COUNT(STUDENT.ID) DESC;
```

	PARENT NAME	NUMBER OF CHILDREN
1	Scarlett Neal	3
2	Athena Neal	2
3	Bert Mccall	2
4	Brenna Wagner	2
5	Jordan Burnett	2
6	Kelly Mullen	2
7	Linus Everett	2
8	Mercedes Ratliff	2
9	Otto Rush	2

--2. Заявка, която извежда броя на отсъствията, които даден преподавател е написал в случай, че е написал.

```
SELECT TEACHER.NAME AS 'TEACHER NAME', COUNT(*) AS 'NUMBER OF
ABSENCES'
FROM TEACHER JOIN ABSENCE ON TEACHER.ID = ABSENCE.TEACHER_ID
GROUP BY TEACHER.NAME;
```

	TEACHER NAME	NUMBER OF ABSENCES
1	Aaron Knight	3
2	Aileen Good	3
3	Dana Moreno	1
4	Daniel Harvey	2
5	Ezra Whitley	1
6	Griffith Summers	1
7	Hall Ortiz	1
8	Hashim Hall	2
9	Igor Mayo	1
10	Len Chan	1
11	Rana Hess	3
12	Veda Farrell	1

--3. Заявка, която извежда името на класа с най-много ученици и класния му ръководител, както и броя на учениците в него.

```
SELECT STUDENT.SCHOOL_YEAR#, STUDENT.CLASS_NAME, TEACHER.NAME
'TEACHER NAME', COUNT(STUDENT.ID) AS 'NUMBER OF STUDENTS'
FROM STUDENT JOIN TEACHER ON STUDENT.FORMTUTOR_ID = TEACHER.ID
GROUP BY TEACHER.NAME, STUDENT.SCHOOL_YEAR#,
STUDENT.CLASS_NAME
HAVING COUNT(STUDENT.ID) >= ALL(SELECT COUNT(STUDENT.ID) FROM
STUDENT,
TEACHER WHERE STUDENT.FORMTUTOR_ID = TEACHER.ID
GROUP BY TEACHER.NAME, STUDENT.SCHOOL_YEAR#,
STUDENT.CLASS_NAME);
```

	SCHOOL_YEAR#	CLASS_NAME	TEACHER NAME	NUMBER OF STUDENTS
1	11	b	Daniel Harvey	7

--4. Заявка, която извежда името на най-малкия и най-големия ученик в училището.

```
SELECT STUDENT.NAME AS 'STUDENT NAME', STUDENT.BIRTHDATE
FROM STUDENT
WHERE STUDENT.BIRTHDATE = (SELECT MIN(STUDENT.BIRTHDATE) FROM
STUDENT)
UNION
SELECT STUDENT.NAME AS 'STUDENT NAME', STUDENT.BIRTHDATE
FROM STUDENT
WHERE STUDENT.BIRTHDATE = (SELECT MAX(STUDENT.BIRTHDATE) FROM
STUDENT);
```

	STUDENT NAME	BIRTHDATE
1	Garth Payne	2003-01-17 00:00:00.000
2	Naida Duffy	2012-05-06 00:00:00.000

--5. Заявка, която извежда средния успех на учениците за всеки клас.

```
SELECT STUDENT.SCHOOL_YEAR#, STUDENT.CLASS_NAME,
ROUND(AVG(STUDENT_GRADES.GRADE_VALUE), 2) AS 'AVERAGE GRADE'
FROM STUDENT JOIN STUDENT_GRADES ON STUDENT.ID =
STUDENT_GRADES.STUDENT_ID
GROUP BY STUDENT.SCHOOL_YEAR#, STUDENT.CLASS_NAME
ORDER BY STUDENT.SCHOOL_YEAR#, STUDENT.CLASS_NAME;
```

	SCHOOL_YEAR#	CLASS_NAME	AVERAGE GRADE
1	3	c	5.500000
2	4	d	5.000000
3	5	a	5.500000
4	6	b	6.000000
5	7	e	5.000000
6	9	c	3.500000
7	10	d	5.380000
8	10	f	4.670000
9	11	b	4.810000
10	12	a	4.330000

--6. Заявка, която извежда името на преподавателя, написал най-много б-ци, както и на този, написал най-много слаби оценки.

```
SELECT TEACHER.NAME AS 'TEACHER_NAME',  
STUDENT_GRADES.GRADE_VALUE  
  
FROM TEACHER JOIN STUDENT_GRADES ON TEACHER.ID =  
STUDENT_GRADES.TEACHER_ID AND STUDENT_GRADES.GRADE_VALUE > 5.5  
  
GROUP BY TEACHER.NAME, STUDENT_GRADES.GRADE_VALUE  
  
HAVING COUNT(STUDENT_GRADES.GRADE_VALUE) >= ALL(SELECT  
COUNT(STUDENT_GRADES.GRADE_VALUE)FROM TEACHER  
  
JOIN STUDENT_GRADES ON TEACHER.ID = STUDENT_GRADES.TEACHER_ID  
AND STUDENT_GRADES.GRADE_VALUE > 5.5  
  
GROUP BY TEACHER.NAME, STUDENT_GRADES.GRADE_VALUE)  
  
UNION  
  
SELECT TEACHER.NAME AS 'TEACHER_NAME',  
STUDENT_GRADES.GRADE_VALUE  
  
FROM TEACHER JOIN STUDENT_GRADES ON TEACHER.ID =  
STUDENT_GRADES.TEACHER_ID AND STUDENT_GRADES.GRADE_VALUE < 3  
  
GROUP BY TEACHER.NAME, STUDENT_GRADES.GRADE_VALUE  
  
HAVING COUNT(STUDENT_GRADES.GRADE_VALUE) >= ALL(SELECT  
COUNT(STUDENT_GRADES.GRADE_VALUE)FROM TEACHER  
  
JOIN STUDENT_GRADES ON TEACHER.ID = STUDENT_GRADES.TEACHER_ID  
AND STUDENT_GRADES.GRADE_VALUE < 3  
  
GROUP BY TEACHER.NAME, STUDENT_GRADES.GRADE_VALUE)  
  
ORDER BY STUDENT_GRADES.GRADE_VALUE;
```

	TEACHER_NAME	GRADE_VALUE
1	Daniel Harvey	2.00
2	Veda Farrell	2.00
3	Hall Ortiz	6.00

--7. Заявка, която извежда името на най-строгия учител, базирайки се на броя написани забележки.

```
SELECT TEACHER.NAME AS 'TEACHER_NAME',  
COUNT(PRAISE_AND_REMARKS.COMMENT) AS 'NUMBER OF REMARKS'  
  
FROM PRAISE_AND_REMARKS, TEACHER  
  
WHERE PRAISE_AND_REMARKS.TEACHER_ID = TEACHER.ID AND  
PRAISE_AND_REMARKS.TYPE = 'r'  
  
GROUP BY TEACHER.NAME  
  
HAVING COUNT(PRAISE_AND_REMARKS.COMMENT) >= ALL (SELECT  
COUNT(PRAISE_AND_REMARKS.COMMENT)  
  
FROM PRAISE_AND_REMARKS, TEACHER  
  
WHERE PRAISE_AND_REMARKS.TEACHER_ID = TEACHER.ID AND  
PRAISE_AND_REMARKS.TYPE = 'r'  
  
GROUP BY TEACHER.NAME);
```

	TEACHER_NAME	NUMBER OF REMARKS
1	Daniel Harvey	2
2	Ezra Whitley	2

--8. Заявка, която извежда броя на учителите, преподаващи едни и същи предмети.

```
SELECT COUNT(DISTINCT s1.TEACHER_ID) AS 'NUMBER OF TEACHERS  
TEACHING SAME SUBJECTS'  
  
FROM SUBJECT s1, SUBJECT s2, TEACHER t1, TEACHER t2  
  
WHERE s1.TEACHER_ID = t1.ID AND s2.TEACHER_ID = t2.ID AND s1.NAME =  
s2.NAME AND t1.ID != t2.ID;
```

	NUMBER OF TEACHERS TEACHING SAME SUBJECTS
1	12

--9. Заявка, която извежда имената на учениците с най-висок успех.

```
SELECT STUDENT.NAME AS 'STUDENT_NAME', STUDENT.SCHOOL_YEAR#,
STUDENT.CLASS_NAME, AVG(STUDENT_GRADES.GRADE_VALUE) AS 'VALUE'

FROM STUDENT JOIN STUDENT_GRADES ON STUDENT.ID =
STUDENT_GRADES.STUDENT_ID

GROUP BY STUDENT.NAME, STUDENT.SCHOOL_YEAR#,
STUDENT.CLASS_NAME

HAVING AVG(STUDENT_GRADES.GRADE_VALUE) >= ALL(SELECT
AVG(STUDENT_GRADES.GRADE_VALUE)

FROM STUDENT JOIN STUDENT_GRADES ON STUDENT.ID =
STUDENT_GRADES.STUDENT_ID

GROUP BY STUDENT.NAME, STUDENT.SCHOOL_YEAR#,
STUDENT.CLASS_NAME)

ORDER BY SCHOOL_YEAR#, CLASS_NAME, STUDENT_NAME;
```

	STUDENT_NAME	SCHOOL_YEAR#	CLASS_NAME	VALUE
1	Naida Duffy	3	c	6.000000
2	Zoe Horne	6	b	6.000000
3	Bruce James	10	d	6.000000
4	Courtney Kline	10	d	6.000000
5	Cleo Merritt	10	f	6.000000
6	Lacota Sweeney	11	b	6.000000

--10. Заявка, която извежда името на ученика с най-много отсъствия.

```
SELECT STUDENT.NAME AS 'STUDENT_NAME', COUNT(ABSENCE.DATE) AS
'NUMBER OF ABSENCES'

FROM STUDENT JOIN ABSENCE ON STUDENT.ID = ABSENCE.STUDENT_ID

GROUP BY STUDENT.NAME

HAVING COUNT(ABSENCE.DATE) >= ALL(SELECT COUNT(ABSENCE.DATE)

FROM STUDENT JOIN ABSENCE ON STUDENT.ID = ABSENCE.STUDENT_ID

GROUP BY STUDENT.NAME);
```

	STUDENT_NAME	NUMBER OF ABSENCES
1	Kiona Manning	2
2	Mona Hall	2
3	Tatum Mills	2
4	Xander Guy	2
5	Zephania Davidson	2

5. Заключение върху:

5.1. Подобрения с добавяне на ограничения на ниво таблица

Добавили сме следните ограничения, съответно:

- за дефиниране на първичните ключове на таблиците

```
---/Define primary keys/---  
ALTER TABLE STUDENT ADD CONSTRAINT PK_STUDENT PRIMARY KEY(ID);  
ALTER TABLE TEACHER ADD CONSTRAINT PK_TEACHER PRIMARY KEY(ID);  
ALTER TABLE HISTORY_TEACHER ADD CONSTRAINT PK_H_TEACHER PRIMARY KEY(EventId);  
ALTER TABLE PARENT ADD CONSTRAINT PK_PARENT PRIMARY KEY(EMAIL);  
ALTER TABLE STUDENT_PARENT ADD CONSTRAINT PK_STUDENT_PARENT PRIMARY KEY(STUDENT_ID, PARENT_EMAIL);  
ALTER TABLE SUBJECT ADD CONSTRAINT PK_SUBJECT PRIMARY KEY(NAME, TEACHER_ID);  
ALTER TABLE STUDENT_GRADES ADD CONSTRAINT PK_GRADE PRIMARY KEY( STUDENT_ID, SUBJECT_NAME, DATE);  
ALTER TABLE ABSENCE ADD CONSTRAINT PK_ABSENCE PRIMARY KEY( STUDENT_ID, SUBJECT_NAME, DATE);  
ALTER TABLE PRAISE_AND_REMARKS ADD CONSTRAINT PK_PRAISE_AND_REMARKS PRIMARY KEY( STUDENT_ID, SUBJECT_NAME, DATE);
```

- установяване на връзките между таблиците

```
---/Make relations/---  
ALTER TABLE STUDENT_PARENT ADD CONSTRAINT FK_STUDENT_PARENT_STUDENT FOREIGN KEY(STUDENT_ID) REFERENCES STUDENT(ID);  
ALTER TABLE STUDENT_PARENT ADD CONSTRAINT FK_STUDENT_PARENT_PARENT FOREIGN KEY(PARENT_EMAIL) REFERENCES PARENT(EMAIL);  
  
ALTER TABLE STUDENT ADD CONSTRAINT FK_STUDENT_TEACHER FOREIGN KEY(FORMTUTOR_ID) REFERENCES TEACHER(ID);  
  
ALTER TABLE SUBJECT ADD CONSTRAINT FK_SUBJECT_TEACHER FOREIGN KEY(TEACHER_ID) REFERENCES TEACHER(ID);  
  
ALTER TABLE ABSENCE ADD CONSTRAINT FK_ABSENCE_SUBJECT FOREIGN KEY(SUBJECT_NAME, TEACHER_ID) REFERENCES SUBJECT(NAME, TEACHER_ID);  
ALTER TABLE ABSENCE ADD CONSTRAINT FK_ABSENCE_STUDENT FOREIGN KEY(STUDENT_ID) REFERENCES STUDENT(ID);  
  
ALTER TABLE PRAISE_AND_REMARKS ADD CONSTRAINT FK_PRAISE_AND_REMARKS_SUBJECT FOREIGN KEY(SUBJECT_NAME, TEACHER_ID) REFERENCES SUBJECT(NAME, TEACHER_ID);  
ALTER TABLE PRAISE_AND_REMARKS ADD CONSTRAINT FK_PRAISE_AND_REMARKS_STUDENT FOREIGN KEY(STUDENT_ID) REFERENCES STUDENT(ID);  
  
ALTER TABLE STUDENT_GRADES ADD CONSTRAINT FK_STUDENT_GRADES_SUBJECT FOREIGN KEY(SUBJECT_NAME, TEACHER_ID) REFERENCES SUBJECT(NAME, TEACHER_ID);  
ALTER TABLE STUDENT_GRADES ADD CONSTRAINT FK_STUDENT_GRADES_STUDENT FOREIGN KEY(STUDENT_ID) REFERENCES STUDENT(ID);
```

5.2. Доразвиване на системата с добавяне на:

5.2.1. Изгледи

--Изглед, представящ имената на учениците, заедно с имената на родителите им и адреса, на който живеят

GO

CREATE VIEW student_parent_view

AS

SELECT s.name as student_name, p.name as parent_name, p.address

FROM student s inner JOIN STUDENT_PARENT sp

ON s.id = sp.STUDENT_ID

inner join parent p on p.email = sp.PARENT_EMAIL;

GO

	student_name	parent_name	address
1	Anastasia Pearson	Althea Alexander	Ap #763-1691 Tincidunt Rd.
2	Bruce James	Kelly Mullen	545-3900 Leo Avenue
3	Caldwell Mclean	Daphne Pittman	P.O. Box 503, 2540 Porttitor Street
4	Chiquita Ward	Mercedes Ratliff	Ap #727-9127 Odio Ave
5	Cleo Merritt	Rebecca Newton	Ap #374-4028 Malesuada Av.
6	Courtney Kline	Lawrence Hunt	482-7750 Mauris Road
7	Emerson Lyons	Scarlett Neal	Ap #784-6500 Nulla. Rd.
8	Gannon Collier	Brenna Wagner	4032 Malesuada Av.
9	Garth Payne	Otto Rush	Ap #913-9585 Magna Ave
10	Gay Herman	Fulton Shannon	Ap #217-2853 Eget Avenue
11	Germane Macdonald	Bert Mccall	Ap #966-7316 Egestas Avenue
12	Hanae Maxwell	Athena Neal	P.O. Box 291, 9079 Eros. St.
13	Hedwig Mann	Nicholas Skinner	Ap #555-2914 Magna. St.
14	Ivana Huffman	Jordan Burnett	P.O. Box 330, 6911 Volutpat. Rd.
15	Kiona Manning	Cameron Delacruz	758-2849 Bibendum St.
16	Lacota Sweeney	Hyacinth Greene	Ap #448-2296 Luctus Rd.
17	Libby Benson	Kibo Forbes	711-7319 Nascetur Av.
18	Madonna Osborne	Deirdre Buchanan	Ap #896-2184 Nullam Av.
19	Mona Hall	Elvis Kirkland	195-917 Dui. St.
20	Naida Duffy	Kelly Mullen	545-3900 Leo Avenue
21	Steven Chambers	Brenna Wagner	4032 Malesuada Av.
22	Tamara Melendez	Scarlett Neal	Ap #784-6500 Nulla. Rd.
23	Tanya Fulton	Athena Neal	P.O. Box 291, 9079 Eros. St.
24	Tatiana Gomez	Linus Everett	Ap #325-3327 Velit. Rd.
25	Tatum Mills	Linus Everett	Ap #325-3327 Velit. Rd.
26	Vivien Crane	Scarlett Neal	Ap #784-6500 Nulla. Rd.
27	Xander Guy	Otto Rush	Ap #913-9585 Magna Ave
28	Zelda Bolton	Jordan Burnett	P.O. Box 330, 6911 Volutpat. Rd.
29	Zephania Davidson	Mercedes Ratliff	Ap #727-9127 Odio Ave
30	Zoe Horne	Bert Mccall	Ap #966-7316 Egestas Avenue

--Изглед, представящ имената на учениците и съответен предмет, който изучават

GO

CREATE VIEW student_subject_view

AS

SELECT DISTINCT STUDENT.NAME AS 'STUDENT_NAME', SUBJECT.NAME AS 'SUBJECT'

FROM STUDENT JOIN TEACHER ON STUDENT.FORMTUTOR_ID = TEACHER.ID

JOIN SUBJECT ON TEACHER.ID = SUBJECT.TEACHER_ID

UNION

SELECT DISTINCT STUDENT.NAME AS 'STUDENT_NAME',
STUDENT_GRADES.SUBJECT_NAME AS 'SUBJECT'

FROM STUDENT JOIN STUDENT_GRADES ON STUDENT.ID =
STUDENT_GRADES.STUDENT_ID

UNION

SELECT DISTINCT STUDENT.NAME AS 'STUDENT_NAME',
ABSENCE.SUBJECT_NAME AS 'SUBJECT'

FROM STUDENT JOIN ABSENCE ON STUDENT.ID = ABSENCE.STUDENT_ID

UNION

SELECT DISTINCT STUDENT.NAME AS 'STUDENT_NAME',
PRAISE_AND_REMARKS.SUBJECT_NAME AS 'SUBJECT'

FROM STUDENT JOIN PRAISE_AND_REMARKS ON STUDENT.ID =
PRAISE_AND_REMARKS.STUDENT_ID

GO

	STUDENT_NAME	SUBJECT
1	Anastasia Pearson	Bulgarian language
2	Anastasia Pearson	Chemistry
3	Anastasia Pearson	Geography
4	Anastasia Pearson	History
5	Bruce James	Chemistry
6	Bruce James	English language
7	Bruce James	Sport
8	Caldwell Mclean	Bulgarian language
9	Caldwell Mclean	English language
10	Caldwell Mclean	Geography
11	Caldwell Mclean	Physics
12	Chiquita Ward	Biology
13	Chiquita Ward	History
14	Cleo Merritt	Biology
15	Cleo Merritt	Bulgarian language
16	Cleo Merritt	Geography
17	Courtney Kline	Bulgarian language
18	Courtney Kline	English language
19	Courtney Kline	Geography
20	Emerson Lyons	Bulgarian language
21	Emerson Lyons	German language
22	Emerson Lyons	IT
23	Emerson Lyons	Mathematics
24	Gannon Collier	Chemistry
25	Gannon Collier	Physics
26	Garth Payne	Geography
27	Garth Payne	History
28	Garth Payne	IT
29	Gay Herman	Chemistry
30	Gay Herman	History
31	Gay Herman	Physics
32	Germane Macdo...	Bulgarian language
33	Germane Macdo...	English language
34	Germane Macdo...	Sport
35	Hanae Maxwell	Chemistry

36	Hanae Maxwell	English language
37	Hanae Maxwell	Physics
38	Hedwig Mann	Chemistry
39	Hedwig Mann	Mathematics
40	Hedwig Mann	Physics
41	Ivana Huffman	Biology
42	Ivana Huffman	Bulgarian language
43	Ivana Huffman	English language
44	Ivana Huffman	History
45	Kiona Manning	Biology
46	Kiona Manning	English language
47	Kiona Manning	Geography
48	Kiona Manning	German language
49	Lacota Sweeney	Chemistry
50	Lacota Sweeney	Physics
51	Libby Benson	English language
52	Libby Benson	Physics
53	Madonna Osborne	Bulgarian language
54	Madonna Osborne	English language
55	Madonna Osborne	Physics
56	Mona Hall	Biology
57	Mona Hall	Bulgarian language
58	Mona Hall	English language
59	Mona Hall	History
60	Mona Hall	IT
61	Mona Hall	Music
62	Mona Hall	Physics
63	Naida Duffy	Bulgarian language
64	Naida Duffy	German language
65	Naida Duffy	Music
66	Steven Chambers	German language
67	Steven Chambers	IT
68	Steven Chambers	Mathematics
69	Tamara Melendez	Music
70	Tanya Fulton	Chemistry

71	Tanya Fulton	English language
72	Tanya Fulton	German language
73	Tanya Fulton	History
74	Tanya Fulton	IT
75	Tanya Fulton	Mathematics
76	Tanya Fulton	Music
77	Tatiana Gomez	Chemistry
78	Tatiana Gomez	English language
79	Tatiana Gomez	IT
80	Tatum Mills	Biology
81	Tatum Mills	English language
82	Tatum Mills	German language
83	Tatum Mills	History
84	Tatum Mills	Mathematics
85	Vivien Crane	Chemistry
86	Vivien Crane	IT
87	Vivien Crane	Music
88	Vivien Crane	Physics
89	Xander Guy	English language
90	Xander Guy	Geography
91	Xander Guy	Music
92	Xander Guy	Physics
93	Zelda Bolton	Chemistry
94	Zelda Bolton	Mathematics
95	Zelda Bolton	Physics
96	Zephania Davids...	English language
97	Zephania Davids...	Physics
98	Zoe Horne	German language
99	Zoe Horne	History

--Изглед, представящ имената на учениците и съответният им класен ръководител

GO

CREATE VIEW student_formtutor_view

AS

SELECT STUDENT.NAME as student_name, TEACHER.NAME as teacher_name

FROM STUDENT JOIN TEACHER ON STUDENT.FORMTUTOR_ID = TEACHER.ID

GO

	student_name	teacher_name
1	Anastasia Pearson	Aaron Knight
2	Bruce James	Veda Farrell
3	Caldwell Mclean	Veda Farrell
4	Chiquita Ward	Aileen Lott
5	Cleo Merritt	Aaron Knight
6	Courtney Kline	Veda Farrell
7	Emerson Lyons	Griffith Summers
8	Gannon Collier	Daniel Harvey
9	Garth Payne	Virginia Noble
10	Gay Herman	Daniel Harvey
11	Germane Macdo...	Veda Farrell
12	Hanae Maxwell	Daniel Harvey
13	Hedwig Mann	Daniel Harvey
14	Ivana Huffman	Aileen Lott
15	Kiona Manning	Rana Hess
16	Lacota Sweeney	Daniel Harvey
17	Libby Benson	Veda Farrell
18	Madonna Osborne	Hashim Hall
19	Mona Hall	Hall Ortiz
20	Naida Duffy	Len Chan
21	Steven Chambers	Griffith Summers
22	Tamara Melendez	Len Chan
23	Tanya Fulton	Griffith Summers
24	Tatiana Gomez	Virginia Noble
25	Tatum Mills	Aileen Lott
26	Vivien Crane	Daniel Harvey
27	Xander Guy	Len Chan
28	Zelda Bolton	Daniel Harvey
29	Zephania Davids...	Veda Farrell
30	Zoe Horne	Igor Mayo

5.2.2. Индекси

```
create unique index student_index on student(pin);
create unique index parent_email_index on parent(email);
create unique index parent_phone_index on parent(PHONE_NUMBER);
create unique index absence_index on ABSENCE(STUDENT_ID, TEACHER_ID);
create unique index praise_and_remarks_index on PRAISE_AND_REMARKS(STUDENT_ID, TEACHER_ID);
create unique index student_grades_index on STUDENT_GRADES(STUDENT_ID, SUBJECT_NAME);
```

5.2.3. Тригери

--Тригер за добавяне на нов преподавател

```
GO
if exists (select 1
           from sys.objects
           where type = 'TR'
                 and name = 'teacher_history_insert')
    drop trigger teacher_history_insert;
GO

CREATE TRIGGER teacher_history_insert
ON TEACHER
AFTER INSERT
AS
BEGIN
    INSERT INTO HISTORY_TEACHER(TeacherId, EventType, BeginDate, Name)
    SELECT
        i.ID,
        'insert',
        GETDATE(),
        i.Name
    FROM inserted i;
END;
```

--Тригер за добавяне на нов учебен предмет

```
--
GO
if exists (select 1
          from sys.objects
          where type = 'TR'
                and name = 'subject_history_insert')
    drop trigger subject_history_insert;
GO

CREATE TRIGGER subject_history_insert
ON SUBJECT
AFTER INSERT
AS
BEGIN
    UPDATE HISTORY_TEACHER set SubjectName = (select name from inserted)
    where teacherid = (select i.teacher_id from inserted as i);
END;
```

--Тригер за обновяване на данни за конкретен преподавател

```
GO
if exists (select 1
          from sys.objects
          where type = 'TR'
                and name = 'teacher_history_update')
    drop trigger teacher_history_update;
GO

CREATE TRIGGER teacher_history_update
ON TEACHER
AFTER UPDATE
AS
BEGIN
    INSERT INTO HISTORY_TEACHER(TeacherId, EventType, BeginDate, Name)
    SELECT
        i.ID,
        'update',
        GETDATE(),
        i.Name
    FROM inserted i;
END;
```

--Тригер за промяна на адрес на даден родител

```
if exists (select 1
           from sys.objects
           where type = 'TR'
                and name = 'parent_address_update')
    drop trigger parent_address_update;
GO

CREATE TRIGGER parent_address_update
ON PARENT
AFTER UPDATE
AS
BEGIN
    DECLARE @OldAddress AS varchar(255)
    SET @OldAddress = (select address from deleted);
    DECLARE @NewAddress AS varchar(255)
    SET @NewAddress = (select address from inserted);
    PRINT @OldAddress
    print @NewAddress
    if(@OldAddress = @NewAddress)
        return;
    UPDATE STUDENT set address = (select i.address from inserted as i)
    where id in (select sp.student_id from STUDENT_PARENT sp
                where sp.PARENT_EMAIL = (select email from inserted));
END;
```

6. Оценка на реализацията, препоръки и възможни подобрения

- Като оценка на реализацията можем да отбележим, че базата ни е ефективна, задоволява нуждите на заведения с насоченост в сферата на образованието, предоставя възможности за бързо и лесно модифициране и е построена спрямо нужните изисквания.
- Възможно подобрение на системата е справянето с проблема премахване на редове от таблиците, което само по себе си води до преправяне на цялата структурна логика на релациите между отделните таблици, тъй като крайната зависимост на всички едни от други на този етап възпрепятства премахването на даден ключов компонент.
- Като допълнителни подобрения можем да разширим идейната цел на базата и да включим нови графи с 'Извънкласни дейности', също и да бъдат посочени различни роли на преподавателите, могат да се добавят възможности за съхраняване на предадени от учениците материали, като по този начин се следи тяхната активност и учебно разписание за всеки от отделните класове.