


```
        ON DELETE CASCADE

);

CREATE TABLE Number(
    name VARCHAR(30) PRIMARY KEY,
    fax VARCHAR(30),
    phone VARCHAR(20) NOT NULL,
    CONSTRAINT number_ch_phone CHECK (phone not like '%[^0-9]%),
    CONSTRAINT number_fk_name FOREIGN KEY (name)
    REFERENCES LanguageSchool(name)
    ON UPDATE CASCADE
    ON DELETE CASCADE
);
```

```
CREATE TABLE Teacher(
    langSchoolCardNumber CHAR(8) PRIMARY KEY,
    CONSTRAINT teacher_fk_langSchoolCardNumber FOREIGN KEY (langSchoolCardNumber)
    REFERENCES Person(langSchoolCardNumber)
    ON UPDATE CASCADE
    ON DELETE CASCADE
);
```

```
CREATE TABLE TeacherAccount(
    login VARCHAR(25) PRIMARY KEY
);
```

```
CREATE TABLE TeacherAccountOwnerShip(
    teacherId CHAR(8) PRIMARY KEY,
    account VARCHAR(25) NOT NULL UNIQUE,
    CONSTRAINT teacherAccountOwnerShip_fk_teacherId FOREIGN KEY (teacherId)
    REFERENCES Teacher(langSchoolCardNumber)
    ON UPDATE CASCADE
```

```
ON DELETE CASCADE,  
CONSTRAINT teacherAccountOwnerShip_fk_account FOREIGN KEY (account)  
REFERENCES TeacherAccount(login)  
ON UPDATE CASCADE  
ON DELETE CASCADE  
);
```

```
CREATE TABLE Student(  
    langSchoolCardNumber CHAR(8) PRIMARY KEY,  
    startingLanguageLevel CHAR(2) NOT NULL,  
    CONSTRAINT student_fk_langSchoolCardNumber FOREIGN KEY (langSchoolCardNumber)  
    REFERENCES Person(langSchoolCardNumber)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE  
);
```

```
CREATE TABLE Invitation(  
    studentId CHAR(8) PRIMARY KEY,  
    friendId CHAR(8) NOT NULL UNIQUE,  
    CONSTRAINT invitation_fk_studentId FOREIGN KEY (studentId)  
    REFERENCES Student(langSchoolCardNumber)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE,  
    CONSTRAINT invitation_fk_friendId FOREIGN KEY (friendId)  
    REFERENCES Student(langSchoolCardNumber)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE  
);
```

```
CREATE TABLE StudentAccount(  
    login VARCHAR(25) PRIMARY KEY  
);
```

```
CREATE TABLE StudentAccountOwnership(  
    studentId CHAR(8) PRIMARY KEY,  
    account VARCHAR(25) NOT NULL UNIQUE,  
    CONSTRAINT studentAccountOwnership_fk_studentId FOREIGN KEY (studentId)  
    REFERENCES Student(langSchoolCardNumber)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE,  
    CONSTRAINT studentAccountOwnership_fk_account FOREIGN KEY (account)  
    REFERENCES StudentAccount(login)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE  
);
```

```
CREATE TABLE Staff(  
    langSchoolCardNumber CHAR(8) PRIMARY KEY,  
    position VARCHAR(15) NOT NULL,  
    CONSTRAINT staff_fk_langSchoolCardNumber FOREIGN KEY (langSchoolCardNumber)  
    REFERENCES Person(langSchoolCardNumber)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE  
);
```

```
CREATE TABLE Payment(  
    studentId CHAR(8) NOT NULL,  
    number CHAR(15) NOT NULL,  
    method VARCHAR(6) NOT NULL,  
    PRIMARY KEY(studentId, number),  
    CONSTRAINT payment_fk_studentId FOREIGN KEY (studentId)  
    REFERENCES Student(langSchoolCardNumber)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE
```

);

```
CREATE TABLE PaymentControl(  
    employeeld CHAR(8) NOT NULL,  
    payment CHAR(15) NOT NULL,  
    studentId CHAR(8) NOT NULL,  
    PRIMARY KEY (employeeld, payment),  
    CONSTRAINT paymentControl_fk_employeeld FOREIGN KEY (employeeld)  
    REFERENCES Staff(langSchoolCardNumber)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE,  
    CONSTRAINT paymentControl_fk_payment FOREIGN KEY (studentId, payment)  
    REFERENCES Payment(studentId, number)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE  
);
```

```
CREATE TABLE StudyGroup (  
    teacherId CHAR(8) NOT NULL,  
    level CHAR(2) NOT NULL,  
    language VARCHAR(10) NOT NULL,  
    PRIMARY KEY(teacherId, level, language),  
    CONSTRAINT group_fk_teacherId FOREIGN KEY (teacherId)  
    REFERENCES Teacher(langSchoolCardNumber)  
    ON UPDATE CASCADE  
    ON DELETE CASCADE  
);
```

```
CREATE TABLE StudyGroupCreation(  
    employeeld VARCHAR(8) NOT NULL,  
    teacherId VARCHAR(8) NOT NULL,  
    level CHAR(2) NOT NULL,
```

```

language VARCHAR(10) NOT NULL,
PRIMARY KEY(employeeId, teacherId, level, language),
CONSTRAINT studyGroupCreation_fk_employee FOREIGN KEY (employeeId)
REFERENCES Staff(langSchoolCardNumber)
ON UPDATE CASCADE
ON DELETE CASCADE,
CONSTRAINT studyGroupCreation_fk_teacher_level_lang FOREIGN KEY (teacherId, level,
language)
REFERENCES StudyGroup(teacherId, level, language)
ON UPDATE CASCADE
ON DELETE CASCADE
);

```

```

CREATE TABLE Lesson(
    studentId VARCHAR(8) NOT NULL,
    teacherId VARCHAR(8) NOT NULL,
    level CHAR(2) NOT NULL,
    language VARCHAR(10) NOT NULL,
    PRIMARY KEY(studentId, teacherId, level, language),
    CONSTRAINT lesson_fk_studentId FOREIGN KEY (studentId)
    REFERENCES Student(langSchoolCardNumber)
    ON UPDATE CASCADE
    ON DELETE CASCADE,
    CONSTRAINT lesson_fk_teacher_level_lang FOREIGN KEY (teacherId, level, language)
    REFERENCES StudyGroup(teacherId, level, language)
    ON UPDATE CASCADE
    ON DELETE CASCADE
);

```

```

CREATE TABLE Website(
    domain VARCHAR(50) PRIMARY KEY,
    teacherAccount VARCHAR(25) NOT NULL,
    studentAccount VARCHAR(25) NOT NULL,

```

```

CONSTRAINT website_fk_teacherAccount FOREIGN KEY (teacherAccount)
REFERENCES TeacherAccount(login)
ON UPDATE CASCADE
ON DELETE CASCADE,
CONSTRAINT website_fk_studentAccount FOREIGN KEY (studentAccount)
REFERENCES StudentAccount(login)
ON UPDATE CASCADE
ON DELETE CASCADE
);

```

```

CREATE TABLE WebsiteOwnership(
    school VARCHAR(30) NOT NULL UNIQUE,
    domain VARCHAR(50) PRIMARY KEY,
    CONSTRAINT websiteOwnership_fk_school FOREIGN KEY (school)
    REFERENCES LanguageSchool(name)
    ON UPDATE CASCADE
    ON DELETE CASCADE,
    CONSTRAINT websiteOwnership_fk_domain FOREIGN KEY (domain)
    REFERENCES Website(domain)
    ON UPDATE CASCADE
    ON DELETE CASCADE
);

```


DATA QUERIES.

- 1) All teachers' names in ascending order (by name);

```

SELECT Teacher.*, Person.name
FROM Teacher
LEFT JOIN Person ON
(Teacher.langSchoolCardNumber = Person.langSchoolCardNumber)
ORDER BY name;

```


	Data Output	Explain	Messages	Notifications
	 langschoolcardnumber character (8)		name character varying (20)	
1	89284302		Bartel Carbin	
2	68087772		Buddie Hayball	
3	41361017		Codie Hylden	
4	81522030		Flossie Butcher	
5	02678669		Garrard Whitwood	
6	14979929		Gus Nordass	
7	27587298		Iona Gymlett	
8	19221167		Laverne Aistrop	
9	28010606		Lizette Timmis	
10	71722865		Wilona Dullaghan	

2) Data of students who study French language on B1 level

```

SELECT Person.name, Person.street, Person.email
FROM Person
WHERE
    Person.langSchoolCardNumber in (
        SELECT Student.langSchoolCardNumber
        FROM Student
        INNER JOIN Lesson ON (Student.langSchoolCardNumber = Lesson.studentId)
        WHERE (Lesson.level = 'B1') AND (Lesson.language = 'French')
    );

```

	Data Output	Explain	Messages	Notifications
	 name character varying (20)		street character varying (50)	email character varying (50)
1	Alic Gambles		34423 Forster Lane	agamblesc@washington.edu

3) Number of groups each teacher teaches.

```

SELECT StudyGroup.teacherid, COUNT(StudyGroup.level)
FROM StudyGroup
GROUP BY StudyGroup.teacherid;

```


	Data Output	Explain	Messages	Notifications
	teacherid character (8)	count bigint		
1	27587298	4		
2	41361017	1		
3	89284302	1		
4	68087772	1		
5	19221167	2		
6	14979929	1		
7	02678669	4		
8	81522030	1		
9	71722865	2		
10	28010606	1		

4) Data of people whose name does not start with 'A'.

SELECT Person.*

FROM Person

EXCEPT

SELECT Person.*

FROM Person

WHERE (Person.name LIKE 'A%')

	passport character varying (20)	langschoolcardnumber character (8)	name character varying (20)	street character varying (50)	email character varying (50)
1	925855502	68087772	Buddie Hayball	74249 Tennyson Crossing	bhayball5@jimdo.com
2	673123041	14979929	Gus Nordass	608 Rigney Way	gnordass0@psu.edu
3	542423519	82829705	Claudian Wallbrook	8135 Blaine Trail	cwallbrooke@auda.org.au
4	696800118	71722865	Wilona Dullaghan	5835 Dixon Plaza	wdullaghan6@columbia.edu
5	546433334	81522030	Flossie Butcher	495 Annamark Road	fbutcher2@blog.com
6	575611545	02678669	Garrard Whitwood	5 North Hill	gwhitwood9@woothemes.com
7	658084913	28010606	Lizette Timmis	1 Stoughton Crossing	ltimmis1@indiegogo.com
8	912111839	27587298	Iona Gymlett	56 Haas Center	igymlett8@behance.net
9	887647045	89284302	Bartel Carbin	3152 Prentice Avenue	bcarbin4@qq.com
10	656908828	61078773	Lindsey Crock	25 Scoville Road	lcrockf@blog.com
11	883687516	46831570	Murial Pessolt	2948 Maple Wood Park	mpessolt@stanford.edu
12	781367937	41361017	Codie Hylden	9 Logan Trail	chylden3@about.me
13	816584317	34105645	Filippa Siccombe	5668 Annamark Lane	fsiccombeg@theatlantic.com
14	645750906	02255540	Mada Henlev	54 Arizona Circle	mhanlevi@skype.com