

# Database Design for the Bright Future School



By

**Cloud Tech Crew**

Jana Jaber

202001222

&

Hadi Bazzi

202004224

## **Introduction:**

Cloud-Tech is a new database company established recently by Jana Jaber, Hadi Bazzi and Mohammad Jaber. The main goal of this company is to design and implement efficient databases satisfying the customer's preferences and requirements. A businessman decided to invest in building a new school called Bright Future School. He contacted us to design and implement a useful database for his school. We took his requirements and preferences into consideration and constructed an ER diagram for this database based on specific system description and constraints.

## **System Description and Constraints:**

A school is an educational organization where all people care about when it comes to enrolling their children to get education. Thus, for this sake we intended to design and implement an active functional database.

In a school there are different DEPARTMENTS. Each department has a unique name, unique number, location, fax, number of employees, phone number, and an EMPLOYEE who *manages* a specific DEPARTMENT. We keep track of the start date of department manager.

Each EMPLOYEE has a unique ID, degree(s), name, email, phone number(s), gender, salary, date of birth, address, profession and social security identification.

EMPLOYEE(s) *work for* BRANCHEs. Each branch has a unique branch number, fax, location, and phone number(s). We keep track of the date of which each employee started working for a specific branch.

An EMPLOYEE *supervises* other EMPLOYEE(s).

Each EMPLOYEE *has a* DEPENDENT(s). Each dependent has a dependent ID, name, gender, relationship, and date of birth.

EMPLOYEE *organizes* ACTIVITY(s). Each activity has a unique activity number, type, occasion, expenses, and guests. We keep track of the name(s) of the organizers.

Each ACTIVITY *takes place* in a FACILITY. Each facility has a facility ID, capacity, location, type, and status.

A TEACHER *is an* EMPLOYEE. Each teacher has an email, teacher ID, name, phone number(s), gender, comments, salary, date of birth, class grade, address, social security identification, language(s), and degree(s).

Each TEACHER *teaches* STUDENT(S). Each student has a name, student ID, gender, warnings, date of birth, address, rank, language, and a class grade.

A STUDENT *goes to* BRANCH.

An EMPLOYEE *supervises* STUDENT(s).

Each STUDENT *has a* PARENT(s). Each parent has a name, profession, unique phone number, date of birth, gender, and an address.

STUDENT(s) *participates in* ACTIVITY(s). We keep track of the number of participants.

Each STUDENT *has an* ACCOUNT. Each account has a unique account number, deposit, remaining fees, payment method and a date.

Each STUDENT *has a* GRADE(s). Each grade has a letter grade, type of grade, grade for each term, and a total.

A GRADE *is of a* SUBJECT(s). Each subject has a language(s), type, unique name, and number of hours.

Each STUDENT *takes a* SUBJECT(s).

A TEACHER *teaches* a SUBJECT(s). We keep track of the number of hours each teacher teaches of a specific subject.

A SUBJECT(s) *is taught* in a FACILITY(s).

A HEALTH RECORD *belongs to* a STUDENT. Each health record has a height, weight, blood type, disease(s), allergy(s), age, unique record ID, and gender.

## **Entity Types:**

### **1) DEPARTMENT(s)**

A school is organized into different departments. There is the academic department (1), health department (2), admission department (3), sports departments (4), etc.... Each department has a phone, fax, number of employees working for it, specific unique number, and a unique name

which serve as two key attributes for this entity. Also, it has a location which is a multi-valued attribute since we may have the same department in different locations.

## 2) STUDENT

Students are an important part of the school. Each student has a unique ID (key attribute), name composed of the (first name, middle name, and last name), gender, date of birth, address to specify where he/she live, the language to specify whether the student is English or French educated, the rank of the student whether he/she is placed on the distinction list or not, in which class grade she/he is studying, and the number of warnings he/she has received (oral or written warnings), which serves as a multi-valued attribute so that one student may have multiple warnings.

## 3) TEACHER

Teachers constitute an essential part of the educational organization. Each teacher has a unique ID (key attribute), a name of which is a composite attribute composed of (first name, middle initial, and last name), gender, date of birth, address to identify his/her residence, the language(s) he/she acquires where she/he can acquire more than one language, as well as the degree(s) he/she holds where he/she may hold multiple degrees, his/her phone number(s) where it may be home phone number or cell phone number (multi-valued attribute also), his/her email, the salary she/he earns, his/her social security identification, the different class grade(s) he/she teaches considering the fact that she/he may be assigned to teach different class grades, and comments (feedback received from students, parents or supervisors), which is also considered as a multi-valued attribute.

## 4) PARENT

Parents are the students' guardians. Each parent has a unique phone number to communicate with him/her in case of any concern regarding his/her child (student) serving as a key attribute, name, gender, date of birth, address of domicile, and his/her profession.

## 5) BRANCH:

The school may have several branches spread over different locations. The location is specified by the zip code, street, and the city where it is centered (composite attribute). Also, it has a key attribute represented by a unique number to differentiate among branches (all have the same school's name), fax, and phone number(s) (multi-valued attribute).

#### 6) ACTIVITY:

The school usually organizes different activities for students. Each activity has a type (sports, technological, social...), unique activity number (key attribute) differentiating distinct activities, guests who may include professionals in certain domains from outside the school, occasion (feast, festivals, different events...), and expenses.

#### 7) HEALTH RECORD:

To ensure a safe environment for students, each student has a health record. The health record has a unique ID as a key attribute and constitutes of the student's height, weight, gender, age, blood type, and the allergies or diseases he/she may suffer from which are considered as multi-valued attributes.

#### 8) ACCOUNT:

To ensure that all students have paid their tuition fees, the account entity is created to keep track of each student's finances. Each account has a unique account number as a key attribute, deposit for which the student has paid, the remaining fees for the student to pay, the payment method whether it is in cash, Lebanese Lira, Dollar, check or a combination of two (multi-valued attribute), date indicating when the account was opened and when it was closed in case a student has left the school or graduated.

#### 9) EMPLOYEE:

The school has several employees working in different domains for example the staff responsible for financial issues, administration, teachers, janitors, security guards, etc.... Each employee has a single ID as a key attribute, name, social security identification, email, profession as mentioned, gender, salary he/she earns, date of birth, address, degree he/she carries where it is possible to

carry several degrees or no degree (multi-valued attribute), and a phone number(s) which as well is considered as a multi-valued attribute.

#### **10) DEPENDENT:**

To keep track of employees' dependents, we consider the DEPENDENT entity. Each dependent has an individual ID as a key attribute, name, gender, date of birth, and his/her relation to the employee (father, mother, wife, husband, daughter, or son).

#### **11) GRADE:**

To keep track of students' academic performance, GRADE entity is created. Each grade has a total representing the total grade, first term grade, midterm grade, final term grade, and letter grade. This entity has no key attribute, so it is considered a weak entity; however, it has a partial key which is the type attribute to identify the type of the grade (term grades or total grade).

#### **12) SUBJECT:**

This entity was created to support the existence of different subjects. Each subject has a specific name made up of the subject name and the class grade which combined make up the key attribute of this entity, type of this subject (humanity, art, science, etc....), hour number to specify the number of hours this subject is given weekly, and language where the same subject can be given in one or two languages (multi-valued attribute).

#### **13) FACILITY:**

There are many facilities in the school including classrooms, playground, offices, labs, or theatres. Each facility has a unique site ID serving as a key attribute, capacity which indicates how many people this facility can handle, the status whether the facility is reserved or not at a specific time, type of the facility as mentioned previously, and a location.

### **Relationship Types:**

1) HAS\_A1:

All students have grades on first term, midterm and final term, and a total grade with a letter grade. Thus, a “HAS\_A1” relationship type is created between STUDENT and GRADES. The cardinality ratio is N to M since one student can have many grades, and one grade can be identical for many students. It is a total participation since all students will have grades and all grades will be given to students.

2) HAS\_A2:

All students have parents. Thus, a “HAS\_A2” relationship type is created between STUDENT and PARENT. It is a total participation relationship since all students have parents (or any guardian) and all parents have students enrolled in the school. The cardinality is N to M since one student may have two parents (mother and father), and one parent may have several children registered in the school.

3) HAS\_A3:

All employees have dependents. So, a HAS\_A3 relationship type is created between EMPLOYEE and DEPENDENT. The cardinality is 1 to N since one employee may have many dependents, but not the opposite. All dependents are related to a specific employee, but not all employees have dependents (Participation constraint).

4) HAS\_AN:

All students have an account to state their tuition fees and the amount paid. Thus, a HAS\_AN relationship type is created between STUDENT and ACCONT. The cardinality ratio is 1 to 1 since each student has one account and vice versa. The participation is total since all students have an account and all the accounts belong to students.

5) IS\_AN:

All the teachers are employees in the school. Thus, IS\_AN relationship type is created between TEACHER and EMPLOYEE. The cardinality ratio is 1 to 1 since each teacher can be one employee and vice versa. The participation is partial since all teachers are employees, but not all employees are teachers.

6) IS\_OF\_A:

The grades given for students are related to specific subject. Thus, IS\_OF\_A relationship type is created between GRADE and SUBJECT. The cardinality ratio is N to M since the same grade may belong to many subjects, and one subject may have many grades. It is a total participation since all grades have a subject and all subjects must have a grade.

7) PARTICIPATES\_IN:

At the school there are activities in which students participate in. Thus, a PARTICIPATES\_IN relationship type is created between STUDENT and ACTIVITY. The cardinality ratio is N to M since one student may participate in many activities, and one activity may include many students participating. The participation is partial since not all students participate in an activity; however, all activities require students' participation.

8) BELONGS\_TO:

Each student has a health care record. So, a BELONGS\_TO relationship type is created between STUDENT and HEALTH\_RECORD. The cardinality ratio is 1 to 1 since one student has one health record and vice versa. The participation is total since all students has health records, and all health records belong to students.

9) SUPERVISES1:

At school an employee may supervise other employees. Thus, a SUPERVISES1 relationship type is created between EMPLOYEE itself. The cardinality is 1 to N since one employee may supervise several employees, but not the opposite. The participation is partial since not all employees are supervisors and not all employees are supervised.

10) SUPERVISES2:

At school students have supervisors. Hence, a SUPERVISES2 relationship type is created between STUDENT and EMPLOYEE. The cardinality is 1 to N since one employee supervises many students, but one student has one supervisor. The participation is partial since not all employees are supervisors, but all students are supervised.

#### 11) ORGANIZES:

Every activity should be organized. Therefore, an ORGANIZES relationship type is created between EMPLOYEE and ACTIVITY. The cardinality is N to M since one activity may be organized by many employees, and one employee may organize more than one activity. The participation is partial because not all employees organize activities, but all activities are organized by employees. This relation has a multivalued attribute “organizers\_name” to keep track of the names of the organizers of each activity.

#### 12) GOES\_TO:

Students are enrolled in different branches of the school. Accordingly, GOES\_TO relationship type is created between STUDENT and BRANCH. The cardinality ratio is 1 to N since one student goes to one branch, but one branch admits many students. The participation is total since all students go to branches, and all branches have students.

#### 13) TAKES\_PLACE\_IN:

Activities are held in different facilities in school. Hence, TAKES\_PLACE\_IN relationship type is created between FACILITY and ACTIVITY. The cardinality ratio is 1 to 1 since one activity takes place in one facility, and one facility can have one activity at a time. The participation is partial because all activities take place in facilities, but not all facilities admit activities.

#### 14) IS\_TAUGHT:

Different subjects are taught in different facility (classes, labs etc....). Thus, IS\_TAUGHT relationship type is created between SUBJECT and FACILITY. The cardinality is N to M since one subject may be taught in different facilities, and many subjects may be taught in one facility.

The participation is partial since all subjects are taught in facilities, but not in all facilities subjects are taught.

15) TAKES:

Students have subjects to study. So, TAKES relationship type is created between STUDENT and SUBJECT. The cardinality is N to M since one subject is given to several students, and one student takes multiple subjects. The participation is total since all students take subjects and vice versa.

16) TEACHES2:

Every teacher is assigned a subject(s) to teach. Thus, TEACHES2 relationship type is created between TEACHER and SUBJECT. The cardinality is N to M since one teacher can teach many subjects, and one subject can be assigned to several teachers. The participation is total since all teachers teach subjects and all subjects are taught by teachers. This relationship type has an attribute “number of hours” to keep track of the number of hours each teacher teaches in a specific subject.

17) TEACHES1:

A TEACHES1 relationship type is created between TEACHER and STUDENT. The cardinality is N to M since one teacher teaches many students, and one student is taught by many teachers. The participation is total because all students have teachers, and all teachers teach students.

18) MANAGES:

Each department is managed. Therefore, MANAGES relationship type is created between DEPARTMENT and EMPLOYEE. The cardinality is 1 to 1 since one department is managed by a single employee, and one employee can manage only one department. The participation is partial since all departments must be managed, but not all employees are managers. This relationship type has “startDate” as an attribute to keep record of the date the employee has started managing a specific department.

19) WORKS\_FOR:

Every employee works for a branch. Thus, WORKS\_FOR relationship type is created between EMPLOYEE and BRANCH. The cardinality is N to M since one employee may work for more than one branch, and one branch has several employees. The participation is total because all employees work for at least one branch, and all branches have employees. This relationship has “startDate” as an attribute to set the date a specific employee started working for a branch.

## **ER-to-Relational Mapping Algorithm:**

After designing an ER diagram for the Bright Future School database specifying different entity types, attributes, and relationship types, a relational model for this database must be designed through a seven-step procedure. The following is a detailed description of applying the different steps to our relational database design.

### **Step 1: Mapping of Regular Entity Types.**

In this step, all regular (strong) entity types should be mapped into relations. Each relation consists of all simple attributes (composite attributes are translated into a set of simple attributes) of the entity type. One of the key attributes in the entity type must be underlined considered as a primary key while others remain as candidate keys. The regular (strong) entities in this database design are DEPARTMENT, STUDENT, EMPLOYEE, BRANCH, TEACHER, PARENT, ACTIVITY, HEALTH RECORD, ACCOUNT, DEPENDENT, SUBJECT, and FACILITY.

#### **1) DEPARTMENT**

<u>Dnumber</u>	Dname	employeeNb	fax	phone

This relation consists of four attributes, “employeeNb”, “fax”, “phone”, “Dname”, and “Dnumber”. The attribute “Dnumber” is chosen to be the primary key of this relation. The “location” attribute in the entity type is a multivalued attribute which will not be present in this relation (will be discussed in a later step).

#### **2) STUDENT**

<u>SID</u>	firstName	middleName	lastName	gender	DOB	address	class_grade
rank	language						

This relation consists of ten attributes; “*gender*”, “*SID*”, “*language*”, “*rank*”, “*address*”, “*class\_grade*”, “*DOB*”, and (“*firstName*”, “*middleName*”, and “*lastName*”) as single attributes of the composite attribute “*name*” in the entity type. “*SID*” is chosen to be the primary key of this relation. The “*warning*” attribute will be discussed at a later step since it is a multivalued attribute in the entity type.

### 3) TEACHER

<u>TID</u>	firstName	middleName	lastName	gender	DOB	address	class_grade
email	salary	ss_ident					

This relation consists of ten attributes; “*gender*”, “*TID*”, “*email*”, “*address*”, “*salary*”, “*ss\_id*”, “*DOB*”, and (“*firstName*”, “*middleName*”, and “*lastName*”) as single attributes of the composite attribute “*name*” in the entity type. The “*TID*” is chosen to be the primary key for this relation. Five attributes from the entity type (“*language*”, “*comment*”, “*phone*”, “*degree*”, and “*class\_grade*”) are multivalued attributes and will be discussed later.

### 4) PARENT

<u>phone</u>	pname	DOB	gender	profession	address

This relation consists of six attributes; “*pname*”, “*profession*”, “*gender*”, “*DOB*”, “*address*”, and “*phone*”. The parent’s “*phone*” is considered as a primary key for this relation.

### 5) BRANCH

<u>branchNb</u>	fax	city	street	zipcode
-----------------	-----	------	--------	---------

This relation consists of five attributes; “*fax*”, “*branchNb*”, and (“*city*”, “*street*”, and “*zip code*”) as single attributes of the composite attribute “*location*” in the entity type BRANCH. The “*branchNb*” is chosen to be the primary key for this relation. The “*phone*” attribute is a multivalued attribute in the BRANCH entity type, so it will be discussed later.

## 6) ACTIVITY

<u>activityNumber</u>	type	occasion	expense
-----------------------	------	----------	---------

This relation has four attributes; “*activityNumber*”, “*type*”, “*occasion*” and “*expense*”. The attribute “*activityNumber*” is considered as the primary key of this relation. The “*guests*” attribute is a multivalued attribute in the ACTIVITY entity type, so it will be discussed later.

## 7) HEALTH RECORD

<u>recordID</u>	weight	height	gender	age	blood_type
-----------------	--------	--------	--------	-----	------------

This relation has six attributes; “*weight*”, “*height*”, “*gender*”, “*blood\_type*”, “*recordID*”, and “*age*”. The “*recordID*” is chosen to be the primary key of this relation. The attributes “*allergy*” and “*disease*” in the HEALTH RECORD entity type are multivalued attributes. Thus, they will be discussed later.

## 8) ACCOUNT

<u>account_nb</u>	remaining_fees	deposit	startDate	endDate
-------------------	----------------	---------	-----------	---------

This relation consists of five attributes; “*deposit*”, “*remaining fees*”, “*account\_nb*”, and (“*startDate*” and “*endDate*”) as single attributes of the composite attribute “*date*” in the entity type ACCOUNT. The attribute account “*number*” is chosen to be the primary key of this relation. The “*payment\_method*” attribute in the ACCOUNT entity type is a multivalued attribute. Therefore, it will be discussed later.

## 9) EMPLOYEE

<u>EID</u>	Ename	profession	email	gender	DOB	address	salary
ss_ident							

This relation has nine attributes; “*profession*”, “*email*”, “*EID*”, “*Ename*”, “*ss\_ident*”, “*address*”, “*DOB*”, “*salary*”, and “*gender*”. The “*EID*” attribute is considered the primary key for this relation. The “*phone*” and “*degree*” attributes in the entity type EMPLOYEE are multivalued attributes, so they will be discussed later.

## 10) DEPENDENT

<u>dependentID</u>	name	gender	DOB	relationship

This relation has five attributes; “*dependentID*”, “*name*”, “*gender*”, “*DOB*”, and “*relationship*”. The primary key of this relation is considered to be the “*dependentID*”.

## 11) SUBJECT

<u>class_grade</u>	<u>subject_name</u>	type	hourNb

This relation consists of four attributes; “*type*”, “*hourNb*”, and (“*class\_grade*” and “*subject\_name*”) as single attributes of the composite attribute “*name*” in the entity type SUBJECT. The combination of the two attributes “*class\_grade*” and “*subject\_name*” form the

primary key of this relation. The “*language*” attribute is a multivalued attribute in the SUBJECT entity type, so it will be discussed later.

## 12) FACILITY

<u>facilityID</u>	capacity	location	status	type
-------------------	----------	----------	--------	------

This relation consists of five attributes; “*facilityID*”, “*capacity*”, “*location*”, “*status*” and “*type*”. The “*facilityID*” attribute is chosen to be the primary key of this relation.

## Step 2: Mapping of Weak Entity Types.

In this step, weak entity types are mapped into relations. Each relation consists of all simple attributes (composite attributes are translated into a set of simple attributes) of the entity type. The primary key attribute(s) of the owner entity type(s) will be included in the relation as foreign key(s). The foreign key(s) along with the partial key(s) of the weak entity type form the primary key of the relation. The weak entity type in this database design is GRADE.

### GRADE

<u>studID</u>	<u>subj_name</u>	<u>class_grade</u>	<u>type</u>	<u>first_term</u>	<u>mid_term</u>
<u>final_term</u>	<u>total</u>	<u>letter_grade</u>			

The GRADE entity type has two owner entity types STUDENT and SUBJECT. Hence, this relation has a primary key formed from the combination of the partial key attribute “*type*” in the GRADE entity type along with the primary keys “*SID*” and (“*subject\_name*” and “*class\_grade*”) as foreign key attributes of the STUDENT and SUBJECT entity types respectively. In addition, this relation consists of “*first\_term*”, “*mid\_term*”, “*final\_term*”, “*total*”, and “*letter\_grade*” as simple attributes.

### **Step 3: Mapping of Binary 1:1 Relationship Types.**

In this step, we are concerned with mapping 1:1 relationship types. There are different approaches, yet the Foreign Key Approach will be used. This approach states that the primary key(s) of the relation representing the entity type with partial participation will serve as foreign key(s) for the relation representing the other participating entity type.

#### HAS\_AN:

##### **ACCOUNT**

<u>account_nb</u>	Remaining_fees	deposit	startDate	endDate	studID
-------------------	----------------	---------	-----------	---------	--------

HAS\_AN relationship type between STUDENT and ACCOUNT is 1:1 with total participation for both entity types. Thus, the primary key “SID” of the STUDENT relation is included in the ACCOUNT relation as a foreign key attribute.

#### IS\_AN:

##### **TEACHER**

TID	firstName	middleName	lastName	gender	DOB	address	class_grade
email	salary	ss_id	employeeID				

IS\_AN relationship type between TEACHER and EMPLOYEE is 1:1. TEACHER has a total participation in this relationship. Hence, the primary key “EID” of the EMPLOYEE relation is chosen to be a foreign key attribute “*employeeID*” in the TEACHER relation.

#### BELONGS\_TO:

##### **HEALTH RECORD**

<u>recordID</u>	weight	height	gender	age	blood_type	studentID
-----------------	--------	--------	--------	-----	------------	-----------

BELONGS\_TO relationship type between STUDENT and HEALTH RECORD is 1:1 with total participation for both entity types. Thus, the primary key “*SID*” of the STUDENT relation is included in the HEALTH RECORD relation as a foreign key attribute.

### TAKES\_PLACE\_IN:

#### ACTIVITY

<u>activityNumber</u>	type	occasion	expenses	guests	facilityID
-----------------------	------	----------	----------	--------	------------

Takes\_PLACE\_IN relationship type between ACTIVITY and FACILITY is 1:1. ACTIVITY entity type has a total participation in this relationship. Thus, the primary key “*facilityID*” of the FACILITY relation is included in the ACTIVITY relation as a foreign key attribute.

### MANAGES:

#### DEPARTMENT

<u>Dnumber</u>	Dname	fax	phone	employeeNb	managerID	start_date
----------------	-------	-----	-------	------------	-----------	------------

MANAGES relationship type between DEPARTMENT and EMPLOYEE is 1:1. DEPARTMENT entity type has a total participation in this relationship. Hence, the primary key “*EID*” of the EMPLOYEE relation is chosen to be a foreign key attribute in the DEPARTMENT relation.

### Step 4: Mapping of Binary 1: N Relationship Types.

In this step, binary 1: N relationship types will be mapped. The primary key of the relation in which the entity type has a cardinality ratio “1” in the relationship will be included in the relation representing the other participating entity type which has cardinality ratio “N” as a foreign key.

### HAS\_A3:

#### DEPENDENT

dependentID	name	gender	DOB	relationship	employeeID

HAS\_A3 relationship type between DEPENDENT and EMPLOYEE entity types is 1: N relationship type. DEPENDENT is at the N-side of the relationship type, so the primary key “EID” of the EMPLOYEE relation is chosen to be a foreign key attribute in the DEPENDENT relation.

### SUPERVISES1:

#### EMPLOYEE

EID	Ename	profession	email	gender	DOB	address	salary
ss_id	supervisorID						

SUPERVISES1 relationship type between EMPLOYEE entity type itself is 1: N. Thus, the primary key of the EMPLOYEE relation “EID” is included also as a foreign key attribute “supervisorID” in the same relation.

### SUPERVISES2:

#### STUDENT

SID	firstName	middleName	lastName	gender	DOB	address	class_grade
rank	language	supervisorID					

SUPERVISES2 relationship type between EMPLOYEE and STUDENT entity types is 1: N. The STUDENT entity type is at the N-side of the relationship type. Therefore, the primary key of the EMPLOYEE relation “*EID*” serves as a foreign key attribute in the STUDENT relation.

### GOES\_TO:

#### STUDENT

<u>SID</u>	firstName	middleName	lastName	gender	DOB	address	class_grade
rank	language	supervisorID	branch_num				

GOES\_TO relationship type between BRANCH and STUDENT entity types is 1: N. The STUDENT entity type is at the N-side of the relationship type, so the primary key of the BRANCH relation is included in the STUDENT relation as a foreign key.

### Step 5: Mapping of Binary M: N Relationship Types.

In this step, binary M: N relationship types will be mapped. Create a new relation for the entity types participating in such relationship type called relation relationship. This relation must include the primary key(s) of the relations that represent the participating entity types as foreign key attribute(s) as well as any additional attributes required.

### HAS\_A1:

<u>studID</u>	<u>subj_name</u>	<u>class_grade</u>	<u>type</u>

HAS\_A1 relationship type between STUDENT and GRADE entity types is M: N. Therefore, a new relation “HAS\_A1” is created to represent this relationship type. HAS\_A1 relation includes the primary

keys “SID” and (“SID”, “*subject\_name*”, “*class\_grade*” and “*type*”) of the relations STUDENT and GRADE respectively as foreign key attributes, which combined make up the primary key of this relation.

#### HAS\_A2:

<u>studID</u>	<u>parent_phone</u>

HAS\_A2 relationship type between PARENT and STUDENT entity types is M: N. Thus, a new relation is created to represent this relationship type. HAS\_A2 relation includes the primary key “SID” of STUDENT relation and “*phone*” of PARENT relation as foreign key attributes. The primary key of this relation is the combination of the two foreign key attributes.

#### IS\_OF\_A:

<u>studID</u>	<u>subj_name</u>	<u>class_grade</u>	<u>type</u>

IS\_OF\_A relationship type between SUBJECT and GRADE entity types is M: N. Therefore, a new relation IS\_OF\_A is created to represent this relationship type. This relation includes the primary keys (“SID”, “*subject\_name*”, “*class\_grade*” and “*type*”) and (“*class\_grade*” and “*subject\_name*”) of the relations GRADE and SUBJECT respectively as foreign key attributes, which combined make up the primary key of this relation.

#### PARTICIPATES\_IN:

<u>studID</u>	<u>activity_nb</u>	<u>participants_nb</u>

PARTICIPATES\_IN relationship type between ACTIVITY and STUDENT is N:M relationship type. Thus, a new PARTICIPATES\_IN relation is created. This relation consists of the primary key(s) of the ACTIVITY and STUDENT relations “*activityNumber*” and “SID” as foreign key attributes which combined make up the primary key for this relation as well as the additional attribute “*participants\_nb*”.

### ORGANIZES:

<u>employeeID</u>	<u>activity_nb</u>
-------------------	--------------------

ORGANIZES relationship type between EMPLOYEE and ACTIVITY entity types is M: N. So, a new relation ORGANIZES is created to represent this relationship type. This relation includes the primary keys “*EID*” and “*activityNumber*” of the relations EMPLOYEE and ACTIVITY as foreign key attributes which combined make up the primary key for this relation. The relationship type ORGANIZES has a multivalued attribute “*organizers\_name*” which will be discussed later.

### IS TAUGHT:

<u>class_grade</u>	<u>subject_name</u>	<u>facilityID</u>
--------------------	---------------------	-------------------

IS\_TAUGHT relationship type between SUBJECT and FACILITY is N: M. Hence, a new relation called IS\_TAUGHT is created. This relation consists of the primary key(s) of the SUBJECT relation (“*class\_grade*” and “*subject\_name*”) and the FACILITY relation (“*facilityID*”) as foreign key attributes which combined make up the primary key for this relation.

### TAKES:

<u>class_grade</u>	<u>subject_name</u>	<u>studID</u>
--------------------	---------------------	---------------

TAKES relationship type between SUBJECT and STUDENT is N: M. Thus, a new TAKES relation is created. This relation consists of the primary key(s) of the SUBJECT and STUDENT relations “*class\_grade*”, “*subject\_name*”, and “*SID*” as foreign key attributes which combined make up the primary key for this relation.

### TEACHES1:

<u>teacherID</u>	<u>studID</u>
------------------	---------------

TEACHES1 relationship type between TEACHER and STUDENT is N:M. Thus, a new TEACHER1 is created. This relation has the primary key(s) of the TEACHER and STUDENT relations “TID” and “SID” as foreign key attributes which combined make up the primary key for this relation.

### TEACHES2:

<u>teacherID</u>	<u>class_grade</u>	<u>subject_name</u>	hour_nb
------------------	--------------------	---------------------	---------

TEACHES2 relationship type is N:M relationship type between SUBJECT and TEACHER. Hence, a new relation TEACHES2 is created. This relation has as attributes “teacherID”, “class\_grade”, and “subject\_name” as foreign key attributes of the primary keys “TID”, “class\_grade”, and “subject\_name” of the TEACHER and SUBJECT relations respectively. In addition, it has the attribute “hour\_nb”.

### WORKS\_FOR:

<u>branch_nb</u>	<u>employeeID</u>	start_date
------------------	-------------------	------------

WORKS\_FOR relationship type between EMPLOYEE and BRANCH is N:M. Therefore, a new relation called WORKS\_FOR is created. This relation consists of the primary keys “EID” and “branchNb” of the EMPLOYEE and BRANCH relations respectively serving as foreign key attributes and making up the primary key for this relation. In addition, the attribute “start\_date” is added to this relation.

## **Step 6: Mapping of Multivalued attributes.**

In this step, multivalued attributes of different entity types will be mapped. A new relation will be created consisting of the attribute corresponding to this multivalued attribute, in addition to the primary key of the relation representing the entity type to which this attribute belongs as a foreign key attribute. The combination of both attributes forms the primary key for this relation.

### ACTIVITY\_GUESTS

<u>activityNb</u>	<u>guest_name</u>
-------------------	-------------------

The “*guests*” attribute in the ACTIVITY entity type is a multivalued attribute. Thus, a new relation “ACTIVITY\_GUESTS” is created. This relation includes the primary key “*activityNumber*” of ACTIVITY relation as a foreign key attribute and “*guest\_name*” attribute which combined create the primary key of ACTIVITY\_GUESTS.

### DEP\_LOCATION:

<u>Department_nb</u>	<u>location</u>
----------------------	-----------------

The “*location*” attribute in the DEPARTMENT entity type is a multivalued attribute. Thus, a new relation “DEP\_LOCATION” is created. This relation includes the primary key “*Dnumber*” of DEPARTMENT relation as a foreign key attribute and “*location*” attribute which combined create the primary key of DEP\_LOCATION.

### STUD\_WARNING:

<u>studID</u>	<u>warning</u>
---------------	----------------

The “*warning*” attribute in the STUDENT entity type is multivalued attribute. Therefore, a new relation called STUD\_WARNING is created. This relation consists of the “*warning*” attribute corresponding to

the multivalued attribute as well as the foreign key attribute “*studID*” of the primary key “*SID*” in the STUDENT relation. Both attributes form the primary key of this relation.

### TEACHER\_LANGUAGE

<u>teacherID</u>	<u>language</u>
------------------	-----------------

The “*language*” attribute in the TEACHER entity type is a multivalued attribute. Thus, a new relation “TEACHER\_LANGUAGE” is created. This relation includes the primary key “*TID*” of TEACHER relation as a foreign key attribute and “*language*” attribute which combined create the primary key of TEACHER\_LANGUAGE.

### TEACHER\_COMMENT:

<u>teacherID</u>	<u>comment</u>
------------------	----------------

The “*comment*” attribute in the TEACHER entity type is a multivalued attribute. Thus, a new relation “TEACHER\_COMMENT” is created. This relation includes the primary key “*TID*” of TEACHER relation as a foreign key attribute and “*comment*” attribute which combined create the primary key of TEACHER\_COMMENT.

### TEACHER(CG):

<u>teacherID</u>	<u>class_grade</u>
------------------	--------------------

The “*class\_grade*” attribute in the TEACHER entity type is a multivalued attribute. Thus, a new relation “TEACHER(CG)” is created. This relation includes the primary key “*TID*” of TEACHER relation as a foreign key attribute and “*class\_grade*” attribute which combined create the primary key of TEACHER(CG).

### TEACHER\_DEGREE:

<u>teacherID</u>	<u>degree</u>
------------------	---------------

The “*degree*” attribute in the TEACHER entity type is a multivalued attribute. Thus, a new relation “TEACHER\_DEGREE” is created. This relation includes the primary key “*TID*” of TEACHER relation as a foreign key attribute and “*degree*” attribute which combined create the primary key of TEACHER\_DEGREE.

### TEACHER\_PHONE:

<u>teacherID</u>	<u>phone</u>
------------------	--------------

The “*phone*” attribute in the TEACHER entity type is a multivalued attribute. Thus, a new relation “TEACHER\_PHONE” is created. This relation includes the primary key “*TID*” of TEACHER relation as a foreign key attribute and “*phone*” attribute which combined create the primary key of TEACHER\_PHONE.

### BRANCH\_PHONE:

<u>branch_nb</u>	<u>phone</u>
------------------	--------------

The “*phone*” attribute in the BRANCH entity type is a multivalued attribute. Thus, a new relation “BRANCH\_PHONE” is created. This relation includes the primary key “*branchNb*” of BRANCH relation as a foreign key attribute and “*phone*” attribute which combined creates the primary key of BRANCH\_PHONE.

### HEALTH\_DISEASE:

<u>recordID</u>	<u>disease</u>
-----------------	----------------

The attribute “*disease*” is a multivalued attribute in the HEALTH RECORD relation. Hence, HEALTH\_DISEASE relation is created. This relation has “*disease*” attribute corresponding to the multivalued attribute as well as the primary key “*recordID*” of the HEALTH RECORD relation serving as a foreign key attribute. Both attributes form the primary key of this relation.

### HEALTH\_ALLERGY:

<u>recordID</u>	<u>allergy</u>
-----------------	----------------

The attribute “*allergy*” is a multivalued attribute in the HEALTH RECORD relation. Hence, HEALTH\_ALLERGY relation is created. This relation has “*allergy*” attribute corresponding to the multivalued attribute as well as the primary key “*recordID*” of the HEALTH RECORD relation serving as a foreign key attribute. Both attributes form the primary key of this relation.

### PAYMENT:

<u>account_nb</u>	<u>payment_menthod</u>
-------------------	------------------------

The “*payment\_method*” attribute in the ACCOUNT entity type is a multivalued attribute. Thus, a new relation “PAYMENT” is created. This relation includes the primary key “*account\_nb*” of ACCOUNT relation as a foreign key attribute and “*payment\_menthod*” attribute which combined create the primary key of PAYMENT.

### EMPLOYEE\_DEGREE:

<u>employeeID</u>	<u>degree</u>
-------------------	---------------

The “*degree*” attribute in the EMPLOYEE entity type is a multivalued attribute. Thus, a new relation “EMPLOYEE\_DEGREE” is created. This relation includes the primary key “*EID*” of TEACHER relation as a foreign key attribute and “*degree*” attribute which combined create the primary key of EMPLOYEE\_DEGREE.

### EMPLOYEE\_PHONE:

<u>employeeID</u>	<u>phone</u>
-------------------	--------------

The “*phone*” attribute in the EMPLOYEE entity type is a multivalued attribute. Thus, a new relation “EMPLOYEE\_PHONE” is created. This relation includes the primary key “*EID*” of EMPLOYEE relation as a foreign key attribute and “*phone*” attribute which combined create the primary key of EMPLOYEE\_PHONE.

### SUBJECT\_LANGUAGE

<u>subject_name</u>	<u>class_grade</u>	<u>language</u>
---------------------	--------------------	-----------------

The “*language*” attribute in the SUBJECT entity type is a multivalued attribute. Thus, a new relation “SUBJECT\_LANGUAGE” is created. This relation includes the primary keys “*class\_grade*” and “*subject\_name*” of SUBJECT relation as a foreign key attribute and “*language*” attribute which combined create the primary key of SUBJECT\_LANGUAGE.

### ORGANIZER\_NAME:

<u>employeeID</u>	<u>activity_nb</u>	<u>organizer_name</u>
-------------------	--------------------	-----------------------

The attribute “*organizer\_name*” is a multivalued attribute in the ORGANIZES relationship type. Hence, a new relation called ORGANIZER\_NAME is created. This relation has “*organizer\_name*” as an attribute corresponding to the multivalued attribute as well as two foreign key attributes “*employeeID*” and “*activity\_nb*” of the primary keys of the relation ORGANIZES. Both attributes form the primary key of this relation.

## **Step 7: Mapping of N-ary Relationship Types.**

In this step, n-ary relationship types are mapped. A new relation is created consisting of the primary keys of all relations representing the n participating entity types as foreign key attributes making up the primary key of this relation. In addition, this relation includes any attributes of the n-ary relationship type. In our database design there is no n-ary relationship type, so this step will not be applied.

### **FINAL DISPLAY:**

#### **DEPARTMENT:**

<u>Dnumber</u>	Dname	fax	phone	employeeNb	managerID	start_date
----------------	-------	-----	-------	------------	-----------	------------

#### **STUDENT:**

<u>SID</u>	firstName	middleName	lastName	gender	DOB	address	class_grade
rank	language	supervisorID	branch_num				

#### **TEACHER:**

<u>TID</u>	firstName	middleName	lastName	gender	DOB	address	class_grade
email	salary	ss_id	employeeID				

#### **PARENT:**

<u>phone</u>	pname	DOB	gender	profession	address
--------------	-------	-----	--------	------------	---------

**BRANCH:**

<u>branchNb</u>	fax	city	street	zipcode
-----------------	-----	------	--------	---------

**ACTIVITY:**

<u>activityNumber</u>	type	occasion	expenses	guests	facilityID
-----------------------	------	----------	----------	--------	------------

**HEALTH RECORD:**

<u>recordID</u>	weight	height	gender	age	blood_type	studentID
-----------------	--------	--------	--------	-----	------------	-----------

**ACCOUNT:**

<u>account_nb</u>	Remaining_fees	deposit	startDate	endDate	studID
-------------------	----------------	---------	-----------	---------	--------

**EMPLOYEE:**

<u>EID</u>	Ename	profession	email	gender	DOB	address	salary
ss_id	supervisorID						

**DEPENDENT:**

<u>dependentID</u>	name	gender	DOB	relationship	employeeID
--------------------	------	--------	-----	--------------	------------

**GRADE:**

<u>studID</u>	<u>subj_name</u>	<u>class_grade</u>	<u>type</u>	<u>first_term</u>	<u>mid_term</u>
<u>final_term</u>	<u>total</u>	<u>letter_grade</u>			

SUBJECT:

<u>class_grade</u>	<u>subject_name</u>	<u>type</u>	<u>hourNb</u>
--------------------	---------------------	-------------	---------------

FACILITY:

<u>facilityID</u>	<u>capacity</u>	<u>location</u>	<u>status</u>	<u>type</u>
-------------------	-----------------	-----------------	---------------	-------------

HAS\_A1:

<u>studID</u>	<u>subj_name</u>	<u>class_grade</u>	<u>type</u>
---------------	------------------	--------------------	-------------

HAS\_A2:

<u>studID</u>	<u>parent_phone</u>
---------------	---------------------

IS\_OF\_A:

<u>studID</u>	<u>subj_name</u>	<u>class_grade</u>	<u>type</u>
---------------	------------------	--------------------	-------------

PARTICIPATES\_IN:

<u>studID</u>	<u>activity_nb</u>	participants_nb
---------------	--------------------	-----------------

ORGANIZES:

<u>employeeID</u>	<u>activity_nb</u>
-------------------	--------------------

IS\_TAUGHT:

<u>class_grade</u>	<u>subject_name</u>	<u>facilityID</u>
--------------------	---------------------	-------------------

TAKES:

<u>class_grade</u>	<u>subject_name</u>	<u>studID</u>
--------------------	---------------------	---------------

TEACHES1:

<u>teacherID</u>	<u>studID</u>
------------------	---------------

TEACHES2:

<u>teacherID</u>	<u>class_grade</u>	<u>subject_name</u>	hour_nb
------------------	--------------------	---------------------	---------

WORKS\_FOR:

<u>branch_nb</u>	<u>employeeID</u>	start_date
------------------	-------------------	------------

DEP\_LOCATION:

<u>Department_nb</u>	<u>location</u>

STUD\_WARNING:

<u>studID</u>	<u>warning</u>

TEACHER\_LANGUAGE

<u>teacherID</u>	<u>language</u>

TEACHER\_COMMENT:

<u>teacherID</u>	<u>comment</u>

TEACHER(CG:

<u>teacherID</u>	<u>class_grade</u>

TEACHER\_DEGREE:

<u>teacherID</u>	<u>degree</u>

TEACHER\_PHONE:

<u>teacherID</u>	<u>phone</u>
------------------	--------------

BRANCH\_PHONE:

<u>branch_nb</u>	<u>phone</u>
------------------	--------------

HEALTH\_DISEASE:

<u>recordID</u>	<u>disease</u>
-----------------	----------------

HEALTH\_ALLERGY:

<u>recordID</u>	<u>allergy</u>
-----------------	----------------

PAYMENT:

<u>account_nb</u>	<u>payment_menthod</u>
-------------------	------------------------

EMPLOYEE\_DEGREE:

<u>employeeID</u>	<u>degree</u>
-------------------	---------------

EMPLOYEE\_PHONE:

<u>employeeID</u>	<u>phone</u>
-------------------	--------------

### SUBJECT\_LANGUAGE

<u>subject_name</u>	<u>class_grade</u>	<u>language</u>
---------------------	--------------------	-----------------

### ORGANIZER\_NAME:

<u>employeeID</u>	<u>activity_nb</u>	<u>organizer_name</u>
-------------------	--------------------	-----------------------

## **Table Structure:**

After mapping the ER diagram into relational database design, it is time to start creating the actual tables for Bright Future School using Oracle Database Server. The first step is to create all the tables. The next step is to insert data into these tables. Finally, we will execute some queries and display the results.

### **1- SUBJECT:**

```
CREATE TABLE SUBJECT
(class_grade VARCHAR (10) NOT NULL,
subject_name VARCHAR (10) NOT NULL,
type VARCHAR (10),
hourNb INT,
PRIMARY KEY (class_grade, subject_name));
```

## **2- FACILITY:**

```
CREATE TABLE FACILITY  
(facilityID CHAR (4) NOT NULL,  
capacity INT,  
location VARCHAR (30),  
status CHAR (2),  
type VARCHAR (15),  
PRIMARY KEY (facilityID));
```

## **3- PARENT:**

```
CREATE TABLE PARENT  
(phone CHAR (12) NOT NULL,  
pname VARCHAR (30) NOT NULL,  
DOB DATE,  
gender CHAR NOT NULL,  
profession VARCHAR (30),  
address VARCHAR (50),  
PRIMARY KEY (phone));
```

## **4- BRANCH:**

```
CREATE TABLE BRANCH  
(branchNb INT NOT NULL,  
fax CHAR (8),  
city VARCHAR (10),  
street VARCHAR (10),  
zipcode CHAR (4),  
PRIMARY KEY (branchNb));
```

## **5- EMPLOYEE:**

```
CREATE TABLE EMPLOYEE  
(EID CHAR (5) NOT NULL,  
Ename VARCHAR (30) NOT NULL,
```

```
profession VARCHAR (25),  
email VARCHAR (30),  
gender CHAR,  
DOB DATE,  
address VARCHAR (50),  
salary INT,  
ss_id CHAR (3),  
supervisorID CHAR (5),  
PRIMARY KEY (EID),  
FOREIGN KEY (supervisorID) REFERENCES EMPLOYEE (EID));
```

## **6- DEPENDENT:**

```
CREATE TABLE DEPENDENT  
(dependentID VARCHAR (5) NOT NULL,  
name VARCHAR (30),  
gender CHAR,  
DOB DATE,  
relationship VARCHAR (10) NOT NULL,  
employeeID CHAR (5),  
PRIMARY KEY (dependentID),  
FOREIGN KEY (employeeID) REFERENCES EMPLOYEE (EID));
```

## **7- STUDENT:**

```
CREATE TABLE STUDENT  
(SID CHAR (6) NOT NULL,  
firstName VARCHAR (10) NOT NULL,  
lastName VARCHAR (10) NOT NULL,  
middleName VARCHAR (10),  
gender CHAR,  
DOB DATE,  
address VARCHAR (50),  
class_grade VARCHAR (10),  
rank INT,
```

```
language VARCHAR (8),  
supervisorID CHAR (5),  
branch_num INT,  
PRIMARY KEY (SID),  
FOREIGN KEY (supervisorID) REFERENCES EMPLOYEE (EID),  
FOREIGN KEY (branch_num) REFERENCES BRANCH (branchNb);
```

### **8- TEACHER:**

```
CREATE TABLE TEACHER  
(TID CHAR (5) NOT NULL,  
firstName VARCHAR (10) NOT NULL,  
middleName VARCHAR (10),  
lastName VARCHAR (10) NOT NULL,  
gender CHAR NOT NULL,  
DOB DATE,  
address VARCHAR (50),  
class_grade VARCHAR (10),  
email VARCHAR (30),  
salary VARCHAR (6),  
ss_ident CHAR (3),  
employeeID CHAR (5),  
PRIMARY KEY (TID),  
FOREIGN KEY (employeeID) REFERENCES EMPLOYEE (EID));
```

### **9- ACTIVITY:**

```
CREATE TABLE ACTIVITY  
(activityNumber INT NOT NULL,  
type VARCHAR (15),  
occasion VARCHAR (15),  
expense VARCHAR (5),  
facilityID CHAR (4),  
PRIMARY KEY (activityNumber),  
FOREIGN KEY (facilityID) REFERENCES FACILITY (facilityID));
```

## **10- HEALTH RECORD:**

```
CREATE TABLE HEALTH_RECORD  
(recordID CHAR (6) NOT NULL,  
weight CHAR (5),  
height CHAR (5),  
gender CHAR NOT NULL,  
age INT NOT NULL,  
blood_type CHAR (3),  
studentID CHAR (6) NOT NULL,  
PRIMARY KEY (recordID),  
FOREIGN KEY (studentID) REFERENCES STUDENT (SID));
```

## **11- ACCOUNT:**

```
CREATE TABLE ACCOUNT  
(account_nb CHAR (6) NOT NULL,  
remaining_fees INT,  
deposit INT,  
startDate DATE,  
endDate DATE,  
studID CHAR (6) NOT NULL,  
PRIMARY KEY (account_nb),  
FOREIGN KEY (studID) REFERENCES STUDENT (SID));
```

## **12- GRADE:**

```
CREATE TABLE GRADE  
(studID CHAR (6) NOT NULL,  
subj_name VARCHAR (10) NOT NULL,  
class_grade VARCHAR (10) NOT NULL,  
type VARCHAR (10) NOT NULL,  
first_term INT,  
mid_term INT,
```

```
final_term INT,  
total DECIMAL (10,2),  
letter_grade CHAR,  
PRIMARY KEY (studID, subj_name, class_grade, type),  
FOREIGN KEY (studID) REFERENCES STUDENT (SID),  
FOREIGN KEY (subj_name, class_grade) REFERENCES SUBJECT (subject_name, class_grade));
```

### **13- PAYMENT:**

```
CREATE TABLE PAYMENT  
(account_nb CHAR (6) NOT NULL,  
payment_method VARCHAR (10) NOT NULL,  
PRIMARY KEY (account_nb, payment_method),  
FOREIGN KEY (account_nb) REFERENCES ACCOUNT (account_nb));
```

### **14- DEPARTMENT:**

```
CREATE TABLE DEPARTMENT  
(Dname VARCHAR (15) NOT NULL,  
Dnumber INT NOT NULL,  
phone CHAR (12),  
fax CHAR (8),  
employeeNb INT,  
managerID CHAR (5) NOT NULL,  
start_date DATE,  
PRIMARY KEY (Dnumber),  
UNIQUE(Dname),  
FOREIGN KEY (managerID) REFERENCES EMPLOYEE (EID));
```

### **15- IS\_OF\_A:**

```
CREATE TABLE IS_OF_A  
(studID CHAR (6) NOT NULL,  
subj_name VARCHAR (10) NOT NULL,  
class_grade VARCHAR (10) NOT NULL,
```

```
type VARCHAR (10) NOT NULL,  
PRIMARY KEY (studID, subj_name, class_grade, type),  
FOREIGN KEY (studID,subj_name, class_grade, type) REFERENCES GRADE (studID, subj_name,  
class_grade, type),  
FOREIGN KEY (subj_name, class_grade) REFERENCES SUBJECT (subject_name, class_grade));
```

#### 16- HAS\_A1:

```
CREATE TABLE HAS_A1  
(studID CHAR (6) NOT NULL,  
subj_name VARCHAR (10) NOT NULL,  
class_grade VARCHAR (10) NOT NULL,  
type VARCHAR (10) NOT NULL,  
PRIMARY KEY (studID, subj_name, class_grade, type),  
FOREIGN KEY (studID,subj_name, class_grade, type) REFERENCES GRADE (studID, subj_name,  
class_grade, type));
```

#### 17- PARTICIPATES\_IN:

```
CREATE TABLE PARTICIPATES_IN  
(studID CHAR (6) NOT NULL,  
activity_nb INT NOT NULL,  
participants_nb INT NOT NULL,  
PRIMARY KEY (studID, activity_nb),  
FOREIGN KEY (studID) REFERENCES STUDENT (SID),  
FOREIGN KEY (activity_nb) REFERENCES ACTIVITY (activityNumber));
```

#### 18- ORGANIZES:

```
CREATE TABLE ORGANIZES  
(employeeID CHAR (5) NOT NULL,  
activity_nb INT NOT NULL,  
PRIMARY KEY (employeeID, activity_nb),  
FOREIGN KEY (employeeID) REFERENCES EMPLOYEE (EID),  
FOREIGN KEY (activity_nb) REFERENCES ACTIVITY (activityNumber));
```

## 19- IS\_TAUGHT:

```
CREATE TABLE IS_TAUGHT
(facilityID CHAR (4) NOT NULL,
class_grade VARCHAR (10) NOT NULL,
subject_name VARCHAR (10) NOT NULL,
PRIMARY KEY (class_grade, subject_name, facilityID),
FOREIGN KEY (subject_name, class_grade) REFERENCES SUBJECT (subject_name, class_grade),
FOREIGN KEY (facilityID) REFERENCES FACILITY (facilityID));
```

## 20- TAKES:

```
CREATE TABLE TAKES
(class_grade VARCHAR (10) NOT NULL,
subject_name VARCHAR (10) NOT NULL,
studID CHAR (6) NOT NULL,
PRIMARY KEY (class_grade, subject_name, studID),
FOREIGN KEY (subject_name, class_grade) REFERENCES SUBJECT (subject_name, class_grade),
FOREIGN KEY (studID) REFERENCES STUDENT (SID));
```

## 21- TEACHES1:

```
CREATE TABLE TEACHES1
(studID CHAR (6) NOT NULL,
teacherID CHAR (5) NOT NULL,
PRIMARY KEY (studID, teacherID),
FOREIGN KEY (studID) REFERENCES STUDENT (SID),
FOREIGN KEY (teacherID) REFERENCES TEACHER (TID));
```

## 22- TEACHES2:

```
CREATE TABLE TEACHES2
(class_grade VARCHAR (10) NOT NULL,
subject_name VARCHAR (10) NOT NULL,
teacherID CHAR (5) NOT NULL,
```

```
hour_nb INT,  
PRIMARY KEY (class_grade, subject_name, teacherID),  
FOREIGN KEY (subject_name, class_grade) REFERENCES SUBJECT (subject_name, class_grade),  
FOREIGN KEY (teacherID) REFERENCES TEACHER (TID));
```

### 23- WORKS\_FOR:

```
CREATE TABLE WORKS_FOR  
(branch_nb INT NOT NULL,  
employeeID CHAR (5) NOT NULL,  
start_date DATE,  
PRIMARY KEY (branch_nb, employeeID),  
FOREIGN KEY (employeeID) REFERENCES EMPLOYEE (EID),  
FOREIGN KEY (branch_nb) REFERENCES BRANCH (branchNb));
```

### 24- STUD\_WARNING:

```
CREATE TABLE STUD_WARNING  
(studID CHAR (6) NOT NULL,  
warning VARCHAR (20) NOT NULL,  
PRIMARY KEY (studID, warning),  
FOREIGN KEY (studID) REFERENCES STUDENT (SID));
```

### 25- TEACHER\_LANGUAGE:

```
CREATE TABLE TEACHER_LANGUAGE  
(teacherID CHAR (5) NOT NULL,  
language VARCHAR (8) NOT NULL,  
PRIMARY KEY (teacherID, language),  
FOREIGN KEY (teacherID) REFERENCES TEACHER (TID));
```

### 26- TEACHER\_COMMENT:

```
CREATE TABLE TEACHER_COMMENT  
(teacherID CHAR (5) NOT NULL,  
Teacher_comment VARCHAR (100) NOT NULL,
```

PRIMARY KEY (teacherID, Teacher\_comment),  
FOREIGN KEY (teacherID) REFERENCES TEACHER (TID));

### 27- TEACHER(CG):

```
CREATE TABLE TEACHER(CG  
(teacherID CHAR (5) NOT NULL,  
class_grade VARCHAR (10) NOT NULL,  
PRIMARY KEY (teacherID, class_grade),  
FOREIGN KEY (teacherID) REFERENCES TEACHER (TID));
```

### 28- TEACHER(DEGREE):

```
CREATE TABLE TEACHER(DEGREE  
(teacherID CHAR (5) NOT NULL,  
degree VARCHAR (50) NOT NULL,  
PRIMARY KEY (teacherID, degree),  
FOREIGN KEY (teacherID) REFERENCES TEACHER (TID));
```

### 29- TEACHER(PHONE):

```
CREATE TABLE TEACHER(PHONE  
(teacherID CHAR (5) NOT NULL,  
phone CHAR (12) NOT NULL,  
PRIMARY KEY (teacherID, phone),  
FOREIGN KEY (teacherID) REFERENCES TEACHER (TID));
```

### 30- BRANCH(PHONE):

```
CREATE TABLE BRANCH(PHONE  
(branch_nb INT NOT NULL,  
phone CHAR (12) NOT NULL,  
PRIMARY KEY (branch_nb, phone),  
FOREIGN KEY (branch_nb) REFERENCES BRANCH (branchNb));
```

### 31- HEALTH(DISEASE):

```
CREATE TABLE HEALTH_DISEASE
(recordID CHAR (6) NOT NULL,
disease VARCHAR (50) NOT NULL,
PRIMARY KEY (recordID, disease),
FOREIGN KEY (recordID) REFERENCES HEALTH_RECORD (recordID));
```

### 32- HEALTH\_ALLERGY:

```
CREATE TABLE HEALTH_ALLERGY
(recordID CHAR (6) NOT NULL,
allergy VARCHAR (50) NOT NULL,
PRIMARY KEY (recordID, allergy),
FOREIGN KEY (recordID) REFERENCES HEALTH_RECORD (recordID));
```

### 33- EMPLOYEE\_DEGREE:

```
CREATE TABLE EMPLOYEE_DEGREE
(employeeID CHAR (5) NOT NULL,
degree VARCHAR (50) NOT NULL,
PRIMARY KEY (employeeID, degree),
FOREIGN KEY (employeeID) REFERENCES EMPLOYEE (EID));
```

### 34- EMPLOYEE\_PHONE:

```
CREATE TABLE EMPLOYEE_PHONE
(employeeID CHAR (5) NOT NULL,
phone CHAR (12) NOT NULL,
PRIMARY KEY (employeeID, phone),
FOREIGN KEY (employeeID) REFERENCES EMPLOYEE (EID));
```

### 35- SUBJECT\_LANGUAGE:

```
CREATE TABLE SUBJECT_LANGUAGE
(subject_name VARCHAR (10) NOT NULL,
class_grade VARCHAR (10) NOT NULL,
language VARCHAR (8) NOT NULL,
```

PRIMARY KEY (subject\_name, class\_grade, language),  
FOREIGN KEY (subject\_name, class\_grade) REFERENCES SUBJECT (subject\_name, class\_grade));

### 36- ORGANIZER\_NAME:

```
CREATE TABLE ORGANIZER_NAME  
(employeeID CHAR (5) NOT NULL,  
activity_nb INT NOT NULL,  
organizer_name VARCHAR (30) NOT NULL,  
PRIMARY KEY (employeeID, activity_nb, organizer_name),  
FOREIGN KEY (employeeID) REFERENCES EMPLOYEE (EID),  
FOREIGN KEY (activity_nb) REFERENCES ACTIVITY (activityNumber));
```

### 37- ACTIVITY\_GUESTS:

```
CREATE TABLE ACTIVITY_GUESTS  
(activityNb INT NOT NULL,  
guest_name VARCHAR (15) NOT NULL,  
PRIMARY KEY (activityNb, guest_name),  
FOREIGN KEY (activityNb) REFERENCES ACTIVITY (activityNumber));
```

### 38- HAS\_A2:

```
CREATE TABLE HAS_A2  
(studID CHAR (6) NOT NULL,  
parent_phone CHAR (12) NOT NULL,  
PRIMARY KEY (studID, parent_phone),  
FOREIGN KEY (studID) REFERENCES STUDENT (SID),  
FOREIGN KEY (parent_phone) REFERENCES PARENT (phone));
```

### 39- DEP\_LOCATION:

```
CREATE TABLE DEP_LOCATION  
(department_nb INT NOT NULL,  
location VARCHAR (30) NOT NULL,  
PRIMARY KEY (department_nb, location),  
FOREIGN KEY (department_nb) REFERENCES DEPARTMENT (Dnumber));
```

## **Table Descriptions:**

After creating the tables on the oracle server, the command line “DESC TABLE\_NAME” is used to display the results of the following creation.

### **1- SUBJECT:**

Column Name	Data Type	Nullable	Default	Primary Key
CLASS_GRADE	VARCHAR2(10)	No	-	1
SUBJECT_NAME	VARCHAR2(10)	No	-	2
TYPE	VARCHAR2(10)	Yes	-	-
HOURNB	NUMBER	Yes	-	-
				1 - 4

### **2- FACILITY:**

Column Name	Data Type	Nullable	Default	Primary Key
FACILITYID	CHAR(4)	No	-	1
CAPACITY	NUMBER	Yes	-	-
LOCATION	VARCHAR2(30)	Yes	-	-
STATUS	CHAR(2)	Yes	-	-
TYPE	VARCHAR2(15)	Yes	-	-
				1 - 5

### **3- PARENT:**

Column Name	Data Type	Nullable	Default	Primary Key
PHONE	CHAR(12)	No	-	1
PNAME	VARCHAR2(30)	No	-	-
DOB	DATE	Yes	-	-
GENDER	CHAR(1)	No	-	-
PROFESSION	VARCHAR2(30)	Yes	-	-
ADDRESS	VARCHAR2(50)	Yes	-	-
				1 - 6

### **4- BRANCH:**

Column Name	Data Type	Nullable	Default	Primary Key
BRANCHNB	NUMBER	No	-	1
FAX	CHAR(8)	Yes	-	-
CITY	VARCHAR2(10)	Yes	-	-
STREET	VARCHAR2(10)	Yes	-	-
ZIPCODE	CHAR(4)	Yes	-	-
				1 - 5

## 5- EMPLOYEE:

Column Name	Data Type	Nullable	Default	Primary Key
EID	CHAR(5)	No	-	1
ENAME	VARCHAR2(30)	No	-	-
PROFESSION	VARCHAR2(25)	Yes	-	-
EMAIL	VARCHAR2(30)	Yes	-	-
GENDER	CHAR(1)	Yes	-	-
DOB	DATE	Yes	-	-
ADDRESS	VARCHAR2(50)	Yes	-	-
SALARY	NUMBER	Yes	-	-
SS_ID	CHAR(3)	Yes	-	-
SUPERVISORID	CHAR(5)	Yes	-	-
				1 - 10

## 6- DEPENDENT:

Column Name	Data Type	Nullable	Default	Primary Key
DEPENDENTID	VARCHAR2(5)	No	-	1
NAME	VARCHAR2(30)	Yes	-	-
GENDER	CHAR(1)	Yes	-	-
DOB	DATE	Yes	-	-
RELATIONSHIP	VARCHAR2(10)	No	-	-
EMPLOYEEID	CHAR(5)	Yes	-	-
				1 - 6

## 7- STUDENT:

Column Name	Data Type	Nullable	Default	Primary Key
SID	CHAR(6)	No	-	1
FIRSTNAME	VARCHAR2(10)	No	-	-
LASTNAME	VARCHAR2(10)	No	-	-
MIDDLENAME	VARCHAR2(10)	Yes	-	-
GENDER	CHAR(1)	Yes	-	-
DOB	DATE	Yes	-	-
ADDRESS	VARCHAR2(50)	Yes	-	-
CLASS_GRADE	VARCHAR2(10)	Yes	-	-
RANK	NUMBER	Yes	-	-
LANGUAGE	VARCHAR2(8)	Yes	-	-
SUPERVISORID	CHAR(5)	Yes	-	-
BRANCH_NUM	NUMBER	Yes	-	-
1 - 12				

## 8- TEACHER:

Column Name	Data Type	Nullable	Default	Primary Key
TID	CHAR(5)	No	-	1
FIRSTNAME	VARCHAR2(10)	No	-	-
MIDDLENAME	VARCHAR2(10)	Yes	-	-
LASTNAME	VARCHAR2(10)	No	-	-
GENDER	CHAR(1)	No	-	-
DOB	DATE	Yes	-	-
ADDRESS	VARCHAR2(50)	Yes	-	-
EMAIL	VARCHAR2(30)	Yes	-	-
SALARY	VARCHAR2(6)	Yes	-	-
SS_IDENT	CHAR(3)	Yes	-	-
EMPLOYEEID	CHAR(5)	Yes	-	-
1 - 11				

## 9- ACTIVITY:

Column Name	Data Type	Nullable	Default	Primary Key
ACTIVITYNUMBER	NUMBER	No	-	1
TYPE	VARCHAR2(15)	Yes	-	-
OCCASION	VARCHAR2(15)	Yes	-	-
EXPENSE	VARCHAR2(5)	Yes	-	-
FACILITYID	CHAR(4)	Yes	-	-
1 - 5				

## 10- HEALTH\_RECORD:

Column Name	Data Type	Nullable	Default	Primary Key
RECORDID	CHAR(6)	No	-	1
WEIGHT	CHAR(5)	Yes	-	-
HEIGHT	CHAR(5)	Yes	-	-
GENDER	CHAR(1)	No	-	-
AGE	NUMBER	No	-	-
BLOOD_TYPE	CHAR(3)	Yes	-	-
STUDENTID	CHAR(6)	No	-	-
1 - 7				

## 11- ACCOUNT:

Column Name	Data Type	Nullable	Default	Primary Key
ACCOUNT_NB	CHAR(6)	No	-	1
REMAINING_FEES	NUMBER	Yes	-	-
DEPOSIT	NUMBER	Yes	-	-
STARTDATE	DATE	Yes	-	-
ENDDATE	DATE	Yes	-	-
STUDID	CHAR(6)	No	-	-
1 - 6				

## 12- GRADE:

Column Name	Data Type	Nullable	Default	Primary Key
STUDID	CHAR(6)	No	-	1
SUBJ_NAME	VARCHAR2(10)	No	-	2
CLASS_GRADE	VARCHAR2(10)	No	-	3
TYPE	VARCHAR2(10)	No	-	4
FIRST_TERM	NUMBER	Yes	-	-
MID_TERM	NUMBER	Yes	-	-
FINAL_TERM	NUMBER	Yes	-	-
TOTAL	NUMBER(10,2)	Yes	-	-
LETTER_GRADE	CHAR(1)	Yes	-	-
1 - 9				

### 13- PAYMENT:

Column Name	Data Type	Nullable	Default	Primary Key
ACCOUNT_NB	CHAR(6)	No	-	1
PAYMENT_METHOD	VARCHAR2(10)	No	-	2
1 - 2				

### 14- DEPARTMENT:

Column Name	Data Type	Nullable	Default	Primary Key
DNUMBER	NUMBER	No	-	1
DNAME	VARCHAR2(15)	No	-	-
PHONE	CHAR(12)	Yes	-	-
FAX	CHAR(8)	Yes	-	-
EMPLOYEEENB	NUMBER	Yes	-	-
MANAGERID	CHAR(5)	No	-	-
START_DATE	DATE	Yes	-	-
1 - 7				

### 15- HAS\_A1:

Column Name	Data Type	Nullable	Default	Primary Key
STUDID	CHAR(6)	No	-	1
SUBJ_NAME	VARCHAR2(10)	No	-	2
CLASS_GRADE	VARCHAR2(10)	No	-	3
TYPE	VARCHAR2(10)	No	-	4
1 - 4				

### 16- IS\_OF\_A:

Column Name	Data Type	Nullable	Default	Primary Key
STUDID	CHAR(6)	No	-	1
SUBJ_NAME	VARCHAR2(10)	No	-	2
CLASS_GRADE	VARCHAR2(10)	No	-	3
TYPE	VARCHAR2(10)	No	-	4
1 - 4				

## 17- PARTICIPATES\_IN:

Column Name	Data Type	Nullable	Default	Primary Key
STUDID	CHAR(6)	No	-	1
ACTIVITY_NB	NUMBER	No	-	2
PARTICIPANTS_NB	NUMBER	No	-	-
1 - 3				

## 18- ORGANIZES:

Column Name	Data Type	Nullable	Default	Primary Key
EMPLOYEEID	CHAR(5)	No	-	1
ACTIVITY_NB	NUMBER	No	-	2
1 - 2				

## 19- IS\_TAUGHT:

Column Name	Data Type	Nullable	Default	Primary Key
CLASS_GRADE	VARCHAR2(10)	No	-	1
SUBJECT_NAME	VARCHAR2(10)	No	-	2
FACILITYID	CHAR(4)	No	-	3
1 - 3				

## 20- TAKES:

Column Name	Data Type	Nullable	Default	Primary Key
CLASS_GRADE	VARCHAR2(10)	No	-	1
SUBJECT_NAME	VARCHAR2(10)	No	-	2
STUDID	CHAR(6)	No	-	3
				1 - 3

## 21- TEACHES1:

Column Name	Data Type	Nullable	Default	Primary Key
STUDID	CHAR(6)	No	-	1
TEACHERID	CHAR(5)	No	-	2
				1 - 2

## 22- TEACHES2:

Column Name	Data Type	Nullable	Default	Primary Key
CLASS_GRADE	VARCHAR2(10)	No	-	1
SUBJECT_NAME	VARCHAR2(10)	No	-	2
TEACHERID	CHAR(5)	No	-	3
HOUR_NB	NUMBER	Yes	-	-
				1 - 4

## 23- WORKS\_FOR:

Column Name	Data Type	Nullable	Default	Primary Key
BRANCH_NB	NUMBER	No	-	1
EMPLOYEEID	CHAR(5)	No	-	2
START_DATE	DATE	Yes	-	-
				1 - 3

## 24- STUD\_WARNING:

Column Name	Data Type	Nullable	Default	Primary Key
STUDID	CHAR(6)	No	-	1
WARNING	VARCHAR2(20)	No	-	2
1 - 2				

## 25- TEACHER\_LANGUAGE:

Column Name	Data Type	Nullable	Default	Primary Key
TEACHERID	CHAR(5)	No	-	1
LANGUAGE	VARCHAR2(8)	No	-	2
1 - 2				

## 26- TEACHER\_COMMENT:

Column Name	Data Type	Nullable	Default	Primary Key
TEACHERID	CHAR(5)	No	-	1
TEACHER_COMMENT	VARCHAR2(100)	No	-	2
1 - 2				

## 27- TEACHER(CG:

Column Name	Data Type	Nullable	Default	Primary Key
TEACHERID	CHAR(5)	No	-	1
CLASS_GRADE	VARCHAR2(10)	No	-	2
1 - 2				

## 28- TEACHER\_DEGREE:

Column Name	Data Type	Nullable	Default	Primary Key
TEACHERID	CHAR(5)	No	-	1
DEGREE	VARCHAR2(50)	No	-	2
1 - 2				

## 29- TEACHER\_PHONE:

Column Name	Data Type	Nullable	Default	Primary Key
TEACHERID	CHAR(5)	No	-	1
PHONE	CHAR(12)	No	-	2
1 - 2				

### 30- BRANCH\_PHONE:

Column Name	Data Type	Nullable	Default	Primary Key
BRANCH_NB	NUMBER	No	-	1
PHONE	CHAR(12)	No	-	2
1 - 2				

### 31- HEALTH\_DISEASE:

Column Name	Data Type	Nullable	Default	Primary Key
RECORDID	CHAR(6)	No	-	1
DISEASE	VARCHAR2(50)	No	-	2
1 - 2				

### 32- HEALTH\_ALLERGY:

Column Name	Data Type	Nullable	Default	Primary Key
RECORDID	CHAR(6)	No	-	1
ALLERGY	VARCHAR2(50)	No	-	2
1 - 2				

### 33- EMPLOYEE\_DEGREE:

Column Name	Data Type	Nullable	Default	Primary Key
EMPLOYEEID	CHAR(5)	No	-	1
DEGREE	VARCHAR2(50)	No	-	2
1 - 2				

### 34- EMPLOYEE\_PHONE:

Column Name	Data Type	Nullable	Default	Primary Key
EMPLOYEEID	CHAR(5)	No	-	1
PHONE	CHAR(12)	No	-	2
1 - 2				

### 35- SUBJECT\_LANGUAGE:

Column Name	Data Type	Nullable	Default	Primary Key
SUBJECT_NAME	VARCHAR2(10)	No	-	1
CLASS_GRADE	VARCHAR2(10)	No	-	2
LANGUAGE	VARCHAR2(8)	No	-	3
1 - 3				

### 36- ORGANIZER\_NAME:

Column Name	Data Type	Nullable	Default	Primary Key
EMPLOYEEID	CHAR(5)	No	-	1
ACTIVITY_NB	NUMBER	No	-	2
ORGANIZER_NAME	VARCHAR2(30)	No	-	3
1 - 3				

### 37- ACTIVITY\_GUESTS:

Column Name	Data Type	Nullable	Default	Primary Key
ACTIVITYNB	NUMBER	No	-	1
GUEST_NAME	VARCHAR2(15)	No	-	2
1 - 2				

### 38- HAS\_A2:

Column Name	Data Type	Nullable	Default	Primary Key
STUDID	CHAR(6)	No	-	1
PARENT_PHONE	CHAR(12)	No	-	2
1 - 2				

## 39- DEP\_LOCATION:

Column Name	Data Type	Nullable	Default	Primary Key
DEPARTMENT_NB	NUMBER	No	-	1
LOCATION	VARCHAR2(30)	No	-	2
1 - 2				

## Inserting Data:

### 1- SUBJECT:

```
begin
  INSERT INTO SUBJECT
    Values ('Grade 12', 'Math', 'Sciences', 7);
  INSERT INTO SUBJECT
    Values ('Grade 5', 'Arabic', 'Humanities',5);
  INSERT INTO SUBJECT
    Values ('Grade 8', 'Chemistry', 'Sciences', 6);
  INSERT INTO SUBJECT
    Values ('Grade 7', 'Biology', 'Sciences', 6);
  INSERT INTO SUBJECT
    Values ('Grade 12', 'English', 'Humanities',5);
  INSERT INTO SUBJECT
    Values ('Grade 9', 'French', 'Humanities',5);
  INSERT INTO SUBJECT
    Values ('Grade 9', 'Physics', 'Sciences', 7);
  INSERT INTO SUBJECT
    Values ('Grade 6', 'civics', 'Humanities', 1);
  INSERT INTO SUBJECT
    Values ('Grade 11', 'Philosophy', 'Humanities', 2);
  INSERT INTO SUBJECT
    Values ('Grade 12', 'Acting', 'Art', 1);
end
```

### 2- FACILITY:

```
begin
INSERT INTO FACILITY
Values ('1001', 30, 'Building 1 floor 2', 'R', 'classroom');
INSERT INTO FACILITY
Values ('1002', 50, 'Building 2 floor 2', 'NR', 'classroom');
INSERT INTO FACILITY
Values ('1003', 45, 'Building 1 floor 3', 'R', 'classroom');
INSERT INTO FACILITY
Values ('1004', 25, 'Building 1 floor 1', 'NR', 'classroom');
INSERT INTO FACILITY
Values ('1090', 100, 'Building 2 floor -1', 'NR', 'stage');
INSERT INTO FACILITY
Values ('1020', 4, 'Building 3 floor 2', 'R', 'Office');
INSERT INTO FACILITY
Values ('1025', 4, 'Building 3 floor 2', 'R', 'Office');
INSERT INTO FACILITY
Values ('1060', 100, 'floor 0', 'NR', 'Playground');
INSERT INTO FACILITY
Values ('1032', 25, 'Building 3 floor 1', 'R', 'Lab');
INSERT INTO FACILITY
Values ('1038', 40, 'Building 3 floor 1', 'NR', 'Lab');
end
```

### 3- PARENT:

```
begin
INSERT INTO PARENT
Values ('+96171102361','Bassam Hamze', DATE '1980-02-11','M','Architect', 'Nabatieh/ Kfaroman');
INSERT INTO PARENT
Values ('+96103821766', 'Allam Jaber', DATE '1976-12-01','M','Business man', 'Nabatieh/
Mayfadoun');
INSERT INTO PARENT
Values ('+96179175075', 'Mohammad Kdouh', DATE '1977-05-15', 'M', 'Clothes Seller','Nabatieh/ Hay
El Bayad');
```

```
INSERT INTO PARENT
Values ('+96170552446', 'Ali Cheaib', DATE '1979-09-24', 'M', 'Actor','Nabatieh/ Al Sharkiyeh');

INSERT INTO PARENT
Values ('+96107767760', 'Samer Wehbe', DATE '1975-08-13', 'M', 'Carpenter','Beirut/ Hamra');

INSERT INTO PARENT
Values ('+96103835537', 'Najwa Houmani', DATE '1981-02-19', 'F','waitor', 'Beirut/ Kraytem');

INSERT INTO PARENT
Values ('+96176659201', 'Tamara Younnes', DATE '1982-10-20', 'F','Doctor', 'Beirut/ Hadath');

INSERT INTO PARENT
Values ('+96170352364', 'Hoda Soboh', DATE '1979-12-29', 'F','Lawyer', 'Beirut/ Antelyas');

INSERT INTO PARENT
Values ('+96103644678', 'Mariam Baaklini', DATE '1978-11-23', 'F', 'Engineer', 'Keserwen/ Harajel');

INSERT INTO PARENT
Values ('+96176928459', 'Lea Dia', DATE '1981-08-21', 'F', 'nurse','Keserwen/ Mayrouba');

end
```

#### 4- BRANCH:

```
begin
INSERT INTO BRANCH
Values (1, '556-3219', 'Keserwen', 'Snoubar', '2222');

INSERT INTO BRANCH
Values (2, '874-2566', 'Beirut', 'Emile Edde', '3333');

INSERT INTO BRANCH
Values (3, '542-3261', 'Nabatieh', 'Dennawi', '4444');

end
```

#### 5- EMPLOYEE:

```
begin
INSERT INTO EMPLOYEE
Values ('10018', 'Wael Jihad Saad', 'Principal', 'waelJ1_@hotmail.com', 'M', DATE '1962-10-15', 'Beirut/
Bir Hasan', 3200, 'yes', NULL);

INSERT INTO EMPLOYEE
```

Values ('10010', 'Jana Ayad Jaber', 'Chairperson', 'jana.jaber03@hotmail.com', 'F', DATE '1990-10-28', 'Beirut/ Zaytouna bay', 4000, 'yes', NULL);

INSERT INTO EMPLOYEE

Values ('10002', 'Mohammad Ali Jaber', 'Coordinator', 'moe.jaber99@hotmail.com', 'M', DATE '1988-11-03', 'Beirut/ Roumieh', 2000, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10003', 'Mohammad Tarek Rahal', 'Academic Assistant', 'moe\_rahal@hotmail.com', 'M', DATE '1980-04-20', 'Beirut/ Kraytem', 2000, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10004', 'Nader Kassem Khadaj', 'Faculty Advisor', 'Nader\_85\_@hotmail.com', 'M', DATE '1985-03-10', 'Beirut/ Hamra', 2200, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10005', 'Maya Omar Darwich', 'Coordinator', 'Maya123\_@hotmail.com', 'F', DATE '1989-12-11', 'Keserwen/ Faraya', 2500, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10006', 'Christina Ali Kassem', 'Coordinator', 'Chris\_Kassem@hotmail.com', 'F', DATE '1983-01-13', 'Keserwen/ Ajaltoun', 2000, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10007', 'Wissam Walid Baaklini', 'Academic Assistant', 'Wissam\_92@hotmail.com', 'M', DATE '1992-05-08', 'Keserwen/ balllouneh', 2000, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10008', 'Khodor Ali Ghalem', 'Faculty Advisor', 'Khodor\_GH@hotmail.com', 'M', DATE '1975-07-14', 'Beirut/ Baabda', 2200, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10009', 'Yasmina AL-Ayache', 'Academic Assistant', 'YasminaAyache09\_@hotmail.com', 'F', DATE '1972-08-18', 'Beirut/ Aley', 2500, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10001', 'Baha Raed Chaar', 'Faculty Advisor', 'Chaar468\_@hotmail.com', 'M', DATE '1970-11-21', 'Beirut/ Aley', 2200, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10011', 'Charbel Joseph Younes', 'Security Guard', 'Charbel7676@hotmail.com', 'M', DATE '1980-08-01', 'Beirut/ Sin El-Fil', 1500, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10012', 'Rawad Ayad Safa', 'Security Guard', 'Kheir\_R@hotmail.com', 'M', DATE '1968-09-11', 'Nabatieh/ Mayfadoun', 1500, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10013', 'Mohammad Ali Shamas', 'Information technology', 'Shamas\_Moe\_@hotmail.com', 'M', DATE '1985-02-09', 'Beirut/ Hazmieh', 3000, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10014', 'Maha Mahmoud Chakaron', 'Custodian', Null, 'F', DATE '1980-08-01', 'Nabatieh/ Kfarromen', 1000, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10015', 'Mira Jamil Issa', 'cleaner', NULL, 'F', DATE '1972-05-07', 'Beirut/ Jal Al Dib', 1000, 'yes', '10014');

INSERT INTO EMPLOYEE

Values ('10016', 'Ali Sami Sharara', 'Bus driver', 'Sharara\_Ali89\_@hotmail.com', 'M', DATE '1989-12-25', 'Saida', 1200, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10017', 'Diala Jamal Abdo', 'School nurse', 'diala014@hotmail.com', 'F', DATE '1982-05-02', 'Beirut/ Manara', 3000, 'yes', '10018');

INSERT INTO EMPLOYEE

Values ('10030', 'Hussein Ali Bakri', 'Teacher', 'Bakri75\_@hotmail.com', 'M', DATE '1975-12-10', 'Beirut/ Antelyas', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10031', 'Hikmat Toufic Farhat', 'Teacher', 'Hikmat0&\_@hotmail.com', 'M', DATE '1970-06-06', 'Nabatieh/ Kfarjoz', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10032', 'Faisal Adel Al khazen', 'Teacher', 'Faisal434\_@hotmail.com', 'M', DATE '1965-07-10', 'Beirut/ Hamra', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10033', 'Samer Said Habre', 'Teacher', 'Habre\_Samer\_@hotmail.com', 'M', DATE '1968-05-16', 'Beirut/ Hadath', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10034', 'Chadi Kamil Nour', 'Teacher', 'Nour76\_@hotmail.com', 'M', DATE '1976-10-10', 'Beirut/ Kornich Al Mazraa', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10035', 'Maha Daoud Tabet', 'Teacher', 'Maha.Tabet\_@hotmail.com', 'F', DATE '1980-08-19', 'Beirut/ Kmateye', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10036', 'Nadine Fawaz Abbas', 'Teacher', 'Nadine.Abbas1\_@hotmail.com', 'F', DATE '1988-03-11', 'Beirut/ Hamra', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10037', 'Sanaa Hussein Sharafeddine', 'Teacher', 'Sanaa.H\_@hotmail.com', 'F', DATE '1985-07-03', 'Beirut/ Jnah', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10038', 'Zahraa Ali Sweidan', 'Teacher', 'Zahraa.Sweidan02\_@hotmail.com', 'F', DATE '1990-03-17', 'Beirut/ Khalde', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10039', 'Tamara Issam Al-Khishen', 'Teacher', 'Tamara.09\_@hotmail.com', 'F', DATE '1985-09-12', 'Beirut/ Aley', 4500, 'yes', '10010');

INSERT INTO EMPLOYEE

Values ('10040', 'Dana Elie Doumit', 'Supervisor', 'Dana\_@hotmail.com', 'F', DATE '1965-09-11', 'Keserwen/ Faraya', 3500, 'yes', NULL);

INSERT INTO EMPLOYEE

Values ('10050', 'Bassem Ali Salman', 'Chairperson', 'Bassem1\_@hotmail.com', 'M', DATE '1962-09-20', 'Baalbeck/ Chmestar', 4000, 'yes', NULL);

INSERT INTO EMPLOYEE

Values ('10051', 'Rayan Wissam Shehadeh', 'Chairperson', 'RShehadeh1\_@hotmail.com', 'M', DATE '1969-05-06', 'Saida/ Khartoum', 4000, 'yes', NULL);

INSERT INTO EMPLOYEE

Values ('10052', 'Majd Hamadan Fares', 'Chairperson', 'M\_75Fares\_@hotmail.com', 'M', DATE '1975-10-10', 'Sour/ Al Hosh', 4000, 'yes', NULL);

INSERT INTO EMPLOYEE

Values ('10053', 'Mohammad Hisham Ozeir', 'Chairperson', 'MoeOzeir1\_@hotmail.com', 'M', DATE '1980-12-15', 'Beirut/ Airport Road', 5000, 'yes', NULL);

INSERT INTO EMPLOYEE

Values ('10054', 'Karim Elie Akar', 'Chairperson', 'Akar451\_@hotmail.com', 'M', DATE '1985-02-09', 'Beirut/ Cheyah', 3800, 'yes', NULL);

INSERT INTO EMPLOYEE

Values ('10055', 'Nada Hasan Saab', 'Chairperson', 'Nada.Saab7\_@hotmail.com', 'F', DATE '1970-01-29', 'Beirut/ Hazmieh', 5000, 'yes', NULL);  
INSERT INTO EMPLOYEE  
Values ('10056', 'Sara Said Ajami', 'Chairperson', 'Ajami.Sara\_@hotmail.com', 'F', DATE '1987-11-13', 'Beirut/ Chouifat', 4000, 'yes', NULL);  
INSERT INTO EMPLOYEE  
Values ('10057', 'Hussein Hasan Mawla', 'Chairperson', 'Mawla12\_@hotmail.com', 'M', DATE '1973-08-02', 'Beirut/ Ghobeiry', 4000, 'yes', NULL);  
INSERT INTO EMPLOYEE  
Values ('10058', 'Mona Jaber Harajli', 'Chairperson', 'Mona\_J\_@hotmail.com', 'F', DATE '1971-09-14', 'Beirut/ Dbayeh', 4000, 'yes', NULL);  
INSERT INTO EMPLOYEE  
Values ('10059', 'Reem Abdallah Chemaly', 'Chairperson', 'Reem.Chemaly\_@hotmail.com', 'F', DATE '1988-04-03', 'Saida/ Kfarhata', 4000, 'yes', NULL);  
INSERT INTO EMPLOYEE  
Values ('10060', 'Jamil Ali Bdeir', 'Supervisor', 'Jamil\_@hotmail.com', 'M', DATE '1962-01-12', 'Beirut/ Hadath', 3500, 'yes', NULL);  
INSERT INTO EMPLOYEE  
Values ('10070', 'Jad Samer Houmani', 'Supervisor', 'Jad34\_@hotmail.com', 'M', DATE '1970-01-11', 'Nabatieh/ Al Sharkiyeh', 3500, 'yes', NULL);  
end

## 6- DEPENDENT:

begin  
INSERT INTO DEPENDENT  
Values ('20201','Mona Rahal', 'F', DATE'2012-03-16','Daughter', '10003');  
INSERT INTO DEPENDENT  
Values ('20202','Sawsan Khaddaj','F', DATE'1983-07-19','Fiancee', '10004');  
INSERT INTO DEPENDENT  
Values ('20203','Diala Mokbel' , 'F', DATE'1992-08-22','Wife', '10007');  
INSERT INTO DEPENDENT  
Values ('20204','Ahlam Ghalem' , 'F', DATE'2006-12-29','Daughter', '10018');  
INSERT INTO DEPENDENT

```
Values ('20205','Samih Ghalem' , 'M', DATE'2009-01-21','Son', '10018');  
INSERT INTO DEPENDENT  
Values ('20206','Halim Bazzi' , 'M', DATE'1971-11-01','Husband', '10009');  
INSERT INTO DEPENDENT  
Values ('20207','Hadi Bazzi' , 'M', DATE'1988-01-04','Brother', '10010');  
INSERT INTO DEPENDENT  
Values ('20208', 'Fatima Safa' , 'F', DATE'2002-11-14','Daughter', '10012');  
INSERT INTO DEPENDENT  
Values ('20209','Natalia Fares' , 'F', DATE'1980-05-03','Wife', '10052');  
INSERT INTO DEPENDENT  
Values ('20210','Ramzi Sharafeddin' , 'M', DATE'1999-04-09','Son', '10037');  
end
```

## 7- STUDENT:

```
begin  
INSERT INTO STUDENT  
Values ('200201','Tarek', 'Hamze', 'Bassam', 'M', DATE'2004-12-05','Nabatieh/ Kafarouman','Grade  
12',2,'French','10070',3);  
INSERT INTO STUDENT  
Values ('200202','Zein', 'Jaber', 'Allam', 'M', DATE'2006-12-28','Nabatieh/ Mayfadoun','Grade  
9',10,'French','10070',3);  
INSERT INTO STUDENT  
Values ('200203','Israa', 'Kdouh', 'Mohammad', 'F', DATE'2007-03-05','Nabatieh/ Hay El Bayad','Grade  
8',5,'English','10070',3);  
INSERT INTO STUDENT  
Values ('200204','Mohammad', 'Cheaib', 'Ali', 'M', DATE'2004-10-22','Nabatieh/ Al Sharkiyeh','Grade  
12',1,'French','10070',3);  
INSERT INTO STUDENT  
Values ('200205','Jana', 'Wehbe', 'Samer', 'F', DATE'2009-11-28','Beirut/ Hamra','Grade  
7',4,'French','10060',2);  
INSERT INTO STUDENT  
Values ('200206','Farah', 'Jaber', 'Ayad', 'F', DATE'2006-04-05','Beirut/ Kraytem','Grade  
9',2,'English','10060',2);
```

INSERT INTO STUDENT

Values ('200207','Hussein', 'Hayek', 'Hassan', 'M', DATE'2005-03-10','Beirut/ Hadath','Grade 11',7,'English','10060',2);

INSERT INTO STUDENT

Values ('200208','Joey', 'Youness', 'Haysam', 'M', DATE'2010-02-09','Beirut/ Antelyas','Grade 6',4,'French','10060',2);

INSERT INTO STUDENT

Values ('200209','Issa', 'Hassan', 'Hussein', 'M', DATE'2004-07-20','Keserwen/ Harajel','Grade 12',10,'English','10040',1);

INSERT INTO STUDENT

Values ('200210','Joelle', 'Hammoud', 'Bilal', 'F', DATE'2012-01-15','Keserwen/ Mayrouba','Grade 5',7,'French','10040',1);

end

## 8- TEACHER:

begin

INSERT INTO TEACHER

Values ('10501', 'Hussein', 'Ali', 'Bakri', 'M', DATE'1975-12-10', 'Beirut/ Antelyas', 'Bakri75\_@hotmail.com', '4500\$', 'yes', '10030');

INSERT INTO TEACHER

Values ('10502', 'Hikmat', 'Toufic', 'Farhat', 'M', DATE'1970-06-06', 'Nabatieh/ Kfarjoz', 'Hikmat0&\_@hotmail.com', '4500\$', 'yes', '10031');

INSERT INTO TEACHER

Values ('10503', 'Faisal', 'Adel', 'Al khazen', 'M', DATE'1965-07-10', 'Beirut/ Hamra', 'Faisal434\_@hotmail.com', '4500\$', 'yes', '10032');

INSERT INTO TEACHER

Values ('10504', 'Samer', 'Said', 'Habre', 'M', DATE'1968-05-16', 'Beirut/ Hadath', 'Habre\_Samer\_@hotmail.com', '4500\$', 'yes', '10033');

INSERT INTO TEACHER

Values ('10505', 'Chadi', 'Kamil', 'Nour', 'M', DATE'1976-10-10', 'Beirut/ Kornich Al Mazraa', 'Nour76\_@hotmail.com', '4500\$', 'yes', '10034');

INSERT INTO TEACHER

Values ('10506', 'Maha', 'Daoud', 'Tabet', 'F', DATE'1980-08-19', 'Beirut/ Kmatelye',  
'Maha.Tabet\_@hotmail.com', '4500\$', 'yes', '10035');

INSERT INTO TEACHER

Values ('10507', 'Nadine', 'Fawaz', 'Abbas', 'F', DATE'1988-03-11', 'Beirut/ Hamra',  
'Nadine.Abbas1\_@hotmail.com', '4500\$', 'yes', '10036');

INSERT INTO TEACHER

Values ('10508', 'Sanaa', 'Hussein', 'Sharfedine', 'F', DATE'1985-07-03', 'Beirut/ Jnah',  
'Sanaa.H\_@hotmail.com', '4500\$', 'yes', '10037');

INSERT INTO TEACHER

Values ('10509', 'Zahraa', 'Ali', 'Sweidan', 'F', DATE'1990-03-17', 'Beirut/ Khalde',  
'Zahraa.Sweidan02\_@hotmail.com', '4500\$', 'yes', '10038');

INSERT INTO TEACHER

Values ('10510', 'Tamara', 'Issam', 'Al-Khishen', 'F', DATE'1985-09-12', 'Beirut/ Aley',  
'Tamara.09\_@hotmail.com', '4500\$', 'yes', '10039');

end

## 9- ACTIVITY:

begin

INSERT INTO ACTIVITY

Values (1,'Songs','Independence','1000\$','1060');

INSERT INTO ACTIVITY

Values (2,'Songs','Christmas','1000\$','1090');

INSERT INTO ACTIVITY

Values (3,'Marathon','Sports day','500\$','1060');

INSERT INTO ACTIVITY

Values (4,'Songs','End of the year','1000\$','1060');

INSERT INTO ACTIVITY

Values (5,'Mentoring',NULL,'200\$','1002');

INSERT INTO ACTIVITY

Values (6,'Acting','Christmas','2000\$','1090');

INSERT INTO ACTIVITY

Values (7,'Acting','Mothers Day','2000\$','1090');

INSERT INTO ACTIVITY

```
Values (8,'Robotics',NULL,'2000$', '1060');

INSERT INTO ACTIVITY

Values (9,'Debate','MUN','1300$', '1090');

INSERT INTO ACTIVITY

Values (10,'Academic','competiton','500$', '1002');

end
```

## 10- HEALTH RECORD:

```
begin

INSERT INTO HEALTH_RECORD

Values ('200101','70kg','180cm','M', 18,'O+', '200201');

INSERT INTO HEALTH_RECORD

Values ('200102','65kg','170cm','M',16,'O-', '200202');

INSERT INTO HEALTH_RECORD

Values ('200103','50kg','130cm','F',15,'AB+', '200203') ;

INSERT INTO HEALTH_RECORD

Values ('200104','80kg','182cm','M',18,'AB-', '200204') ;

INSERT INTO HEALTH_RECORD

Values ('200105','40kg','145cm','F',13,'A+', '200205') ;

INSERT INTO HEALTH_RECORD

Values ('200106','45kg','165cm','F',16,'A-', '200206') ;

INSERT INTO HEALTH_RECORD

Values ('200107','65kg','175cm','M',17,'O+', '200207') ;

INSERT INTO HEALTH_RECORD

Values ('200108','42kg','162cm','F',12,'O-', '200208') ;

INSERT INTO HEALTH_RECORD

Values ('200109','70kg' , '170cm','M',18,'AB+', '200209') ;

INSERT INTO HEALTH_RECORD

Values ('200110','37kg' , '130cm','F',10,'AB-', '200210') ;

end
```

## 11- ACCOUNT:

```
begin
```

```
INSERT INTO ACCOUNT
Values ('101001',700,200, DATE'2015-01-04', DATE'2021-06-12', '200201');

INSERT INTO ACCOUNT
Values ('101002',1000,300, DATE'2010-11-14', DATE'2022-03-11', '200202');

INSERT INTO ACCOUNT
Values ('101003',2000,200, DATE'2011-04-03', DATE'2023-08-19', '200203');

INSERT INTO ACCOUNT
Values ('101004',500,300, DATE'2008-11-06', DATE'2020-12-12', '200204');

INSERT INTO ACCOUNT
Values ('101005',2300,200, DATE'2012-10-04', DATE'2022-10-22', '200205');

INSERT INTO ACCOUNT
Values ('101006',7000,300, DATE'2010-04-25', DATE'2023-12-23', '200206');

INSERT INTO ACCOUNT
Values ('101007',100,300, DATE'2009-02-03', DATE'2020-07-12', '200207');

INSERT INTO ACCOUNT
Values ('101008',900,300, DATE'2015-12-29', DATE'2024-05-17', '200208');

INSERT INTO ACCOUNT
Values ('101009',3000,200, DATE'2009-10-13', DATE'2021-03-14', '200209');

INSERT INTO ACCOUNT
Values ('101010',750,200, DATE'2016-03-09', DATE'2026-05-12', '200210');

end
```

## 12- GRADE:

```
begin

INSERT INTO GRADE
Values ('200201', 'Math', 'Grade 12', 'first term', 95, 90, 100,95.00, 'A');

INSERT INTO GRADE
Values ('200202', 'Physics', 'Grade 9', 'final term', 75,80,55,70.00, 'C');

INSERT INTO GRADE
Values ('200203', 'Chemistry', 'Grade 8', 'first term', 86,60,75, 80.33, 'B');

INSERT INTO GRADE
Values ('200204', 'Acting', 'Grade 12', 'midterm', 97, 95, 100, 98.00, 'A');

INSERT INTO GRADE
```

```
Values ('200205', 'Biology', 'Grade 7', 'final term', 90, 90, 80,86.66, 'B');

INSERT INTO GRADE

Values ('200206', 'French', 'Grade 9', 'midterm', 95,98,100,97.66, 'A');

INSERT INTO GRADE

Values ('200207', 'Philosophy', 'Grade 11', 'first term', 80,78,82,80, 'B');

INSERT INTO GRADE

Values ('200208', 'civics', 'Grade 6', 'final term', 85,87,82,84.66, 'B');

INSERT INTO GRADE

Values ('200209', 'English', 'Grade 12', 'midterm', 65,70,71,68.66, 'D');

INSERT INTO GRADE

Values ('200210', 'Arabic', 'Grade 5', 'first term', 65,74,73,70.66, 'C');

end
```

### 13- PAYMENT:

```
begin

INSERT INTO PAYMENT

Values ('101001','Cash/ LB');

INSERT INTO PAYMENT

Values ('101002','Cash/ USD');

INSERT INTO PAYMENT

Values ('101003','Check/ LB');

INSERT INTO PAYMENT

Values ('101004','Check/ USD');

INSERT INTO PAYMENT

Values ('101005','Cash/ LB');

INSERT INTO PAYMENT

Values ('101006','Cash/ USD');

INSERT INTO PAYMENT

Values ('101007','Check/ LB');

INSERT INTO PAYMENT

Values ('101008','Check/ USD');

INSERT INTO PAYMENT

Values ('101009','Cash/LB');
```

```
INSERT INTO PAYMENT
```

```
Values ('101010','Cash/ USD');
```

```
end
```

#### 14- DEPARTMENT:

```
begin
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Research',1, '+96170987432', '009-8712', 20, '10050', DATE'2013-02-11');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Biology', 2,'+96170930431', '008-2516', 10, '10051', DATE'2005-09-18');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Athletics',3, '+96178030733', '007-2130', 15, '10052', DATE'2012-12-08');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Mathematics', 4, '+96181100431', '006-1274', 10, '10053', DATE'2007-02-03');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Theatre/Dance',5, '+96181654044', '005-1374', 12, '10054', DATE'2013-01-25');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Arts/Humanities', 6, '+96170787317', '004-0154', 10, '10055', DATE'2007-11-20');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Chemistry', 7,'+96171870398', '003-9812', 20, '10056', DATE'2009-10-10');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Social Sciences', 8,'+96170921401', '002-5423', 10, '10057', DATE'2006-06-08');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Languages',9, '+96171876120', '002-4171', 20, '10058', DATE'2010-12-09');
```

```
INSERT INTO DEPARTMENT
```

```
Values ('Arabic',10, '+96181876410', '001-1112', 10, '10059', DATE'2011-03-18');
```

```
end
```

#### 15- HAS\_A1:

```
begin
```

```
INSERT INTO HAS_A1
```

```
Values ('200203', 'Chemistry', 'Grade 8', 'first term');
```

```
INSERT INTO HAS_A1
```

```
Values ('200204', 'Acting', 'Grade 12', 'midterm');
INSERT INTO HAS_A1
Values ('200205', 'Biology', 'Grade 7', 'final term');
INSERT INTO HAS_A1
Values ('200206', 'French', 'Grade 9', 'midterm');
INSERT INTO HAS_A1
Values ('200207', 'Philosophy', 'Grade 11', 'first term');
INSERT INTO HAS_A1
Values ('200208', 'civics', 'Grade 6', 'final term');
INSERT INTO HAS_A1
Values ('200209', 'English', 'Grade 12', 'midterm');
INSERT INTO HAS_A1
Values ('200210', 'Arabic', 'Grade 5', 'first term');
INSERT INTO HAS_A1
Values ('200202', 'Physics', 'Grade 9', 'final term');
INSERT INTO HAS_A1
Values ('200201', 'Math', 'Grade 12', 'first term');
end
```

## 16- IS\_OF\_A:

```
begin
INSERT INTO IS_OF_A
Values ('200201', 'Math', 'Grade 12', 'first term');
INSERT INTO IS_OF_A
Values ('200202', 'Physics', 'Grade 9', 'final term');
INSERT INTO IS_OF_A
Values ('200203', 'Chemistry', 'Grade 8', 'first term');
INSERT INTO IS_OF_A
Values ('200204', 'Acting', 'Grade 12', 'midterm');
INSERT INTO IS_OF_A
Values ('200205', 'Biology', 'Grade 7', 'final term');
INSERT INTO IS_OF_A
Values ('200206', 'French', 'Grade 9', 'midterm');
```

```
INSERT INTO IS_OF_A
Values ('200207', 'Philosophy', 'Grade 11', 'first term');
INSERT INTO IS_OF_A
Values ('200208', 'civics', 'Grade 6', 'final term');
INSERT INTO IS_OF_A
Values ('200209', 'English', 'Grade 12', 'midterm');
INSERT INTO IS_OF_A
Values ('200210', 'Arabic', 'Grade 5', 'first term');
end
```

## 17- PARTICIPATES\_IN:

```
begin
INSERT INTO PARTICIPATES_IN
Values ('200201',1,20);
INSERT INTO PARTICIPATES_IN
Values ('200202',2,5);
INSERT INTO PARTICIPATES_IN
Values ('200203',3,15);
INSERT INTO PARTICIPATES_IN
Values ('200204',4,50);
INSERT INTO PARTICIPATES_IN
Values ('200205',5,2);
INSERT INTO PARTICIPATES_IN
Values ('200206',6,23);
INSERT INTO PARTICIPATES_IN
Values ('200207',7,40);
INSERT INTO PARTICIPATES_IN
Values ('200208',8,10);
INSERT INTO PARTICIPATES_IN
Values ('200209',9,60);
INSERT INTO PARTICIPATES_IN
Values ('200210',10,100);
end
```

## 18- ORGANIZES:

```
begin
INSERT INTO ORGANIZES
Values ('10030',1);
INSERT INTO ORGANIZES
Values ('10031',1);
INSERT INTO ORGANIZES
Values ('10032',3);
INSERT INTO ORGANIZES
Values ('10033',4);
INSERT INTO ORGANIZES
Values ('10034',4);
INSERT INTO ORGANIZES
Values ('10035',6);
INSERT INTO ORGANIZES
Values ('10036',7);
INSERT INTO ORGANIZES
Values ('10037',8);
INSERT INTO ORGANIZES
Values ('10038',8);
INSERT INTO ORGANIZES
Values ('10039',10);
end
```

## 19- IS\_TAUGHT:

```
begin
INSERT INTO IS_TAUGHT
Values ('1001','Grade 12','Math') ;
INSERT INTO IS_TAUGHT
Values ('1002','Grade 5','Arabic');
INSERT INTO IS_TAUGHT
Values ('1001','Grade 8','Chemistry');
```

```
INSERT INTO IS_TAUGHT
Values ('1003','Grade 7','Biology');
INSERT INTO IS_TAUGHT
Values ('1002','Grade 6','civics');
INSERT INTO IS_TAUGHT
Values ('1004','Grade 11','Philosophy');
INSERT INTO IS_TAUGHT
Values ('1002','Grade 9','French');
INSERT INTO IS_TAUGHT
Values ('1004','Grade 9','Physics');
INSERT INTO IS_TAUGHT
Values ('1002','Grade 12','English') ;
INSERT INTO IS_TAUGHT
Values ('1003','Grade 12','Acting');
end
```

## 20- TAKES:

```
begin
INSERT INTO TAKES
Values ('Grade 12','English','200204');
INSERT INTO TAKES
Values ('Grade 12','Math','200206');
INSERT INTO TAKES
Values ('Grade 8','Chemistry','200208');
INSERT INTO TAKES
Values ('Grade 8','Chemistry','200209');
INSERT INTO TAKES
Values ('Grade 5','Arabic','200210');
INSERT INTO TAKES
Values ('Grade 11','Philosophy','200207');
INSERT INTO TAKES
Values ('Grade 9','French','200202');
INSERT INTO TAKES
```

```
Values ('Grade 8','Chemistry','200203');  
INSERT INTO TAKES  
Values ('Grade 7','Biology','200205');  
INSERT INTO TAKES  
Values ('Grade 12','Acting','200201');end
```

## 21- TEACHES1:

```
begin  
INSERT INTO TEACHES1  
Values ('200204','10501');  
INSERT INTO TEACHES1  
Values ('200201','10502');  
INSERT INTO TEACHES1  
Values ('200202','10503');  
INSERT INTO TEACHES1  
Values ('200203','10504');  
INSERT INTO TEACHES1  
Values ('200205','10505');  
INSERT INTO TEACHES1  
Values ('200206','10506');  
INSERT INTO TEACHES1  
Values ('200207','10507');  
INSERT INTO TEACHES1  
Values ('200208','10508');  
INSERT INTO TEACHES1  
Values ('200209','10509');  
INSERT INTO TEACHES1  
Values ('200210','10510');  
end
```

## 22- TEACHES2:

```
begin  
INSERT INTO TEACHES2
```

```
Values ('Grade 12','Math','10501',4);
INSERT INTO TEACHES2
Values ('Grade 5', 'Arabic', '10502',3);
INSERT INTO TEACHES2
Values ('Grade 8', 'Chemistry', '10503', 4);
INSERT INTO TEACHES2
Values ('Grade 7', 'Biology', '10504', 4);
INSERT INTO TEACHES2
Values ('Grade 12', 'English', '10505',3);
INSERT INTO TEACHES2
Values ('Grade 9', 'French', '10506',3);
INSERT INTO TEACHES2
Values ('Grade 9', 'Physics', '10507', 5);
INSERT INTO TEACHES2
Values ('Grade 6', 'civics', '10508', 2);
INSERT INTO TEACHES2
Values ('Grade 11', 'Philosophy', '10509', 2);
INSERT INTO TEACHES2
Values ('Grade 12', 'Acting', '10510', 1);
end
```

### 23- WORKS\_FOR:

```
begin
INSERT INTO WORKS_FOR
Values (1,'10040', DATE '2009-07-11');
INSERT INTO WORKS_FOR
Values (1,'10050', DATE '2012-09-19');
INSERT INTO WORKS_FOR
Values (3,'10051', DATE '2017-11-15');
INSERT INTO WORKS_FOR
Values (2, '10053', DATE '2020-05-12');
INSERT INTO WORKS_FOR
Values (2,'10054', DATE '2010-12-22');
```

```
INSERT INTO WORKS_FOR
Values (2,'10055', DATE '2013-01-21');
INSERT INTO WORKS_FOR
Values (3,'10059', DATE '2008-12-18');
INSERT INTO WORKS_FOR
Values (3,'10070', DATE '2018-07-16');
INSERT INTO WORKS_FOR
Values (1,'10005', DATE '2016-08-14');
INSERT INTO WORKS_FOR
Values (2,'10004', DATE '2015-02-23');
INSERT INTO WORKS_FOR
VALUES (2, '10011', DATE '2006-09-04');
INSERT INTO WORKS_FOR
VALUES (2, '10015', DATE '2010-11-09');
end
```

#### 24- STUD\_WARNING:

```
begin
INSERT INTO STUD_WARNING
Values ('200201','Poor attendance');
INSERT INTO STUD_WARNING
Values ('200201','No mask');
INSERT INTO STUD_WARNING
Values ('200203','Fight');
INSERT INTO STUD_WARNING
Values ('200204','No mask');
INSERT INTO STUD_WARNING
Values ('200204','Academic warning');
INSERT INTO STUD_WARNING
Values ('200206','Poor attendance');
INSERT INTO STUD_WARNING
Values ('200207','Fight');
INSERT INTO STUD_WARNING
```

```
Values ('200207','No mask');  
INSERT INTO STUD_WARNING  
Values ('200207','Poor attendance');  
INSERT INTO STUD_WARNING  
Values ('200210','Academic warning');  
end
```

## 25- TEACHER\_LANGUAGE:

```
begin  
INSERT INTO TEACHER_LANGUAGE  
Values ('10501', 'English');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10502', 'English');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10503', 'English');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10503', 'French');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10504', 'English');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10505', 'French');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10506', 'French');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10507', 'English');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10507', 'French');  
INSERT INTO TEACHER_LANGUAGE  
Values ('10509', 'English');  
end
```

## 26- TEACHER\_COMMENT:

```
begin
```

```
INSERT INTO TEACHER_COMMENT
Values ('10501', 'Excellent and professional in communicating with students');
INSERT INTO TEACHER_COMMENT
Values ('10501', 'His lessons were engaging and useful.');
INSERT INTO TEACHER_COMMENT
Values ('10506', 'She paces the class just right so you feel challenged but not overwhelmed.');
INSERT INTO TEACHER_COMMENT
Values ('10508', 'She is patient and eager to help');
INSERT INTO TEACHER_COMMENT
Values ('10503', 'His manner of teaching is so wonderful and refreshing ');
INSERT INTO TEACHER_COMMENT
Values ('10503', 'He is great at building confidence and keeping lessons fun ');
INSERT INTO TEACHER_COMMENT
Values ('10502', 'He has an incredible gift for teaching ');
INSERT INTO TEACHER_COMMENT
Values ('10509', 'Her teaching methods are great ');
INSERT INTO TEACHER_COMMENT
Values ('10505', 'He is very clear and concise ');
INSERT INTO TEACHER_COMMENT
Values ('10505', 'His classes seem like play but there is a lot of serious learning going on.');
end
```

## 27- TEACHER(CG):

```
begin
INSERT INTO TEACHER_CG
Values ('10501', 'Grade 12');
INSERT INTO TEACHER_CG
Values ('10501', 'Grade 11');
INSERT INTO TEACHER_CG
Values ('10502', 'Grade 9');
INSERT INTO TEACHER_CG
Values ('10503', 'Grade 5');
INSERT INTO TEACHER_CG
```

```
Values ('10504', 'Grade 10');
INSERT INTO TEACHER(CG
Values ('10504', 'Grade 8');
INSERT INTO TEACHER(CG
Values ('10505', 'Grade 4');
INSERT INTO TEACHER(CG
Values ('10506', 'Grade 6');
INSERT INTO TEACHER(CG
Values ('10506', 'Grade 7');
INSERT INTO TEACHER(CG
Values ('10507', 'Grade 2');
end
```

## 28- TEACHER DEGREE:

```
begin
INSERT INTO TEACHER_DEGREE
Values ('10501','Bachelor of Mathematics');
INSERT INTO TEACHER_DEGREE
Values ('10502','Bachelor of Arabic Studies');
INSERT INTO TEACHER_DEGREE
Values ('10503','Bachelor of Chemistry');
INSERT INTO TEACHER_DEGREE
Values ('10504','Bachelor of Biology');
INSERT INTO TEACHER_DEGREE
Values ('10505','Bachelor of Mathematics');
INSERT INTO TEACHER_DEGREE
Values ('10506','Bachelor of Cultural Studies');
INSERT INTO TEACHER_DEGREE
Values ('10507','Bachelor of Physics');
INSERT INTO TEACHER_DEGREE
Values ('10508','Bachelor of Civics');
INSERT INTO TEACHER_DEGREE
Values ('10509','Bachelor of Philosophy');
```

```
INSERT INTO TEACHER_DEGREE  
Values ('10510','Bachelor of Acting and Films');  
end
```

### 29- TEACHER\_PHONE:

```
begin  
INSERT INTO TEACHER_PHONE  
Values ('10501', '+96103245987');  
INSERT INTO TEACHER_PHONE  
Values ('10502', '+96181435987');  
INSERT INTO TEACHER_PHONE  
Values ('10502', '+96101098321');  
INSERT INTO TEACHER_PHONE  
Values ('10503', '+96176325981');  
INSERT INTO TEACHER_PHONE  
Values ('10504', '+96171541871');  
INSERT INTO TEACHER_PHONE  
Values ('10505', '+96170876120');  
INSERT INTO TEACHER_PHONE  
Values ('10505', '+96107234019');  
INSERT INTO TEACHER_PHONE  
Values ('10506', '+96181401274');  
INSERT INTO TEACHER_PHONE  
Values ('10507', '+96103126113');  
INSERT INTO TEACHER_PHONE  
Values ('10508', '+96170654231');  
end
```

### 30- BRANCH\_PHONE:

```
begin  
INSERT INTO BRANCH_PHONE  
Values (1,'+96170984231');  
INSERT INTO BRANCH_PHONE
```

```
Values (1,'+96171451873');
INSERT INTO BRANCH_PHONE
Values (1,'+96109671209');
INSERT INTO BRANCH_PHONE
Values (1,'+96109564312');
INSERT INTO BRANCH_PHONE
Values (2,'+96171986534');
INSERT INTO BRANCH_PHONE
Values (2,'+96170984231');
INSERT INTO BRANCH_PHONE
Values (2,'+96105431287');
INSERT INTO BRANCH_PHONE
Values (3,'+96170432111');
INSERT INTO BRANCH_PHONE
Values (3,'+96103444121');
INSERT INTO BRANCH_PHONE
Values (3,'+96107666323');
end
```

### 31- HEALTH\_DISEASE:

```
begin
INSERT INTO HEALTH_DISEASE
Values ('200103','Depression');
INSERT INTO HEALTH_DISEASE
Values ('200104','Diabetes');
INSERT INTO HEALTH_DISEASE
Values ('200104','Asthma');
INSERT INTO HEALTH_DISEASE
Values ('200106','Anxiety');
INSERT INTO HEALTH_DISEASE
Values ('200107','Cellulitis');
INSERT INTO HEALTH_DISEASE
Values ('200107','Diarrhoea');
```

```
INSERT INTO HEALTH_DISEASE
Values ('200109','Cholesterol');
INSERT INTO HEALTH_DISEASE
Values ('200110','Iron deficiency');
end
```

### 32- HEALTH\_ALLERGY:

```
begin
INSERT INTO HEALTH_ALLERGY
Values ('200101','Peanuts allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200101','Fish allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200102','Cats allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200103','Mold allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200103','Dust mites allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200104','Eggs allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200105','Peanuts allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200106','Drug allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200107','Dairy allergy');
INSERT INTO HEALTH_ALLERGY
Values ('200108','Birds allergy');
end
```

### 33- EMPLOYEE\_DEGREE:

```
begin
INSERT INTO EMPLOYEE_DEGREE
```

```
Values ('10010', 'Master degree in business administration');

INSERT INTO EMPLOYEE_DEGREE

Values ('10002', 'Master in Computer science');

INSERT INTO EMPLOYEE_DEGREE

Values ('10013', 'B.S in Computer Science');

INSERT INTO EMPLOYEE_DEGREE

Values ('10013', 'B.S in Math');

INSERT INTO EMPLOYEE_DEGREE

Values ('10011', 'Baccalaureate');

INSERT INTO EMPLOYEE_DEGREE

Values ('10012', 'Baccalaureate');

INSERT INTO EMPLOYEE_DEGREE

Values ('10009', 'B.S in Physics');

INSERT INTO EMPLOYEE_DEGREE

Values ('10009', 'M.S Math');

INSERT INTO EMPLOYEE_DEGREE

Values ('10007', 'B.S in Biology');

INSERT INTO EMPLOYEE_DEGREE

Values ('10008', 'M.S in Actuarial Science');

INSERT INTO EMPLOYEE_DEGREE

Values ('10015', 'Baccalaureate');

INSERT INTO EMPLOYEE_DEGREE

Values ('10016', 'Baccalaureate');

end
```

#### 34- EMPLOYEE\_PHONE:

```
INSERT INTO EMPLOYEE_PHONE

Values ('10002','+96103645476');

INSERT INTO EMPLOYEE_PHONE

Values ('10002','+96101672132');

INSERT INTO EMPLOYEE_PHONE

Values ('10004','+96171593853');

INSERT INTO EMPLOYEE_PHONE
```

```
Values ('10005','+96176462747');
INSERT INTO EMPLOYEE_PHONE
Values ('10005','+96105461718');
INSERT INTO EMPLOYEE_PHONE
Values ('10007','+96176303030');
INSERT INTO EMPLOYEE_PHONE
Values ('10008','+96107363065');
INSERT INTO EMPLOYEE_PHONE
Values ('10009','+96176446453');
INSERT INTO EMPLOYEE_PHONE
Values ('10009','+96109231567');
```

### 35- SUBJECT LANGUAGE:

```
INSERT INTO SUBJECT_LANGUAGE
Values ('Math', 'Grade 12','English');
INSERT INTO SUBJECT_LANGUAGE
Values ('Arabic', 'Grade 5','Arabic');
INSERT INTO SUBJECT_LANGUAGE
Values ('Chemistry', 'Grade 8','English');
INSERT INTO SUBJECT_LANGUAGE
Values ('Biology', 'Grade 7','French');
INSERT INTO SUBJECT_LANGUAGE
Values ('English', 'Grade 12','English');
INSERT INTO SUBJECT_LANGUAGE
Values ('French', 'Grade 9','French');
INSERT INTO SUBJECT_LANGUAGE
Values ('Physics', 'Grade 9','English');
INSERT INTO SUBJECT_LANGUAGE
Values ('civics', 'Grade 6','Arbic');
INSERT INTO SUBJECT_LANGUAGE
Values ('Philosophy', 'Grade 11','English');
INSERT INTO SUBJECT_LANGUAGE
```

Values ('Acting', 'Grade 12','English');

### **36- ORGANIZER\_NAME:**

```
INSERT INTO ORGANIZER_NAME  
Values ('10030',1,'Hussein Ali Bakri');  
INSERT INTO ORGANIZER_NAME  
Values ('10031',1,'Hikmat Toufic Farhat');  
INSERT INTO ORGANIZER_NAME  
Values ('10032',3,'Faisal Adel Al khazen');  
INSERT INTO ORGANIZER_NAME  
Values ('10033',4,'Samer Said Habre');  
INSERT INTO ORGANIZER_NAME  
Values ('10034',4, 'Chadi Kamil Nour');  
INSERT INTO ORGANIZER_NAME  
Values ('10035', 6,'Maha Daoud Tabet');  
INSERT INTO ORGANIZER_NAME\  
Values ('10036',7,'Nadine Fawaz Abbas');  
INSERT INTO ORGANIZER_NAME  
Values ('10037', 8, 'Sanaa Hussein Sharafeddine');  
INSERT INTO ORGANIZER_NAME  
Values ('10038', 8,'Zahraa Ali Sweidan');  
INSERT INTO ORGANIZER_NAME  
Values ('10039', 10,'Tamara Issam Al-Khishen');
```

### **37- ACTIVITY\_GUESTS:**

```
INSERT INTO ACTIVITY_GUESTS  
Values (1,'Raed Mohsen');  
INSERT INTO ACTIVITY_GUESTS  
Values (2,'Ramzi Haraty');  
INSERT INTO ACTIVITY_GUESTS  
Values (3,'Hala Chamoun');  
INSERT INTO ACTIVITY_GUESTS  
Values (3,'Diana Itani');
```

```
INSERT INTO ACTIVITY_GUESTS  
Values (5,'Michel Mawad');  
INSERT INTO ACTIVITY_GUESTS  
Values (5,'Tala Ahmad');  
INSERT INTO ACTIVITY_GUESTS  
Values (7,'Reem Deeb');  
INSERT INTO ACTIVITY_GUESTS  
Values (8,'Christin Karam');  
INSERT INTO ACTIVITY_GUESTS  
Values (8,'Mahmoud Bazzi');  
INSERT INTO ACTIVITY_GUESTS  
Values (10,'Mohamad Bakri');
```

### 38- HAS\_A2:

```
INSERT INTO HAS_A2  
Values ('200201', '+96171102361');  
INSERT INTO HAS_A2  
Values ('200202', '+96103821766');  
INSERT INTO HAS_A2  
Values ('200203', '+96179175075');  
INSERT INTO HAS_A2  
Values ('200204', '+96170552446');  
INSERT INTO HAS_A2  
Values ('200205', '+96107767760');  
INSERT INTO HAS_A2  
Values ('200206', '+96103835537');  
INSERT INTO HAS_A2  
Values ('200207', '+96176659201');  
INSERT INTO HAS_A2  
Values ('200208', '+96170352364');  
INSERT INTO HAS_A2  
Values ('200209', '+96103644678');  
INSERT INTO HAS_A2
```

Values ('200210', '+96176928459');

### **39- DEP\_LOCATION:**

INSERT INTO DEP\_LOCATION

Values (1,'Keserwan/ Building 1/Floor 2');

INSERT INTO DEP\_LOCATION

Values (2,'Keserwan/ Building 2/Floor 1');

INSERT INTO DEP\_LOCATION

Values (3,'Keserwan/ Building 1/Floor 3');

INSERT INTO DEP\_LOCATION

Values (4,'Beirut/ Building 1/Floor 1');

INSERT INTO DEP\_LOCATION

Values (5,'Beirut/ Building 3/Floor 4');

INSERT INTO DEP\_LOCATION

Values (6,'Beirut/ Building 2/Floor 2');

INSERT INTO DEP\_LOCATION

Values (7,'Beirut/ Building 1/Floor 1');

INSERT INTO DEP\_LOCATION

Values (8,'Nabatieh/ Building 1/Floor 2');

INSERT INTO DEP\_LOCATION

Values (9,'Nabatieh/ Building 3/Floor 4');

INSERT INTO DEP\_LOCATION

Values (10,'Nabatieh/ Building 2/Floor 1');

### **Final Tables State:**

#### **1- SUBJECT:**

CLASS_GRADE	SUBJECT_NAME	TYPE	HOURNB
Grade 12	Math	Sciences	7
Grade 5	Arabic	Humanities	5
Grade 8	Chemistry	Sciences	6
Grade 7	Biology	Sciences	6
Grade 12	English	Humanities	5
Grade 9	French	Humanities	5
Grade 9	Physics	Sciences	7
Grade 6	civics	Humanities	1
Grade 11	Philosophy	Humanities	2
Grade 12	Acting	Art	1

## 2- FACILITY:

FACILITYID	CAPACITY	LOCATION	STATUS	TYPE
1001	30	Building 1 floor 2	R	classroom
1002	50	Building 2 floor 2	NR	classroom
1003	45	Building 1 floor 3	R	classroom
1004	25	Building 1 floor 1	NR	classroom
1090	100	Building 2 floor -1	NR	stage
1020	4	Building 3 floor 2	R	Office
1025	4	Building 3 floor 2	R	Office
1060	100	floor 0	NR	Playground
1032	25	Building 3 floor 1	R	Lab
1038	40	Building 3 floor 1	NR	Lab

## 3- PARENT:

PHONE	PNAME	DOB	GENDER	PROFESSION	ADDRESS
+96176659201	Tamara Younnes	10/20/1982	F	Doctor	Beirut/ Hadath
+96170352364	Hoda Soboh	12/29/1979	F	Lawyer	Beirut/ Antelyas
+96103644678	Mariam Baaklini	11/23/1978	F	Engineer	Keserwen/ Harajel
+96176928459	Lea Dia	08/21/1981	F	nurse	Keserwen/ Mayrouba
+96170552446	Ali Cheaib	09/24/1979	M	Actor	Nabatieh/ Al Sharkiyeh
+96107767760	Samer Wehbe	08/13/1975	M	Carpenter	Beirut/ Hamra
+96103835537	Najwa Houmani	02/19/1981	F	waitor	Beirut/ Kraytem
+96171102361	Bassam Hamze	02/11/1980	M	Architect	Nabatieh/ Kfarroman
+96103821766	Allam Jaber	12/01/1976	M	Business man	Nabatieh/ Mayfadoun
+96179175075	Mohammad Kdouh	05/15/1977	M	Clothes Seller	Nabatieh/ Hay El Bayad

## 4- BRANCH:

BRANCHNB	FAX	CITY	STREET	ZIPCODE
1	556-3219	Keserwen	Snoubar	2222
2	874-2566	Beirut	Emile Edde	3333
3	542-3261	Nabatieh	Dennawi	4444

## 5- EMPLOYEE:

EID	ENAME	PROFESSION	EMAIL	GENDER	DOB	ADDRESS	SALARY	SS_ID	SUPERVISORID
10004	Nader Kassem Khadaj	Faculty Advisor	Nader_85_@hotmail.com	M	03/10/1985	Beirut/ Hamra	2200	yes	10018
10005	Maya Omar Darwich	Coordinator	Maya123_@hotmail.com	F	12/11/1989	Keserwen/ Faraya	2500	yes	10018
10006	Christina Ali Kassem	Coordinator	Chris_Kassem@hotmail.com	F	01/13/1983	Keserwen/ Ajaltoun	2000	yes	10018
10018	Wael Jihad Saad	Principal	waelJ1_@hotmail.com	M	10/15/1962	Beirut/ Bir Hasan	3200	yes	-
10010	Jana Ayad Jaber	Chairperson	jana.jaber03@hotmail.com	F	10/28/1990	Beirut/ Zaytouna bay	4000	yes	-
10002	Mohammad Ali Jaber	Coordinator	moe.jaber99@hotmail.com	M	11/03/1988	Beirut/ Roumieh	2000	yes	10018
10003	Mohammad Tarek Rahal	Academic Assistant	moe_rahal@hotmail.com	M	04/20/1980	Beirut/ Kraytem	2000	yes	10018
10007	Wissam Walid Baaklini	Academic Assistant	Wissam_92@hotmail.com	M	05/08/1992	Keserwen/ baallouneh	2000	yes	10018
10008	Khodor Ali Ghalem	Faculty Advisor	Khodor_GH@hotmail.com	M	07/14/1975	Beirut/ Baabda	2200	yes	10018
10009	Yasmine AL-Ayache	Academic Assistant	YasmineAyache09_@hotmail.com	F	08/18/1972	Beirut/ Aley	2500	yes	10018
10001	Bahaa Raed Chaar	Faculty Advisor	Chhaar468_@hotmail.com	M	11/21/1970	Beirut/ Aley	2200	yes	10018
10011	Charbel Joseph Younes	Security Guard	Charbel7676@hotmail.com	M	08/01/1980	Beirut/ Sin El Fil	1500	yes	10018
10012	Rawad Ayad Safa	Security Guard	Kheir_R@hotmail.com	M	09/11/1968	Nabatieh/ Mayfadoun	1500	yes	10018
10013	Mohammad Ali Shamas	Information technology	Shamas_Moe_@hotmail.com	M	02/09/1985	Beirut/ Hazmieh	3000	yes	10018
10014	Maha Mahmoud Chakaron	Custodian	-	F	08/01/1980	Nabatieh/ Kfarromen	1000	yes	10018
10015	Mira Jamil Issa	cleaner	-	F	05/07/1972	Beirut/ Jal Al Dib	1000	yes	10014
10016	Ali Sami Sharara	Bus driver	Sharara_Ali89_@hotmail.com	M	12/25/1969	Saida	1200	yes	10018
10017	Diala Jamal Abdo	School nurse	diala014@hotmail.com	F	05/02/1982	Beirut/ Manara	3000	yes	10018
10030	Hussein Ali Bakri	Teacher	Bakri75_@hotmail.com	M	12/10/1975	Beirut/ Antelyas	4500	yes	10010
10031	Hikmat Toufic Farhat	Teacher	Hikmat08_@hotmail.com	M	06/06/1970	Nabatieh/ Kfarjoz	4500	yes	10010
10032	Faisal Adel Al khazen	Teacher	Faisal434_@hotmail.com	M	07/10/1965	Beirut/ Hamra	4500	yes	10010
10033	Samer Said Habre	Teacher	Habre_Samer_@hotmail.com	M	05/16/1968	Beirut/ Hadath	4500	yes	10010
10034	Chadi Kamil Nour	Teacher	Nour76_@hotmail.com	M	10/10/1976	Beirut/ Kornich Al Mazraa	4500	yes	10010
10035	Maha Daoud Tabet	Teacher	Maha.Tabet_@hotmail.com	F	08/19/1980	Beirut/ Kmataeye	4500	yes	10010
10036	Nadine Fawaz Abbas	Teacher	Nadine.Fawaz_@hotmail.com	F	03/11/1988	Beirut/ Hamra	4500	yes	10010
10037	Sanaa Hussein Sharafeddine	Teacher	Sanaa.H_@hotmail.com	F	07/03/1985	Beirut/ Jnah	4500	yes	10010
10038	Zahraa Ali Sweidan	Teacher	Zahraa.Sweidan02_@hotmail.com	F	03/17/1990	Beirut/ Khalde	4500	yes	10010
10039	Tamara Issam Al-Khishen	Teacher	Tamara.09_@hotmail.com	F	09/12/1985	Beirut/ Aley	4500	yes	10010
10040	Dana Elie Doumit	Supervisor	Dana_@hotmail.com	F	09/11/1965	Keserwen/ Faraya	3500	yes	-
10050	Bassem Ali Salman	Chairperson	Bassem1_@hotmail.com	M	09/20/1962	Baalbeck/Chmestar	4000	yes	-
10051	Rayan Wissam Shehadeh	Chairperson	RShehadeh1_@hotmail.com	M	05/06/1969	Saida/ Khartoum	4000	yes	-
10052	Majd Hamadan Fares	Chairperson	M_75Fares_@hotmail.com	M	10/10/1975	Sour/ Al Host	4000	yes	-
10053	Mohammad Hisham Ozeir	Chairperson	MoeOzeir1_@hotmail.com	M	12/15/1980	Beirut/ Airport Road	5000	yes	-
10054	Karim Elie Akar	Chairperson	Akar451_@hotmail.com	M	02/09/1985	Beirut/ Cheyah	3800	yes	-
10055	Nada Hasan Saab	Chairperson	Nada.Saab7_@hotmail.com	F	01/29/1970	Beirut/ Hazmieh	5000	yes	-
10056	Sara Said Ajami	Chairperson	Ajami.Sara_@hotmail.com	F	11/13/1987	Beirut/ Chouifat	4000	yes	-
10057	Hussein Hasan Mawla	Chairperson	Mawla12_@hotmail.com	M	08/02/1973	Beirut/ Ghobeiry	4000	yes	-
10058	Mona Jaber Harajli	Chairperson	Mona_J_@hotmail.com	F	09/14/1971	Beirut/ Dbayeh	4000	yes	-
10059	Reem Abdallah Chemaly	Chairperson	Reem.Chemaly_@hotmail.com	F	04/03/1988	Saida/ Kfarhata	4000	yes	-
10070	Jad Samer Houmani	Supervisor	Jad34_@hotmail.com	M	01/11/1970	Nabatieh/ Al Sharkiyeh	3500	yes	-
10060	Jamil Ali Bdeir	Supervisor	Jamil_@hotmail.com	M	01/12/1962	Beirut/ Hadath	3500	yes	-

## 6- DEPENDENT:

DEPENDENTID	NAME	GENDER	DOB	RELATIONSHIP	EMPLOYEEID
20201	Mona Rahal	F	03/16/2012	Daughter	10003
20202	Sawsan Khaddaj	F	07/19/1983	Fiancee	10004
20203	Diala Mokbel	F	08/22/1992	Wife	10007
20204	Ahlam Ghalem	F	12/29/2006	Daughter	10018
20205	Samih Ghalem	M	01/21/2009	Son	10018
20206	Halim Bazzi	M	11/01/1971	Husband	10009
20207	Hadi Bazzi	M	01/04/1988	Brother	10010
20208	Fatima Safa	F	11/14/2002	Daughter	10012
20209	Natalia Fares	F	05/03/1980	Wife	10052
20210	Ramzi Sharafeddin	M	04/09/1999	Son	10037

## 7- STUDENT:

SID	FIRSTNAME	LASTNAME	MIDDLENAME	GENDER	DOB	ADDRESS	CLASS_GRADE	RANK	LANGUAGE	SUPERVISORID	BRANCH_NUM
200201	Tarek	Hamze	Bassam	M	12/05/2004	Nabatieh/ Kafarouman	Grade 12	2	French	10070	3
200202	Zein	Jaber	Allam	M	12/28/2006	Nabatieh/ Mayfadoun	Grade 9	10	French	10070	3
200203	Israa	Kdouh	Mohammad	F	03/05/2007	Nabatieh/ Hay El Bayed	Grade 8	5	English	10070	3
200204	Mohammad	Cheab	Ali	M	10/22/2004	Nabatieh/ Al Sharkiyeh	Grade 12	1	French	10070	3
200205	Jana	Wehbe	Samer	F	11/28/2009	Beirut/ Hamra	Grade 7	4	French	10060	2
200206	Farah	Jaber	Ayad	F	04/05/2006	Beirut/ Kraytem	Grade 9	2	English	10060	2
200207	Hussein	Hayek	Hassan	M	03/10/2005	Beirut/ Hadath	Grade 11	7	English	10060	2
200208	Joey	Youness	Haysam	M	02/09/2010	Beirut/ Antelyas	Grade 6	4	French	10060	2
200209	Issa	Hassan	Hussein	M	07/20/2004	Keserwen/ Harajel	Grade 12	10	English	10040	1
200210	Joelle	Hammond	Bilal	F	01/15/2012	Keserwen/ Mayrouba	Grade 5	7	French	10040	1

## 8- TEACHER:

TID	FIRSTNAME	MIDDLENAME	LASTNAME	GENDER	DOB	ADDRESS	EMAIL	SALARY	SS_IDENT	EMPLOYEEID
10501	Hussein	Ali	Bakri	M	12/10/1975	Beirut/ Antelyas	Bakri75_@hotmail.com	4500\$	yes	10030
10502	Hikmat	Toufic	Farhat	M	06/06/1970	Nabatieh/ Kfarjoz	Hikmat08_@hotmail.com	4500\$	yes	10031
10503	Faisal	Adel	Al khazen	M	07/10/1965	Beirut/ Hamra	Faisal434_@hotmail.com	4500\$	yes	10032
10504	Samer	Said	Habre	M	05/16/1968	Beirut/ Hadath	Habre_Samer_@hotmail.com	4500\$	yes	10033
10505	Chadi	Kamil	Nour	M	10/10/1976	Beirut/ Kornich Al Mazraa	Nour76_@hotmail.com	4500\$	yes	10034
10506	Maha	Daoud	Tabel	F	08/19/1980	Beirut/ Kmatelye	Maha.Tabel_@hotmail.com	4500\$	yes	10035
10507	Nadine	Fawaz	Abbas	F	03/11/1988	Beirut/ Hamra	Nadine Abbas1_@hotmail.com	4500\$	yes	10036
10508	Sanaa	Hussein	Sharfedine	F	07/03/1985	Beirut/ Jnah	Sanaa.H_@hotmail.com	4500\$	yes	10037
10509	Zahraa	Ali	Sweidan	F	03/17/1990	Beirut/ Khalde	Zahraa.Sweidan02_@hotmail.com	4500\$	yes	10038
10510	Tamara	Issam	Al-Khishen	F	09/12/1985	Beirut/ Aley	Tamara.09_@hotmail.com	4500\$	yes	10039

## 9- ACTIVITY:

ACTIVITYNUMBER	TYPE	OCCASION	EXPENSE	FACILITYID
1	Songs	Independence	1000\$	1060
2	Songs	Christmas	1000\$	1090
3	Marathon	Sports day	500\$	1060
4	Songs	End of the year	1000\$	1060
5	Mentoring	-	200\$	1002
6	Acting	Christmas	2000\$	1090
7	Acting	Mothers Day	2000\$	1090
8	Robotics	-	2000\$	1060
9	Debate	MUN	1300\$	1090
10	Academic	competiton	500\$	1002

## 10- HEALTH\_RECORD:

RECORDID	WEIGHT	HEIGHT	GENDER	AGE	BLOOD_TYPE	STUDENTID
200101	70kg	180cm	M	18	O+	200201
200102	65kg	170cm	M	16	O-	200202
200103	50kg	130cm	F	15	AB+	200203
200104	80kg	182cm	M	18	AB-	200204
200105	40kg	145cm	F	13	A+	200205
200106	45kg	165cm	F	16	A-	200206
200107	65kg	175cm	M	17	O+	200207
200108	42kg	162cm	F	12	O-	200208
200109	70kg	170cm	M	18	AB+	200209
200110	37kg	130cm	F	10	AB-	200210

## 11- ACCOUNT:

ACCOUNT_NB	REMAINING_FEES	DEPOSIT	STARTDATE	ENDDATE	STUDID
101001	700	200	01/04/2015	06/12/2021	200201
101002	1000	300	11/14/2010	03/11/2022	200202
101003	2000	200	04/03/2011	08/19/2023	200203
101004	500	300	11/06/2008	12/12/2020	200204
101005	2300	200	10/04/2012	10/22/2022	200205
101006	7000	300	04/25/2010	12/23/2023	200206
101007	100	300	02/03/2009	07/12/2020	200207
101008	900	300	12/29/2015	05/17/2024	200208
101009	3000	200	10/13/2009	03/14/2021	200209
101010	750	200	03/09/2016	05/12/2026	200210

## 12- GRADE:

STUDID	SUBJ_NAME	CLASS_GRADE	TYPE	FIRST_TERM	MID_TERM	FINAL_TERM	TOTAL	LETTER_GRADE
200201	Math	Grade 12	first term	95	90	100	95	A
200202	Physics	Grade 9	final term	75	80	55	70	C
200203	Chemistry	Grade 8	first term	86	60	75	80.33	B
200204	Acting	Grade 12	midterm	97	95	100	98	A
200205	Biology	Grade 7	final term	90	90	80	86.66	B
200206	French	Grade 9	midterm	95	98	100	97.66	A
200207	Philosophy	Grade 11	first term	80	78	82	80	B
200208	civics	Grade 6	final term	85	87	82	84.66	B
200209	English	Grade 12	midterm	65	70	71	68.66	D
200210	Arabic	Grade 5	first term	65	74	73	70.66	C

### 13- PAYMENT:

ACCOUNT_NB	PAYMENT_METHOD
101001	Cash/ LB
101002	Cash/ USD
101003	Check/ LB
101004	Check/ USD
101005	Cash/ LB
101006	Cash/ USD
101007	Check/ LB
101008	Check/ USD
101009	Cash/LB
101010	Cash/ USD

### 14- DEPARTMENT:

DNAME	DNUMBER	PHONE	FAX	EMPLOYEEID	MANAGERID	START_DATE
Research	1	+96170987432	009-8712	20	10050	02/11/2013
Biology	2	+96170930431	008-2516	10	10051	09/18/2005
Athletics	3	+96178030733	007-2130	15	10052	12/08/2012
Mathematics	4	+96181100431	006-1274	10	10053	02/03/2007
Theatre/Dance	5	+96181654044	005-1374	12	10054	01/25/2013
Arts/Humanities	6	+96170787317	004-0154	10	10055	11/20/2007
Chemistry	7	+96171870398	003-9812	20	10056	10/10/2009
Social Sciences	8	+96170921401	002-5423	10	10057	06/08/2006
Languages	9	+96171876120	002-4171	20	10058	12/09/2010
Arabic	10	+96181876410	001-1112	10	10059	03/18/2011

### 15- HAS\_A1:

STUDID	SUBJ_NAME	CLASS_GRADE	TYPE
200201	Math	Grade 12	first term
200202	Physics	Grade 9	final term
200203	Chemistry	Grade 8	first term
200204	Acting	Grade 12	midterm
200205	Biology	Grade 7	final term
200206	French	Grade 9	midterm
200207	Philosophy	Grade 11	first term
200208	civics	Grade 6	final term
200209	English	Grade 12	midterm
200210	Arabic	Grade 5	first term

## 16- IS\_OF\_A:

STUDID	SUBJ_NAME	CLASS_GRADE	TYPE
200201	Math	Grade 12	first term
200202	Physics	Grade 9	final term
200203	Chemistry	Grade 8	first term
200204	Acting	Grade 12	midterm
200205	Biology	Grade 7	final term
200206	French	Grade 9	midterm
200207	Philosophy	Grade 11	first term
200208	civics	Grade 6	final term
200209	English	Grade 12	midterm
200210	Arabic	Grade 5	first term

## 17- PARTICIPATES\_IN:

STUDID	ACTIVITY_NB	PARTICIPANTS_NB
200201	1	20
200202	2	5
200203	3	15
200204	4	50
200205	5	2
200206	6	23
200207	7	40
200208	8	10
200209	9	60
200210	10	100

## 18- ORGANIZES:

EMPLOYEEID	ACTIVITY_NB
10030	1
10031	1
10032	3
10033	4
10034	4
10035	6
10036	7
10037	8
10038	8
10039	10

## 19- IS TAUGHT:

FACILITYID	CLASS_GRADE	SUBJECT_NAME
1004	Grade 11	Philosophy
1003	Grade 12	Acting
1002	Grade 12	English
1001	Grade 12	Math
1002	Grade 5	Arabic
1002	Grade 6	civics
1003	Grade 7	Biology
1001	Grade 8	Chemistry
1002	Grade 9	French
1004	Grade 9	Physics

## 20- TAKES:

CLASS_GRADE	SUBJECT_NAME	STUDID
Grade 11	Philosophy	200207
Grade 12	Acting	200201
Grade 12	English	200204
Grade 12	Math	200206
Grade 5	Arabic	200210
Grade 7	Biology	200205
Grade 8	Chemistry	200203
Grade 8	Chemistry	200208
Grade 8	Chemistry	200209
Grade 9	French	200202

## 21- TEACHES1:

STUDID	TEACHERID
200201	10502
200202	10503
200203	10504
200204	10501
200205	10505
200206	10506
200207	10507
200208	10508
200209	10509
200210	10510

## 22- TEACHES2:

CLASS_GRADE	SUBJECT_NAME	TEACHERID	HOUR_NB
Grade 12	Math	10501	4
Grade 5	Arabic	10502	3
Grade 8	Chemistry	10503	4
Grade 7	Biology	10504	4
Grade 12	English	10505	3
Grade 9	French	10506	3
Grade 9	Physics	10507	5
Grade 6	civics	10508	2
Grade 11	Philosophy	10509	2
Grade 12	Acting	10510	1

## 23- WORKS\_FOR:

BRANCH_NB	EMPLOYEEID	START_DATE
1	10040	07/11/2009
1	10050	09/19/2012
3	10051	11/15/2017
2	10053	05/12/2020
2	10054	12/22/2010
2	10055	01/21/2013
3	10059	12/18/2008
3	10070	07/16/2018
1	10005	08/14/2016
2	10004	02/23/2015
2	10011	09/04/2006
2	10015	11/09/2010

## 24- STUD\_WARNING:

STUDID	WARNING
200201	No mask
200201	Poor attendance
200203	Fight
200204	Academic warning
200204	No mask
200206	Poor attendance
200207	Fight
200207	No mask
200207	Poor attendance
200210	Academic warning

## 25- TEACHER\_LANGUAGE:

TEACHERID	LANGUAGE
10501	English
10502	English
10503	English
10503	French
10504	English
10505	French
10506	French
10507	English
10507	French
10509	English

## 26- TEACHER\_COMMENT:

TEACHERID	TEACHER_COMMENT
10501	Excellent and professional in communicating with students
10501	His lessons were engaging and useful.
10502	He has an incredible gift for teaching
10503	He is great at building confidence and keeping lessons fun
10503	His manner of teaching is so wonderful and refreshing
10505	He is very clear and concise
10505	His classes seem like play but there is a lot of serious learning going on.
10506	She paces the class just right so you feel challenged but not overwhelmed.
10508	She is patient and eager to help
10509	Her teaching methods are great

## 27- TEACHER(CG:

TEACHERID	CLASS_GRADE
10501	Grade 11
10501	Grade 12
10502	Grade 9
10503	Grade 5
10504	Grade 10
10504	Grade 8
10505	Grade 4
10506	Grade 6
10506	Grade 7
10507	Grade 2

## 28- TEACHER\_DEGREE:

TEACHERID	DEGREE
10501	Bachelor of Mathematics
10502	Bachelor of Arabic Studies
10503	Bachelor of Chemistry
10504	Bachelor of Biology
10505	Bachelor of Mathematics
10506	Bachelor of Cultural Studies
10507	Bachelor of Physics
10508	Bachelor of Civics
10509	Bachelor of Philosophy
10510	Bachelor of Acting and Films

## 29- TEACHER\_PHONE:

TEACHERID	PHONE
10501	+96103245987
10502	+96101098321
10502	+96181435987
10503	+96176325981
10504	+96171541871
10505	+96107234019
10505	+96170876120
10506	+96181401274
10507	+96103126113
10508	+96170654231

## 30- BRANCH\_PHONE:

BRANCH_NB	PHONE
1	+96109564312
1	+96109671209
1	+96170984231
1	+96171451873
2	+96105431287
2	+96170984231
2	+96171986534
3	+96103444121
3	+96107666323
3	+96170432111

### 31- HEALTH\_DISEASE:

RECORDID	DISEASE
200103	Depression
200104	Asthma
200104	Diabetes
200106	Anxiety
200107	Cellulitis
200107	Diarrhoea
200109	Cholesterol
200110	Iron deficiency

### 32- HEALTH\_ALLERGY:

RECORDID	ALLERGY
200101	Fish allergy
200101	Peanuts allergy
200102	Cats allergy
200103	Dust mites allergy
200103	Mold allergy
200104	Eggs allergy
200105	Peanuts allergy
200106	Drug allergy
200107	Dairy allergy
200108	Birds allergy

### 33- EMPLOYEE\_DEGREE:

EMPLOYEEID	DEGREE
10002	Master in Computer science
10007	B.S in Biology
10008	M.S in Actuarial Science
10009	B.S in Physics
10009	M.S Math
10010	Master degree in business administration
10011	Baccalaureate
10012	Baccalaureate
10013	B.S in Computer Science
10013	B.S in Math
10015	Baccalaureate
10016	Baccalaureate

#### 34- EMPLOYEE\_PHONE:

EMPLOYEEID	PHONE
10002	+96101672132
10002	+96103645476
10004	+96171593853
10005	+96105461718
10005	+96176462747
10007	+96176303030
10008	+96107363065
10009	+96109231567
10009	+96176446453

#### 35- SUBJECT\_LANGUAGE:

SUBJECT_NAME	CLASS_GRADE	LANGUAGE
Acting	Grade 12	English
Arabic	Grade 5	Arabic
Biology	Grade 7	French
Chemistry	Grade 8	English
English	Grade 12	English
French	Grade 9	French
Math	Grade 12	English
Philosophy	Grade 11	English
Physics	Grade 9	English
civics	Grade 6	Arbic

#### 36- ORGANIZER\_NAME:

EMPLOYEEID	ACTIVITY_NB	ORGANIZER_NAME
10030	1	Hussein Ali Bakri
10031	1	Hikmat Toufic Farhat
10032	3	Faisal Adel Al khazen
10033	4	Samer Said Habre
10034	4	Chadi Kamil Nour
10035	6	Maha Daoud Tabet
10036	7	Nadine Fawaz Abbas
10037	8	Sanaa Hussein Sharafeddine
10038	8	Zahraa Ali Sweidan
10039	10	Tamara Issam Al-Khishen

### 37- ACTIVITY\_GUESTS:

ACTIVITYNB	GUEST_NAME
1	Raed Mohsen
2	Ramzi Haraty
3	Diana Itani
3	Hala Chamoun
5	Michel Mawad
5	Tala Ahmad
7	Reem Deeb
8	Christin Karam
8	Mahmoud Bazzi
10	Mohamad Bakri

### 38- HAS\_A2:

STUDID	PARENT_PHONE
200201	+96171102361
200202	+96103821766
200203	+96179175075
200204	+96170552446
200205	+96107767760
200206	+96103835537
200207	+96176659201
200208	+96170352364
200209	+96103644678
200210	+96176928459

## **39- DEP\_LOCATION:**

DEPARTMENT_NB	LOCATION
1	Keserwan/ Building 1/Floor 2
2	Keserwan/ Building 2/Floor 1
3	Keserwan/ Building 1/Floor 3
4	Beirut/ Building 1/Floor 1
5	Beirut/ Building 3/Floor 4
6	Beirut/ Building 2/Floor 2
7	Beirut/ Building 1/Floor 1
8	Nabatieh/ Building 1/Floor 2
9	Nabatieh/ Building 3/Floor 4
10	Nabatieh/ Building 2/Floor 1

## **Queries:**

### **Query 1:**

Retrieve the name and the total final grade of students who have obtained a letter grade of 'A' on any subject, and show the resulting remaining fees of these students if they are given a 10 percent discount along with their payment method.

```
SELECT STUDENT.firstName, STUDENT.lastName, GRADE.TOTAL AS TOTAL_GRADE,  
ACCOUNT.remaining_fees * 0.9 AS UPDATED_REMAININGFEES, PAYMENT.payment_method  
FROM STUDENT, GRADE, ACCOUNT, PAYMENT  
WHERE STUDENT.SID = GRADE.STUDID AND GRADE.letter_Grade = 'A' AND  
ACCOUNT.STUDID = STUDENT.SID AND PAYMENT.account_nb = ACCOUNT.account_nb;
```

### **Output:**

FIRSTNAME	LASTNAME	TOTAL_GRADE	UPDATED_REMAININGFEES	PAYMENT_METHOD
Tarek	Hamze	95	630	Cash/ LB
Mohammad	Cheib	98	450	Check/ USD
Farah	Jaber	97.66	6300	Cash/ USD

### **Query 2:**

Retrieve the name and the ID of the students who have more than 2 warnings and the name of a teacher who teaches him/her along with the name of his/her supervisor in order to decide on what actions should be taken regarding the student behavior.

```

SELECT EMPLOYEE.Ename AS SUPERVISOR_NAME, TEACHER.firstName AS
TEACHER_FIRSTNAME, TEACHER.lastName AS TEACHER_LastNAME, STUDENT.firstName AS
STUDENT_FIRSTNAME, STUDENT.lastName AS STUDENT_LASTNAME, SID AS STUDENT_ID,
COUNT(*) AS NUMBER_OF_WARNINGS

FROM STUD_WARNING, STUDENT, TEACHES1, TEACHER, EMPLOYEE

WHERE TEACHES1.STUDID = STUDENT.SID AND STUD_WARNING.STUDID = STUDENT.SID
AND TEACHER.TID = TEACHES1.TEACHERID AND EMPLOYEE.EID =
STUDENT.SUPERVISORID

GROUP BY EMPLOYEE.Ename, TEACHER.lastName, TEACHER.firstName, STUDENT.firstName,
STUDENT.lastName, SID
HAVING COUNT(*) > 1;

```

#### **Output:**

SUPERVISOR_NAME	TEACHER_FIRSTNAME	TEACHER_LASTNAME	STUDENT_FIRSTNAME	STUDENT_LASTNAME	STUDENT_ID	NUMBER_OF_WARNINGS
Jad Samer Houmani	Hikmat	Farhat	Tarek	Hamze	200201	2
Jamil Ali Bdeir	Nadine	Abbas	Hussein	Hayek	200207	3
Jad Samer Houmani	Hussein	Bakri	Mohammad	Cheab	200204	2

#### **Query 3:**

Retrieve the subject name, class grade, and type of every subject that is taught in the facility of facility ID ‘1004’, in addition to the name of the teacher who teaches this subject, for the purpose of servicing this facility (classroom).

```

SELECT TEACHER.firstName AS TEACHER_FIRSTNAME, TEACHER.lastName AS
TEACHER_LASTNAME, SUBJECT.SUBJECT_NAME, SUBJECT.CLASS_GRADE,
SUBJECT.TYPE, FACILITY.FACILITYID

```

FROM SUBJECT, FACILITY, IS\_TAUGHT, TEACHES2, TEACHER

WHERE IS\_TAUGHT.FACILITYID = '1004' AND IS\_TAUGHT.SUBJECT\_NAME =  
SUBJECT.SUBJECT\_NAME AND IS\_TAUGHT.CLASS\_GRADE = SUBJECT.CLASS\_GRADE  
AND IS\_TAUGHT.FACILITYID = FACILITY.FACILITYID AND  
SUBJECT.SUBJECT\_NAME = TEACHES2.SUBJECT\_NAME AND SUBJECT.CLASS\_GRADE =  
TEACHES2.CLASS\_GRADE AND TEACHES2.TEACHERID = TEACHER.TID;

**Output:**

TEACHER_FIRSTNAME	TEACHER_LASTNAME	SUBJECT_NAME	CLASS_GRADE	TYPE	FACILITYID
Zahraa	Sweidan	Philosophy	Grade 11	Humanities	1004
Nadine	Abbas	Physics	Grade 9	Sciences	1004

**Query 4:**

Retrieve the name, ID, parent phone number, and health record ID of every student who suffers from more than 2 diseases to be able to communicate with his/her parent to take the appropriate measures regarding his/her health and in case of any emergency.

SELECT STUDENT.firstName AS STUDENT\_FIRSTNAME, STUDENT.lastName AS STUDENT\_LASTNAME, STUDENT.SID AS STUDENT\_ID, COUNT(\*) AS NUMBER\_OF\_DISEASES, HEALTH\_RECORD.RECORDID AS HEALTH\_RECORD\_ID,  
HAS\_A2.PARENT\_PHONE  
FROM HEALTH\_DISEASE, HEALTH\_RECORD, STUDENT, HAS\_A2  
WHERE HEALTH\_DISEASE.RECORDID = HEALTH\_RECORD.RECORDID AND  
HEALTH\_RECORD.STUDENTID = STUDENT.SID AND HAS\_A2.STUDID = STUDENT.SID  
GROUP BY STUDENT.SID, HEALTH\_RECORD.RECORDID, HAS\_A2.PARENT\_PHONE,  
STUDENT.firstName, STUDENT.lastName  
HAVING COUNT(\*) > 1

**Output:**

STUDENT_FIRSTNAME	STUDENT_LASTNAME	STUDENT_ID	NUMBER_OF_DISEASES	HEALTH_RECORD_ID	PARENT_PHONE
Hussein	Hayek	200207	2	200107	+96176659201
Mohammad	Cheib	200204	2	200104	+96170552446

**Query 5:**

Retrieve all the details of every outdoor activity that took place in the facility of facility ID '1060' (playground) including the activity number, type, occasion, expense, guest names, organizer names, and participants number.

```

SELECT ACTIVITY.*, ACTIVITY_GUESTS.GUEST_NAME,
ORGANIZER_NAME.ORGANIZER_NAME, PARTICIPATES_IN.PARTICIPANTS_NB
FROM ACTIVITY, ACTIVITY_GUESTS, ORGANIZER_NAME, PARTICIPATES_IN
WHERE ACTIVITY.facilityid = '1060' AND ACTIVITY_GUESTS.ACTIVITYNB =
ACTIVITY.activityNumber AND ORGANIZER_NAME.ACTIVITY_NB = ACTIVITY.activityNumber
AND
PARTICIPATES_IN.ACTIVITY_NB = ACTIVITY.activityNumber;

```

#### **Output:**

ACTIVITYNUMBER	TYPE	OCCASION	EXPENSE	FACILITYID	GUEST_NAME	ORGANIZER_NAME	PARTICIPANTS_NB
1	Songs	Independence	1000\$	1060	Raed Mohsen	Hussein Ali Bakri	20
1	Songs	Independence	1000\$	1060	Raed Mohsen	Hikmat Toufic Farhat	20
3	Marathon	Sports day	500\$	1060	Hala Chamoun	Faisal Adel Al khazen	15
3	Marathon	Sports day	500\$	1060	Diana Itani	Faisal Adel Al khazen	15
8	Robotics	-	2000\$	1060	Mahmoud Bazzi	Sanaa Hussein Sharafeddine	10
8	Robotics	-	2000\$	1060	Christin Karam	Sanaa Hussein Sharafeddine	10
8	Robotics	-	2000\$	1060	Mahmoud Bazzi	Zahraa Ali Sweidan	10
8	Robotics	-	2000\$	1060	Christin Karam	Zahraa Ali Sweidan	10

#### **Query 6:**

The business office requests to retrieve the name and ID of each student whose net tuition fees is more than 1000\$, in addition to their parent phone number to communicate with regarding their financial situation.

```

SELECT STUDENT.SID AS STUDENT_ID, STUDENT.firstName AS STUDENT_FIRSTNAME,
STUDENT.lastName AS STUDENT_LASTNAME, PARENT.PNAME AS PARENT_NAME,
HAS_A2.PARENT_PHONE, (remaining_fees - deposit) AS NET_TOTAL_FEES
FROM ACCOUNT, STUDENT, HAS_A2, PARENT
WHERE ACCOUNT.STUDID = STUDENT.SID AND HAS_A2.STUDID = STUDENT.SID AND
HAS_A2.PARENT_PHONE = PARENT.PHONE AND ((remaining_fees) - (deposit)) > 1000;

```

#### **Output:**

STUDENT_ID	STUDENT_FIRSTNAME	STUDENT_LASTNAME	PARENT_NAME	PARENT_PHONE	NET_TOTAL_FEES
200209	Issa	Hassan	Mariam Baaklini	+96103644678	2800
200205	Jana	Wehbe	Samer Wehbe	+96107767760	2100
200206	Farah	Jaber	Najwa Houmani	+96103835537	6700
200203	Israa	Kdouh	Mohammad Kdouh	+96179175075	1800

### Query 7:

Retrieve the name and the ID of the employee(s) who works as a manager for a certain department in the Beirut branch of the school and earning the maximum salary among all other managers of different departments in this branch. Also, retrieve the number and the name of the department he/she is working for.

```

SELECT DEPARTMENT.DNUMBER AS DEPARTMENT_NUMBER, DEPARTMENT.DNAME AS
DEPARTMENT_NAME, EMPLOYEE.EID AS EMPLOYEE_ID, EMPLOYEE.ENAME AS
EMPLOYEE_NAME, EMPLOYEE.SALARY AS MAXIMUM_SALARY_OFFERED
FROM DEP_LOCATION, DEPARTMENT, EMPLOYEE

WHERE DEP_LOCATION.department_nb = DEPARTMENT.DNUMBER AND
DEP_LOCATION.LOCATION LIKE '%Beirut%' AND
DEPARTMENT.MANAGERID = EMPLOYEE.EID AND
SALARY IN ( SELECT MAX(SALARY)
            FROM EMPLOYEE, DEPARTMENT
            WHERE DEPARTMENT.MANAGERID = EMPLOYEE.EID);
    
```

### **Output:**

DEPARTMENT_NUMBER	DEPARTMENT_NAME	EMPLOYEE_ID	EMPLOYEE_NAME	MAXIMUM_SALARY_OFFERED
4	Mathematics	10053	Mohammad Hisham Ozeir	5000
6	Arts/Humanities	10055	Nada Hasan Saab	5000

### Query 8:

For the purpose of evaluating the performance of math's teachers, the math department requests to retrieve the name, ID, and phone number of each teacher who is english educated, has a bachelors of mathematics, and has at least two comments as feedback.

```
SELECT TEACHER.TID AS TEACHER_ID, TEACHER.FIRSTNAME AS FIRSTNAME,
TEACHER.LASTNAME AS LASTNAME, TEACHER_PHONE.PHONE AS PHONE_NUMBER,
COUNT(*) AS NUMBER_OF_COMMENTS

FROM TEACHER_COMMENT, TEACHER, TEACHER_DEGREE, TEACHER_LANGUAGE,
TEACHER_PHONE

WHERE TEACHER.TID=TEACHER_COMMENT.TEACHERID AND
TEACHER.TID=TEACHER_DEGREE.TEACHERID AND TEACHER_DEGREE.degree = 'Bachelor
of Mathematics' AND TEACHER_LANGUAGE.TEACHERID=TEACHER.TID AND
TEACHER_PHONE.TEACHERID=TEACHER.TID AND
TEACHER_LANGUAGE.language = 'English'

GROUP BY TEACHER.TID, TEACHER.FIRSTNAME, TEACHER.LASTNAME,
TEACHER_PHONE.PHONE
HAVING COUNT(*)>1
ORDER BY TEACHER.FIRSTNAME ASC;
```

#### **Output:**

TEACHER_ID	FIRSTNAME	LASTNAME	PHONE_NUMBER	NUMBER_OF_COMMENTS
10501	Hussein	Bakri	+96103245987	2

#### **Query 9:**

The school administration decides to help its employees who are in need and have limited professional achievements. Thus, it increases the salary of all employees who have a baccalaureate degree and working in the second branch (Beirut Branch) by 10 percent.

```
UPDATE EMPLOYEE
```

SET SALARY = 1.1\*SALARY

WHERE EID IN ( SELECT WORKS\_FOR.EMPLOYEEID  
FROM WORKS\_FOR, EMPLOYEE\_DEGREE  
WHERE BRANCH\_NB = 2 AND  
WORKS\_FOR.EMPLOYEEID=EMPLOYEE\_DEGREE.EMPLOYEEID AND  
EMPLOYEE\_DEGREE.DEGREE = 'Baccalaureate');

**Output:**

EID	ENAME	PROFESSION	EMAIL	GENDER	DOB	ADDRESS	SALARY	SS_ID	SUPERVISORID
10011	Charbel Joseph Younes	Security Guard	Charbel7676@hotmail.com	M	08/01/1980	Beirut/ Sin El-Fil	1650	yes	10018
10015	Mira Jamil Issa	cleaner	-	F	05/07/1972	Beirut/ Jal Al Dib	1100	yes	10014

**Query 10:**

As a social grant, increase the salary of each employee who has at least one child who is 18 years old or younger by 200\$.

UPDATE EMPLOYEE

SET SALARY = SALARY + 200

WHERE EID IN ( SELECT DEPENDENT.EMPLOYEEID

FROM DEPENDENT

WHERE (RELATIONSHIP = 'Son' OR RELATIONSHIP = 'Daughter') AND DOB >  
DATE'2004-01-01');

**Output:**

EID	ENAME	PROFESSION	EMAIL	GENDER	DOB	ADDRESS	SALARY	SS_ID	SUPERVISORID
10003	Mohammad Tarek Rahal	Academic Assistant	moe_rahal@hotmail.com	M	04/20/1980	Beirut/ Kraytem	2200	yes	10018
10018	Wael Jihad Saad	Principal	waelJ1_@hotmail.com	M	10/15/1962	Beirut/ Bir Hasan	3400	yes	-

**Normalization Up to The BCNF Normal Form:**

After creating all relations, we should ascertain that our database design is of high quality and try to improve it through normalization. Thus, we are going to normalize our database according to different normal forms up to the Boyce-Codd Normal Form. On each relation, the first, second, third, and Boyce-Codd normal forms will be applied respectively. Here is a general description to each of the normal forms mentioned.

### **First Normal Form:**

This normal form stresses on disallowing any composite attributes, multivalued attributes, and nested relations. Thus, all attributes should have single atomic values.

### **Second Normal Form:**

This normal form states that every non-prime attribute in a relation should be full functionally dependent on the primary key of that relation.

A prime attribute is an attribute that is member of the primary key.

Full functional dependency is defined such that if there is a Full FD  $X \rightarrow Y$ , the removal of any attribute from X means that the FD does not hold anymore.

### **Third Normal Form:**

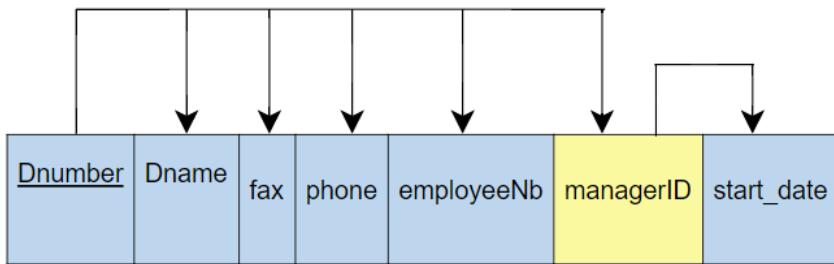
This normal form states that for every FD  $X \rightarrow Y$  existing in a relation R:

- 1) X should be a superkey of relation R  
OR
- 2) Y should be a prime attribute of relation R

### **Boyce-Codd Normal Form:**

This normal form demonstrates that for every FD  $X \rightarrow Y$  existing in a relation R, X should be a superkey of relation R. It is the same as the 3NF but with ignoring the second condition.

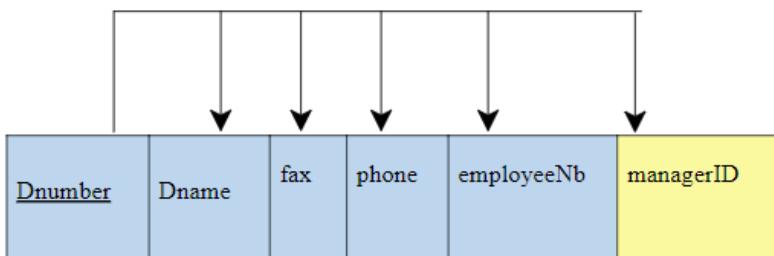
## **DEPARTMENT:**



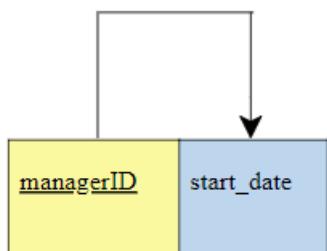
- A. The **DEPARTMENT** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **DEPARTMENT** relation does not satisfy all conditions of the 2NF because not every non-prime attribute in the relation is fully functional dependent on the primary key ("Dnumber"). "start\_date" attribute is dependent on the "managerID" attribute which is not a primary key.

Normalize:

#### DEPARTMENT



#### DEPARTMENT\_MANAGER



- C. Both **DEPARTMENT\_MANAGER** and **DEPARTMENT** relations satisfy all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("Dnumber", "managerID") or Y is a prime attribute.
- D. Both **DEPARTMENT\_MANAGER** and **DEPARTMENT** relations satisfy all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("Dnumber", "managerID").

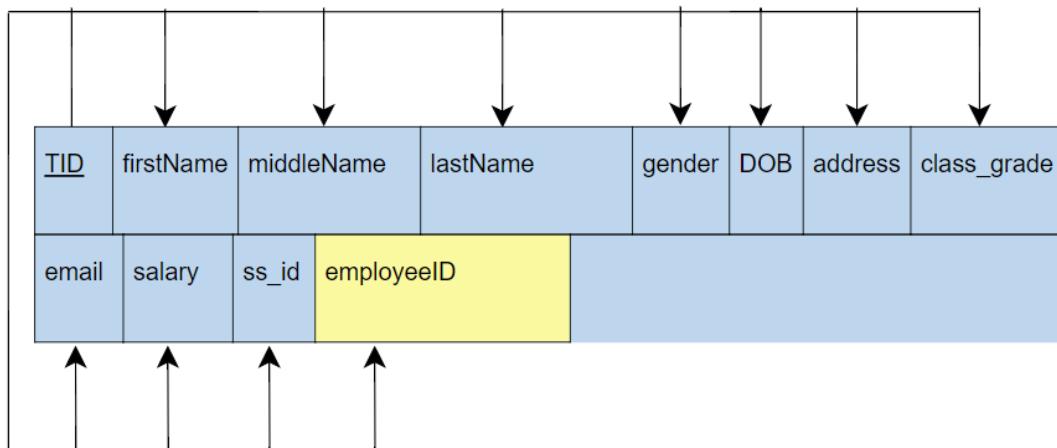
## STUDENT:

<u>SID</u>	firstName	middleName	lastName	gender	DOB	address	class_grade	
rank	language	supervisorID	branch_num					

The diagram illustrates the STUDENT relation. The primary key attribute, SID, is highlighted in blue. Non-primary attributes include firstName, middleName, lastName, gender, DOB, address, class\_grade, rank, language, supervisorID, and branch\_num. The supervisorID and branch\_num attributes are highlighted in yellow, indicating they are part of a composite key.

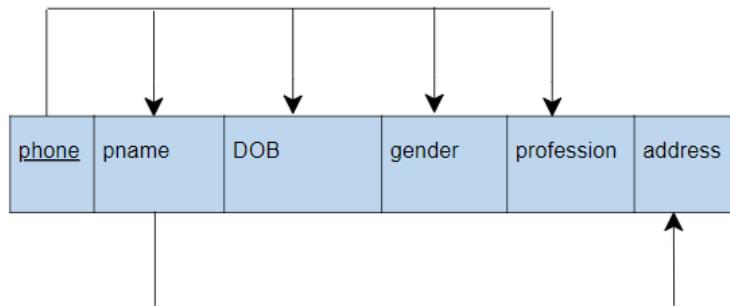
- A. The **STUDENT** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **STUDENT** relation satisfies all conditions of the 2NF since every non-primary attribute is fully functionally dependent on the primary key "SID".
- C. The **STUDENT** relation satisfies all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("SID") or Y is a prime attribute.
- D. The **STUDENT** relation satisfies all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("SID").

## TEACHER:



- A. The **TEACHER** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **TEACHER** relation satisfies all conditions of the 2NF since every non-primary attribute is fully functionally dependent on the primary key "TID".
- C. The **TEACHER** relation satisfies all conditions of the 3NF since it satisfies the 2NF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("TID") or Y is a prime attribute.
- D. The **TEACHER** relation satisfies all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("TID").

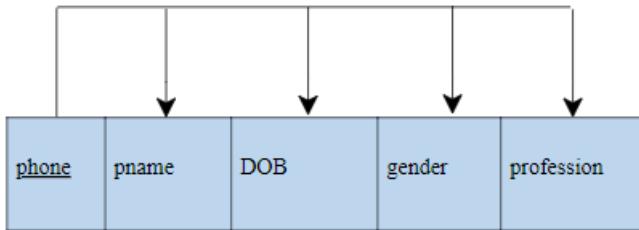
### PARENT:



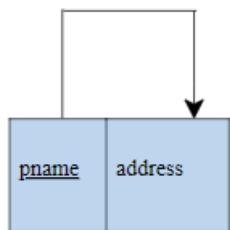
- A. The **PARENT** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **PARENT** relation does not satisfy all conditions of the 2NF because not every non-prime attribute in the relation is fully functional dependent on the primary key ("phone"). "address" attribute is dependent on the "pname" attribute which is not a primary key.

Normalize:

**PARENT**

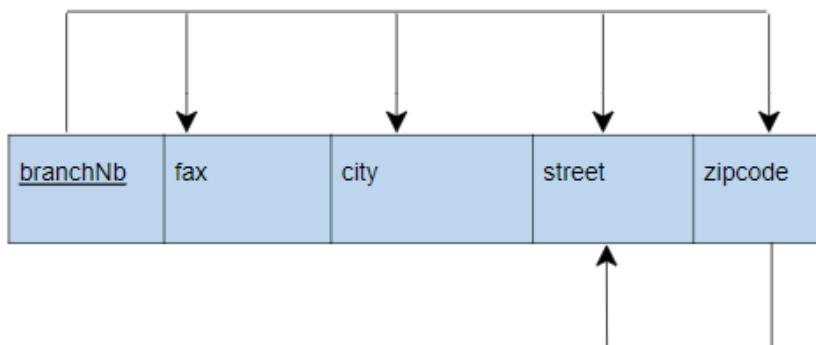


### PARENT\_ADDRESS



- C. Both **PARENT** and **PARENT\_ADDRESS** relations satisfy all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("phone", "pname") or Y is a prime attribute.
- D. Both **PARENT** and **PARENT\_ADDRESS** relations satisfy all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("phone", "pname").

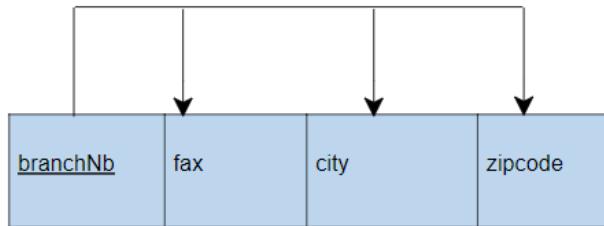
### BRANCH:



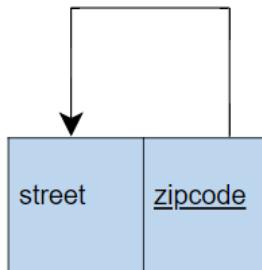
- A. The **BRANCH** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **BRANCH** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key "branchNb".
- C. The **BRANCH** relation does not satisfy all conditions of the 3NF since not for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("branchNb") or Y is a prime attribute. "zipcode" attribute is dependent on the "street" attribute, which is not a superkey, nor "zipcode" is a prime attribute.

Normalize:

## BRANCH

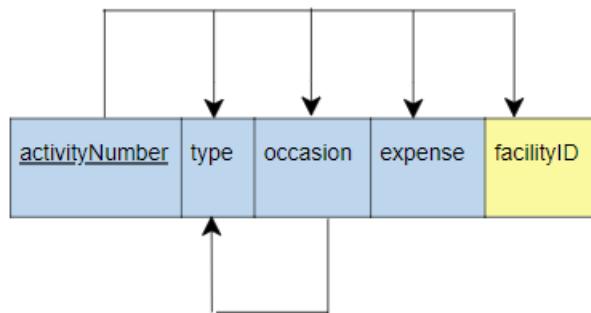


## BRANCH\_STREET



- D. Both **BRANCH** and **BRANCH\_STREET** relations satisfy all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("branchNb", "zipcode").

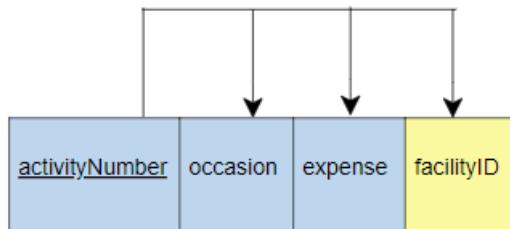
## ACTIVITY:



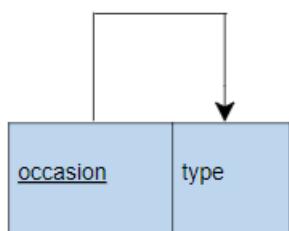
- A. The **ACTIVITY** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **ACTIVITY** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key "activityNumber".
- C. The **ACTIVITY** relation does not satisfy all conditions of the 3NF since not for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("activityNumber") or Y is a prime attribute. "occasion" attribute is dependent on the "type" attribute, which is not a superkey, nor "occasion" is a prime attribute.

Normalize:

### ACTIVITY

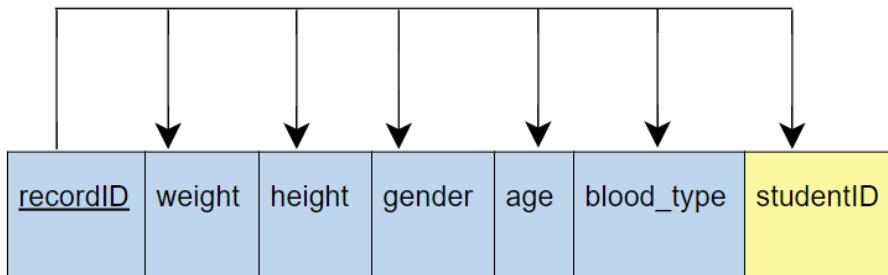


### ACTIVITY\_TYPE



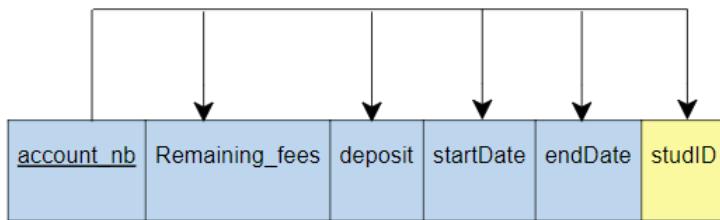
- D. Both **ACTIVITY** and **ACTIVITY\_TYPE** relations satisfy all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("activityNumber", "occasion").

### HEALTH RECORD:



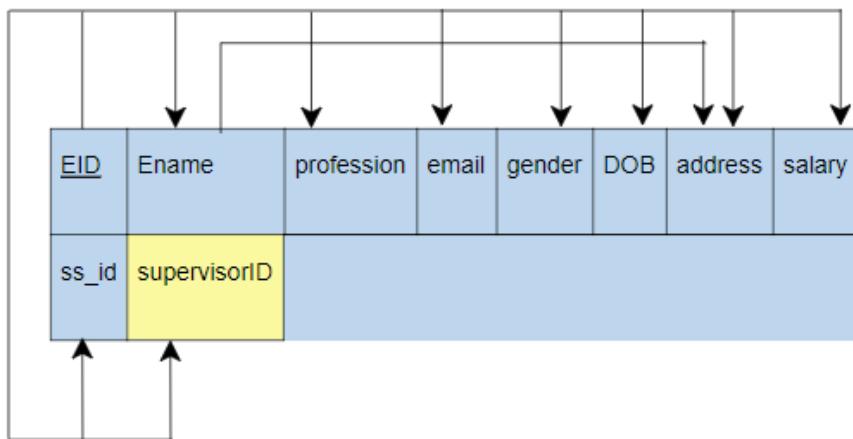
- A. The **HEALTH RECORD** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **HEALTH RECORD** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key "recordID".
- C. The **HEALTH RECORD** relation satisfies all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("recordID") or Y is a prime attribute.
- D. The **HEALTH RECORD** relation satisfies all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("recordID").

## ACCOUNT:



- A. The **ACCOUNT** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **ACCOUNT** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key "account\_nb".
- C. The **ACCOUNT** relation satisfies all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("account\_nb") or Y is a prime attribute.
- D. The **ACCOUNT** relation satisfies all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("account\_nb").

## EMPLOYEE:

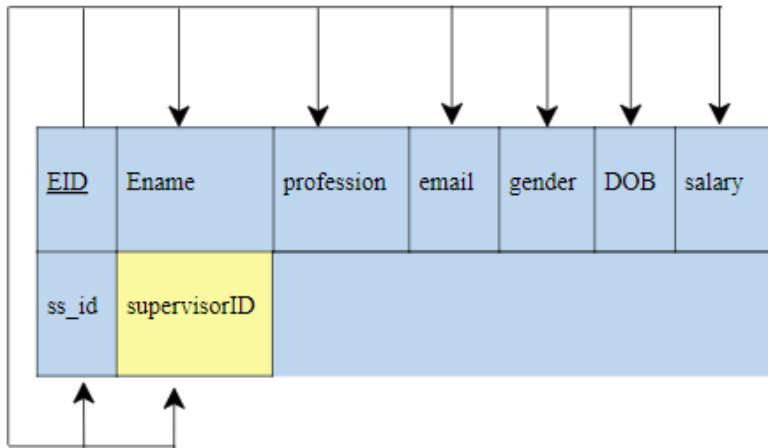


- A. The **EMPLOYEE** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **EMPLOYEE** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key "EID".

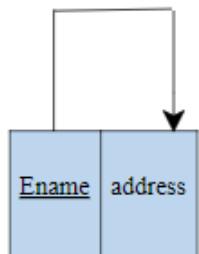
- C. The **EMPLOYEE** relation does not satisfy all conditions of the 3NF since not for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("EID") or Y is a prime attribute. "address" attribute is dependent on the "Ename" attribute, which is not a superkey, nor "address" is a prime attribute.

**Normalize:**

### EMPLOYEE

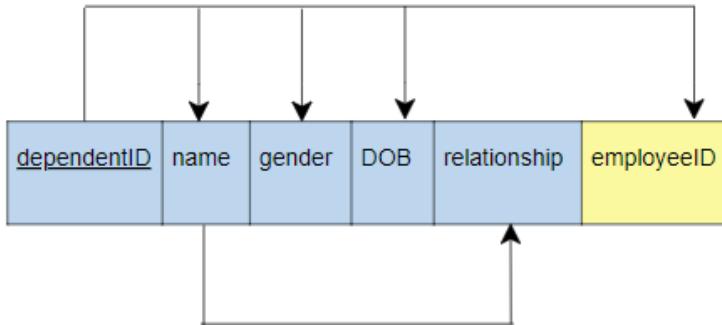


### EMPLOYEE\_ADDRESS



- D. Both **EMPLOYEE** and **EMPLOYEE\_ADDRESS** relations satisfy all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("EID", "Ename").

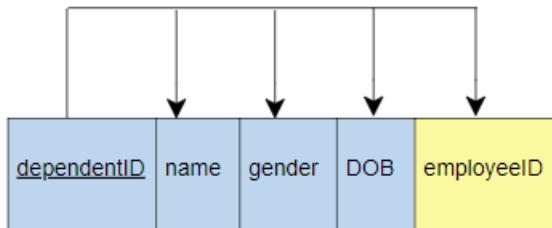
**DEPENDENT:**



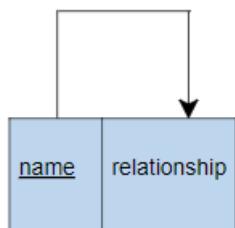
- A. The **DEPENDENT** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **DEPENDENT** relation does not satisfy all conditions of the 2NF because not every non-prime attribute in the relation is fully functional dependent on the primary key ("dependentID"). "relationship" attribute is dependent on the "name" attribute which is not a primary key.

Normalize:

### **DEPENDENT**



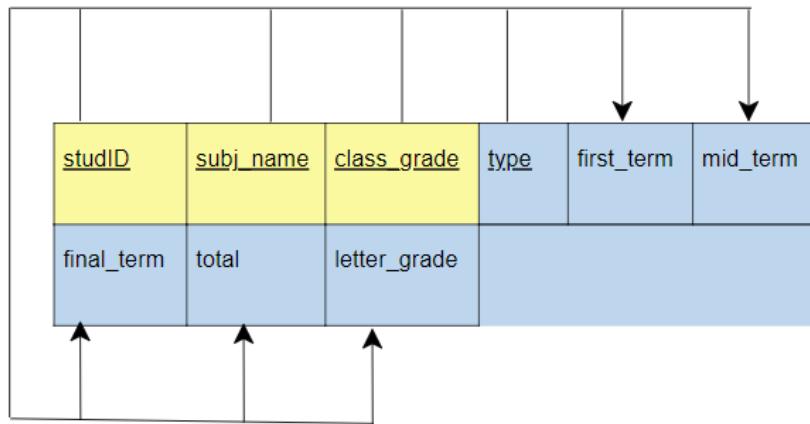
### **DEPENDENT\_RELATIONSHIP**



- C. Both **DEPENDENT** and **DEPENDENT\_RELATIONSHIP** relations satisfy all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("dependentID", "name") or Y is a prime attribute.

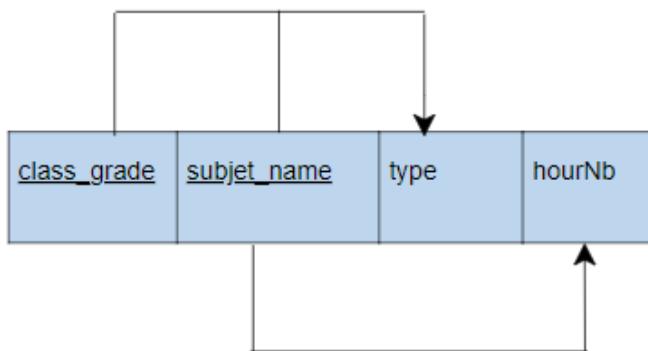
- D. Both **DEPENDENT** and **DEPENDENT\_RELATIONSHIP** relations satisfy all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("dependentID", "name").

### GRADE:



- A. The **GRADE** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **GRADE** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key ("studID", "class\_grade", "subject\_name", "type").
- C. The **GRADE** relation satisfies all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("studID", "class\_grade", "subject\_name", "type") or Y is a prime attribute.
- D. The **GRADE** relation satisfies all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("studID", "class\_grade", "subject\_name", "type").

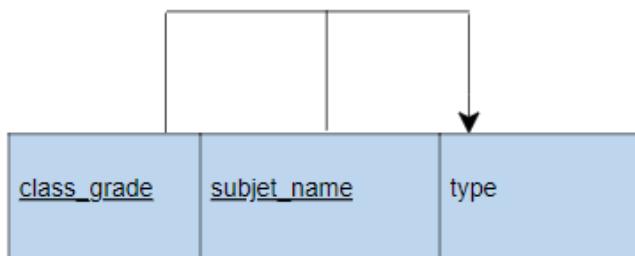
### SUBJECT:



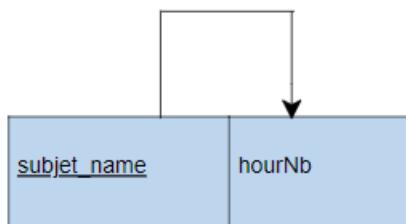
- A. The **SUBJECT** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **SUBJECT** relation does not satisfy all conditions of the 2NF because not every non-prime attribute in the relation is fully functional dependent on the primary key ("class\_grade", "subject\_name"). "hourNb" attribute is partially dependent on the primary key.

**Normalize:**

### **SUBJECT**

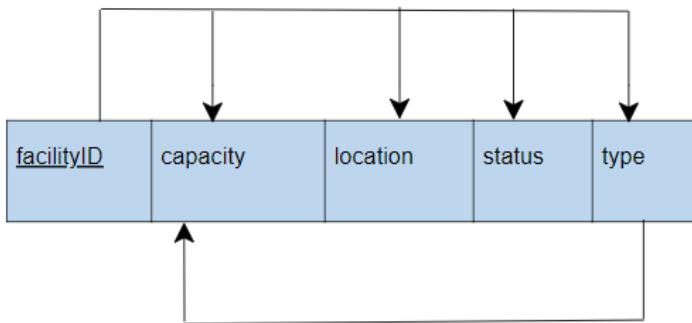


### **SUBJECT\_HOURS**



- C. Both **SUBJECT** and **SUBJECT\_HOURS** relations satisfy all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ( $\{"class\_grade", "subject\_name"\}$ , "subject\_name") or Y is a prime attribute.
- D. Both **SUBJECT** and **SUBJECT\_HOURS** relations satisfy all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ( $\{"class\_grade", "subject\_name"\}$ , "subject\_name").

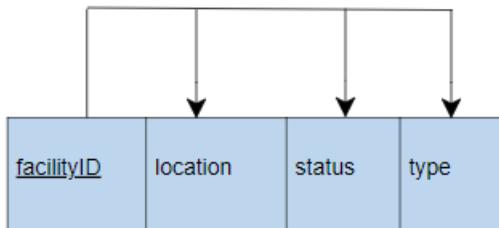
## FACILITY:



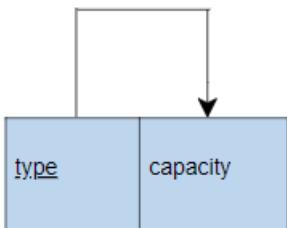
- A. The **FACILITY** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **FACILITY** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key "facilityID".
- C. The **FACILITY** relation does not satisfy all conditions of the 3NF since not for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("facilityID") or Y is a prime attribute. "capacity" attribute is dependent on the "type" attribute, which is not a superkey, nor "capacity" is a prime attribute.

**Normalize:**

## FACILITY



## FACILITY\_CAPACITY



- D. Both **FACILITY** and **FACILITY\_CAPACITY** relations satisfy all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in both relations, X is a superkey ("facilityID", "type").

## PARTICIPATES\_IN:

studID	activity_nb	participants_nb

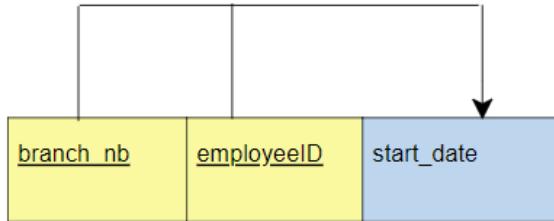
- A. The **PARTICIPATES\_IN** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **PARTICIPATES\_IN** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key ("studID", "activity\_nb").
- C. The **PARTICIPATES\_IN** relation satisfies all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("studID", "activity\_nb") or Y is a prime attribute.
- D. The **PARTICIPATES\_IN** relation satisfies all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("studID", "activity\_nb").

## TEACHES2:

teacherID	class_grade	subject_name	hour_nb

- A. The **TEACHES2** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **TEACHES2** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key ("teacherID", "class\_grade", "subject\_name").
- C. The **TEACHES2** relation satisfies all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("teacherID", "class\_grade", "subject\_name") or Y is a prime attribute.
- D. The **TEACHES2** relation satisfies all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("teacherID", "class\_grade", "subject\_name").

## WORKS\_FOR:



- A. The **WORKS\_FOR** relation satisfies all conditions of the 1NF because all of its attributes are single and atomic. This relation has neither multivalued attributes nor composite attributes.
- B. The **WORKS\_FOR** relation satisfies all conditions of the 2NF since every non-prime attribute is fully functionally dependent on the primary key ("branch\_nb", "employeeID").
- C. The **WORKS\_FOR** relation satisfies all conditions of the 3NF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("branch\_nb", "employeeID") or Y is a prime attribute.
- D. The **WORKS\_FOR** relation satisfies all conditions of the BCNF since for every FD  $X \rightarrow Y$  existing in the relation, X is a superkey ("branch\_nb", "employeeID").

## **Relations Schemas without non-primary attributes:**

### HAS\_A1:

<u>studID</u>	<u>subj_name</u>	<u>class_grade</u>	<u>type</u>
---------------	------------------	--------------------	-------------

### HAS\_A2:

<u>studID</u>	<u>parent_phone</u>
---------------	---------------------

### IS\_OF\_A:

<u>studID</u>	<u>subj_name</u>	<u>class_grade</u>	<u>type</u>
---------------	------------------	--------------------	-------------

ORGANIZES:

<u>employeeID</u>	<u>activity_nb</u>
-------------------	--------------------

IS\_TAUGHT:

<u>class_grade</u>	<u>subject_name</u>	<u>facilityID</u>
--------------------	---------------------	-------------------

TAKES:

<u>class_grade</u>	<u>subject_name</u>	<u>studID</u>
--------------------	---------------------	---------------

TEACHES1:

<u>teacherID</u>	<u>studID</u>
------------------	---------------

DEP\_LOCATION:

<u>Department_nb</u>	<u>location</u>
----------------------	-----------------

STUD\_WARNING:

<u>studID</u>	<u>warning</u>
---------------	----------------

## TEACHER\_LANGUAGE

<u>teacherID</u>	<u>language</u>
------------------	-----------------

## TEACHER\_COMMENT:

<u>teacherID</u>	<u>comment</u>
------------------	----------------

## TEACHER(CG):

<u>teacherID</u>	<u>class_grade</u>
------------------	--------------------

## TEACHER\_DEGREE:

<u>teacherID</u>	<u>degree</u>
------------------	---------------

## TEACHER\_PHONE:

<u>teacherID</u>	<u>phone</u>
------------------	--------------

## BRANCH\_PHONE:

<u>branch_nb</u>	<u>phone</u>
------------------	--------------

HEALTH\_DISEASE:

<u>recordID</u>	<u>disease</u>
-----------------	----------------

HEALTH\_ALLERGY:

<u>recordID</u>	<u>allergy</u>
-----------------	----------------

PAYMENT:

<u>account_nb</u>	<u>payment_menthod</u>
-------------------	------------------------

EMPLOYEE\_DEGREE:

<u>employeeID</u>	<u>degree</u>
-------------------	---------------

EMPLOYEE\_PHONE:

<u>employeeID</u>	<u>phone</u>
-------------------	--------------

SUBJECT\_LANGUAGE

<u>subject_name</u>	<u>class_grade</u>	<u>language</u>
---------------------	--------------------	-----------------

ORGANIZER\_NAME:

<u>employeeID</u>	<u>activity_nb</u>	<u>organizer_name</u>
-------------------	--------------------	-----------------------

**ACTIVITY\_GUESTS:**

<u>activityNb</u>	<u>guest_name</u>
-------------------	-------------------