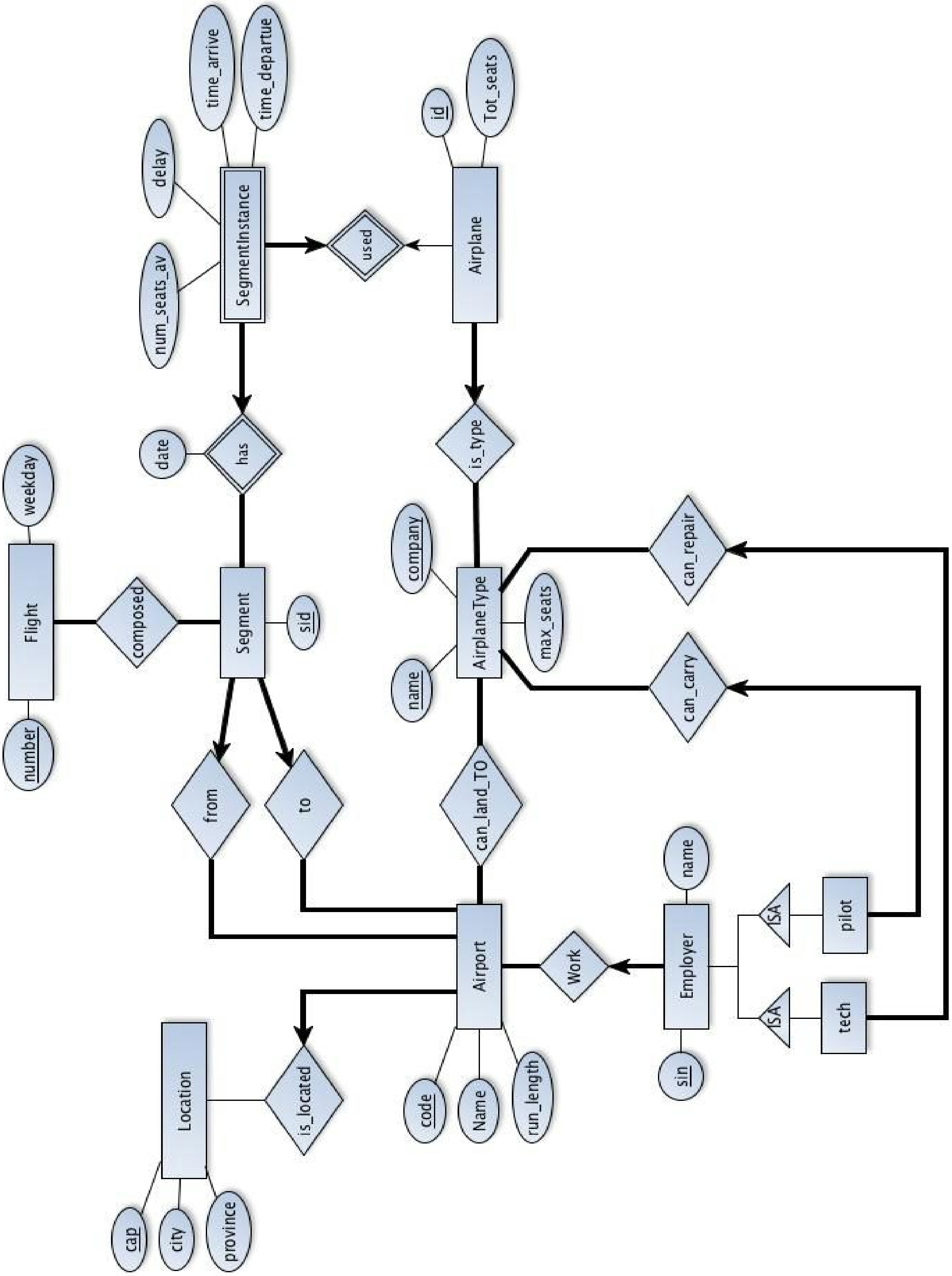
PART A

ER


time_arrive
time_departue
delay
num_seats_av
SegmentInstance
used
Airplane
id
Tot_seats
has
date
is_type
weekday
Flight
number
composed
Segment
sid
company
AirplaneType
name
max_seats
can_repair
can_carry
from
to
can_land_TO
Airport
Work
Employer
name
sin
ISA
pilot
ISA
tech
is_located
Location
cap
city
province
code
Name
run_length

SCHEMA

Ariport (code, name: unique not null, run_length: not null, id_location: not null)
id_location is a foreign key to table Location

CanLandOrTO (airport_code, airt_name, airt_company)
airport_code is a foreign key to table Ariport
airt_name is a foreign key to table AirplaneType
airt_company is a foreign key to table AirplanType

Location (cap, city: not null, province: not null)

AirplaneType (name, company, max_seats: not null)

Airplane (id, tot_seats: not null, name, company)
name is a foreign key to table AriplaneType
company is a foreign key to table AirplaneType

Employer (sin, name: not null, airport_code, type: not null, airt_name, airt_company)
airport_code is a foreign key to table Airport
airt_name is a foreign key to table AirplaneType
airt_company is a foreign key to table AirplaneType

Segment (id, airport_from, airport_to)
airport_from is a foreign key to table Airport
airport_to is a foreign key to table Airport

Composed (segment_id, flight_number)
segment_id is a foreign key to table Segment
flight_number is a foreign key to table Flight

Flight (number, weekday: not null)

SegmentInstance (segment_id, airplane_id, time_departure, time_arrive, date, num_seats_av: not null, delay)
segment_id is a foreign key to table Segment
airplane_id is a foreign key to table Airplane

SQL

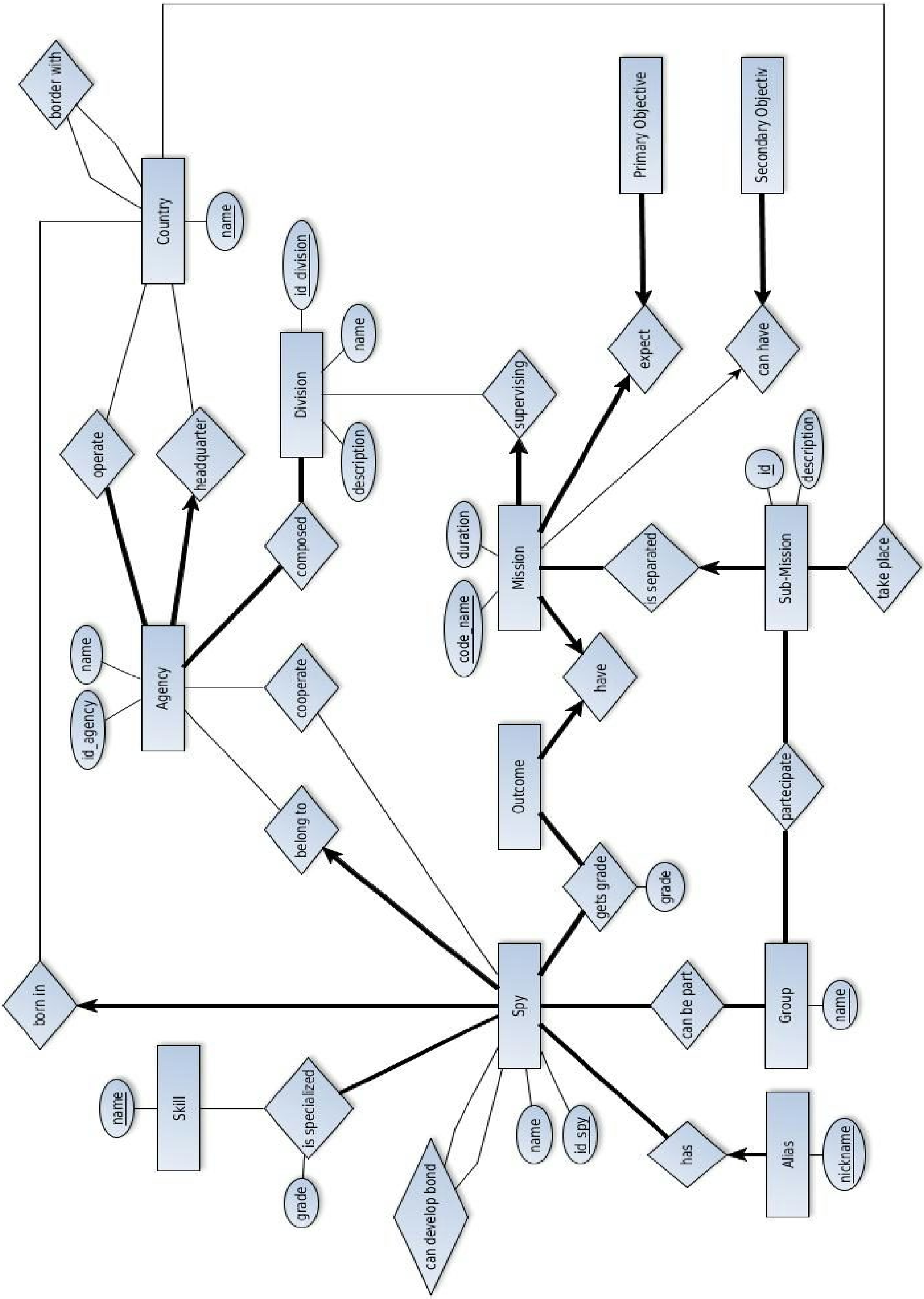
```
CREATE TABLE Location (
    cap INTEGER,
    city CHAR(30) NOT NULL,
    province CHAR(2) NOT NULL,
    PRIMARY KEY( cap )
)

CREATE TABLE Airport (
    code INTEGER,
    name CHAR(20) UNIQUE NOT NULL,
    run_length REAL NOT NULL,
    id_location INTEGER NOT NULL,
    PRIMARY KEY( code ),
    FOREIGN KEY( id_location ) REFERENCES Location.id
        ON DELETE NO ACTION
        ON UPDATE CASCADE
)

CREATE TABLE CanLandOrTO (
    ariport_code INTEGER,
    arit_name CHAR(20),
    airt_company CHAR(20),
    PRIMARY KEY( ariport_code, airt_name, airt_company ),
    FOREIGN KEY( ariport_code ) REFERENCES Ariport.code
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY( airt_name ) REFERENCES AriplaneType.name
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY( arit_company ) REFERENCES AirplaneType.company
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

```
CREATE TABLE AriplaneType (
        name CHAR( 20 ),
        company CHAR( 20 ),
        max_seats NOT NULL,
        PRIMARY KEY( name, company )
)

CREATE TABLE Airplane (
        id INTEGER,
        tot_seats NOT NULL,
        name CHAR( 20 ),
        company CHAR( 20 ),
        PRIMARY KEY( id ),
        FOREIGN KEY( name ) REFERENCES AirplaneType.name
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( company ) REFERENCES AirplaneType.company
                ON DELETE CASCADE
                ON UPDATE CASCADE
)

CREATE TABLE Employer (
        sin INTEGER,
        name CHAR( 20 ),
        airport_code INTEGER,
        type INTEGER,
        airt_name CHAR( 20 ),
        airt_company CHAR( 20 ),
        PRIMARY KEY( sin ),
        FOREIGN KEY( airport_code ) REFERENCES Airport.code
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( airt_name ) REFERENCES AirplaneType.name
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( airt_company ) REFERENCES AirplaneType.company
                ON DELETE CASCADE
                ON UPDATE CASCADE)
```

```
CREATE TABLE Flight (
        number INTEGER,
        weekday INTEGER NOT NULL,
        PRIMARY KEY( number )
)

CREATE TABLE Composed (
        segment_id INTEGER,
        flight_number INTEGER,
        PRIMARY KEY( segment_id, flight_number ),
        FOREIGN KEY( segment_id ) REFERENCES Segment.id
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( flight_number ) REFERENCES Flight.number
                ON DELETE CASCADE
                ON UPDATE CASCADE,
)

CREATE TABLE Segment (
        id INTEGER,
        airport_from INTEGER,
        airport_to INTEGER,
        PRIMARY KEY( id )
)

CREATE TABLE SegmentInstance (
        segment_id INTEGER,
        ariplane_id INTEGER,
        time_departure TIMESTAMP,
        time_arrive TIMESTAMP,
        date DATETIME NOT NULL,
        num_seats_av INTEGER NOT NULL,
        delay INTEGER,
        PRIMARY KEY( segment_id, ariplane_id ),
        FOREIGN KEY( segment_id ) REFERENCES Segment.id
                ON DELETE NO ACTION
                ON UPDATE CASCADE,
        FOREIGN KEY( airplane_id ) REFERENCES Ariplane.id
                ON DELETE CASCADE ON UPDATE CASCADE )
```

PART B

ER


border with
Country
name
id division
name
Division
description
supervising
Primary Objective
Secondary Objectiv
expect
can have
operate
headquarter
composed
duration
code name
Mission
is separated
Sub-Mission
take place
id
description
name
id_agency
Agency
cooperate
have
Outcome
partecipate
belong to
gets grade
grade
born in
Spy
can be part
Group
name
Skill
name
is specialized
grade
can develop bond
name
id spy
has
Alias
nickname

SCHEMA

Assumptions:

- A Spy belongs to at least one agency
- A Spy can cooperate with zero or more agency
- An agency can cooperate with zero or more spies
- An agency can recruit zero or more spies
- A group can be composed by one spy

Country (name)

BorderWith (first_country, second_country)
first_country is a foreign key to table Contry
second_country is a foreign key to table Country

Operate (country_name, id_agency)
country_name is a foreign key to table Country
id_agency is a foreign key to table Agency

Agency (id, name: not null, headquarter_location)
headquarter_location is a foreign key to table Country

Cooperated_with (id_spy, id_agency)
id_spy is a foreign key to table Spy
id_agency is a foreign key to table Agency

Spy (id, name: not null, born, belong_to)
born is a foreign key to table Country
belog_to is a foreign key to table Agency

BondWith (first_spy, second_spy)
first_spy is a foreign key to table Spy
second_spy is a foreign key to table Spy

Alias(nickname, id_spy)
id_spy is a foreign key to table Spy

Skill (name)

Specialized_in(id_spy, name_skill, grade)
id_spy is a foreign key to table Spy
name_skill is a foreign key to table Skill

Group (name)

Part_of (id_spy, name_group)
id_spy is a foreign key to table Spy
name_group is a foreign key to table Group

GradeFrom (id_spy, code_name_mission, grade)
id_spy is a foreign key to table Spy
code_name_mission is a foreign key to table Mission

Mission (code_name, duration: not null, outcome, primary_object:not null, secondary_object, supervising_by)
supervising_by is a foreign key to table Division

SubMission (id, description:not null, code_name_mission:not null)
code_name_mission is a foreign key to table Mission

Partecipate (group_name, id_submission)
group_name is a foreign key to table Group
id_submission is a foreign key to table SubMission

TakePlace (name_country, id_submission)
name_country is a foreign key to table Country
id_submission is a foreign key to table SubMission

Division (id, name: not null, description: not null)

ComposedBy (id_agency, id_division)
id_agency is a foreign key to table Agency
id_division is a foreign key to table Division

SQL

```
TABLE Country (
    name CHAR( 30 ),
    PRIMARY KEY( name )
)

CREATE TABLE BorderWith (
    first_country CHAR( 30 ),
    second_country CHAR( 30 ),
    PRIMARY KEY( first_country, second_country ),
    FOREIGN KEY( first_country ) REFERENCES Country.name
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY( second_country ) REFERENCES Country.name
        ON DELETE CASCADE
        ON UPDATE CASCADE
)

CREATE TABLE Operate (
    country_name CHAR( 30 ),
    id_agency INTEGER,
    PRIMARY KEY( country_name, id_agency ),
    FOREIGN KEY( country_name ) REFERENCES Country.name
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY( id_agency ) REFERENCES Agency.id
        ON DELETE CASCADE
        ON UPDATE CASCADE
)

CREATE TABLE Agency (
    id INEGER,
    name CHAR( 30 ) NOT NULL,
    headquarter_location,
    PRIMARY KEY( id ),
    FOREIGN KEY( headquarter_location ) REFERENCES Country.name
        ON DELETE SET DEFAULT
        ON UPDATE CASCADE )
```

```
CREATE TABLE CooperatedWith (
        id_spy INTEGER,
        id_agency INTEGER,
        PRIMARY KEY( id_spy, id_agency )
        FOREIGN KEY( id_spy ) REFERENCES Spy.id
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( id_agency ) REFERENCES Agency.id
                ON DELETE CASCADE
                ON UPDATE CASCADE
)

CREATE TABLE Spy (
        id INTEGER,
        name CHAR( 30 ) NOT NULL,
        born CHAR( 30 ),
        belong_to INTEGER,
        PRIMARY KEY( id ),
        FOREIGN KEY( born ) REFERENCES Country.name
                ON DELETE SET DEFAULT
                ON UPDATE CASCADE,
        FOREIGN KEY( belong_to ) REFERENCES Agency.id
                ON DELETE SET DEFAULT
                ON UPDATE CASCADE
)

CREATE TABLE BondWith (
        first_spy INTEGER,
        second_spy INTEGER,
        PRIMARY KEY( first_spy, second_spy ),
        FOREIGN KEY( first_spy ) REFERENCES Spy.id
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( second_spy ) REFERENCES Spy.id
                ON DELETE CASCADE
                ON UPDATE CASCADE
)
```

```
CREATE TABLE Alias (
    nickname CHAR( 30 ),
    id_spy INTEGER,
    PRIMARY KEY( nickname ),
    FOREIGN KEY( id_spy ) REFERENCES Spy.id
        ON DELETE CASCADE
        ON UPDATE CASCADE
)

CREATE TABLE Skill (
    name CHAR( 30 ),
    PRIMARY KEY ( name )
)

CREATE TABLE SpecializedIn(
    id_spy INTEGER,
    name_skill CHAR( 30 ),
    grade INTEGER,
    PRIMARY KEY( id_spy, name_skill ),
    FOREIGN KEY( id_spy ) REFERENCES Spy.id
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY( name_skill ) REFERENCES Skill.name
        ON DELETE CASCADE
        ON UPDATE CASCADE
)

CREATE TABLE Group (
    name CHAR( 30 ),
    PRIMARY KEY( name )
)
```

```
CREATE TABLE Part_of (
        id_spy INTEGER,
        name_group CHAR( 30 ),
        PRIMARY KEY( id_spy, name_group ),
        FOREIGN KEY( id_spy ) REFERENCES Spy.id
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( name_group ) REFERENCES Group.name
                ON DELETE CASCADE
                ON UPDATE CASCADE
)

CREATE TABLE GradeFrom (
        id_spy INTEGER,
        code_name_mission CHAR( 30 ),
        grade INTEGER,
        PRIMARY KEY( id_spy, code_name_mission ),
        FOREIGN KEY( id_spy ) REFERENCES Spy.id
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( code_name_mission ) REFERENCES Mission.code_name
                ON DELETE CASCADE
                ON UPDATE CASCADE
)

CREATE TABLE Mission (
        code_name CHAR( 30 ),
        duration: INTEGER  NOT NULL,
        outcome CHAR( 30 ),
        primary_object CHAR( 30 ) NOT NULL,
        secondary_object CHAR( 30 ),
        supervising_by INTEGER,
        PRIMARY KEY( code_name ),
        FOREIGN KEY( supervising_by ) REFERENCES Division.id
                ON DELETE SET DEFAULT
                ON UPDATE CASCADE
)
```

```
CREATE TABLE SubMission (
        id INTEGER,
        description CHAR( 30 ) NOT NULL,
        code_name_mission CHAR( 30 ) NOT NULL,
        PRIMARY KEY( id ),
        FOREIGN KEY( code_name_mission ) REFERENCES Mission.code_name
                ON DELETE CASCADE
                ON UPDATE CASCADE
)

CREATE TABLE Partecipate (
        group_name CHAR( 30 ),
        id_submission  INTEGER,
        PRIMARY KEY( group_name, id_submission ),
        FOREIGN KEY( group_name ) REFERENCES Group.name
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( id_submission ) REFERENCES SubMission.id
                ON DELETE CASCADE
                ON UPDATE CASCADE
)

CREATE TABLE TakePlace (
        name_country  CHAR( 30 ),
        id_submission  INTEGER,
        PRIMARY KEY( name_country, id_submission ),
        FOREIGN KEY( name_country ) REFERENCES Country.name
                ON DELETE SET DEFAULT
                ON UPDATE CASCADE,
        FOREIGN KEY( id_submission ) REFERENCES SubMission.id
                ON DELETE CASCADE
                ON UPDATE CASCADE
)
```

```
CREATE TABLE Division (
        id INTEGER,
        name CHAR( 30 ) NOT NULL,
        description CHAR( 30 ) NOT NULL,
        PRIMARY KEY( id ),
)

CREATE TABLE ComposedBy (
        id_agency INTEGER,
        id_division INTEGER,
        PRIMARY KEY( id_agency, id_division ),
        FOREIGN KEY( id_agency ) REFERENCES Agency.id
                ON DELETE CASCADE
                ON UPDATE CASCADE,
        FOREIGN KEY( id_division ) REFERENCES Division.id
                ON DELETE CASCADE
                ON UPDATE CASCADE
)
```