use sosyalOzgecmis;

Create Table ORGANIZATION

(

Organization\_id int NOT NULL,

Organization\_name varchar(45),

CONSTRAINT PK\_ORGANIZATION PRIMARY KEY (Organization\_id)

);

Create Table PRIVACY

(

Privacy\_id int NOT NULL,

Privacy\_name varchar(20),

CONSTRAINT PK\_Privacy PRIMARY KEY(Privacy\_id)

);

Create Table MEMBER

(

Member\_id int NOT NULL,

/\*Adres\_id int NOT NULL, BUNU SILMEYI UNUTMA!!!\*/

Current\_organization\_id int,

Token int NOT NULL,

Member\_name varchar(45) NOT NULL,

Picture varchar(255),

Active boolean,

Onlıne boolean,

Date\_joined date NOT NULL,

Date\_of\_birthday date check (DATEDİFF(now()-Date\_of\_birthday>18\*365.25) ),/\*bunu ekledik sonradan\*/

Email\_adres varchar(100) NOT NULL,

Email\_password varchar(45) NOT NULL,

First\_name varchar(45) NOT NULL,

Middle\_name varchar(45),

Last\_name varchar(45) NOT NULL,/\*Dondurma Tarihi Eklee\*/

Gender char,

Dondurma\_tarihi date,

CONSTRAINT PK\_MEMBER PRIMARY KEY (Member\_id),

CONSTRAINT FK\_Current\_organization\_id FOREIGN KEY (Current\_organization\_id)

REFERENCES ORGANIZATION(Organization\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

UNIQUE (Token)

);

Create Table ESKI\_KAYITLAR

(

eski\_member\_id int

);

Create Table MESSAGE

(

Message\_id int NOT NULL,

Message varchar(500),

Created\_at date NOT NULL,

Is\_read boolean,

Is\_spam boolean,

Kime int NOT NULL,

Is\_reply boolean,

Member\_id int NOT NULL,

CONSTRAINT PK\_MESSAGE PRIMARY KEY (Message\_id),

CONSTRAINT FK\_Kime FOREIGN KEY (Kime)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Member\_id FOREIGN KEY (Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table GRUP

(

Group\_id int NOT NULL,

Created\_by\_member\_id int,

Group\_name varchar(100) NOT NULL,

Group\_Description varchar(100),

Group\_date\_started date NOT NULL,

Group\_date\_ended date,

Group\_date\_last\_activity datetime,/\*zaman degeri de var\*/

Uygun\_kontenjan int check (Uygun\_kontenjan>0),/\*trigger icin\*/

Privacy\_id int,

CONSTRAINT PK\_GRUP PRIMARY KEY(Group\_id),

CONSTRAINT FK\_Created\_by\_member\_id FOREIGN KEY(Created\_by\_member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Grup FOREIGN KEY(Privacy\_id)

REFERENCES PRIVACY(Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create Table MEMBER\_GRUP

(

Member\_id int NOT NULL,

Group\_id int NOT NULL,

Date\_joined date NOT NULL,

Date\_left date,

CONSTRAINT PK\_MEMBER\_GRUP PRIMARY KEY (Member\_id,Group\_id),

CONSTRAINT FK\_Member\_id\_GRUP FOREIGN KEY(Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Group\_id\_GRUP FOREIGN KEY(Group\_id)

REFERENCES GRUP(Group\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table SIRKET

(

Sirket\_id int NOT NULL,

Sİrket\_created\_by\_member\_id int,

Sirket\_name varchar(20) NOT NULL,

Sirket\_Description varchar(100),

Sirket\_date\_started date NOT NULL,

Sirket\_date\_ended date,

Sirket\_date\_last\_activity datetime,

Privacy\_id int,

CONSTRAINT PK\_SIRKET PRIMARY KEY(Sirket\_id),

CONSTRAINT FK\_Sirket\_Created\_by\_member\_id FOREIGN KEY(Sirket\_created\_by\_member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Sirket FOREIGN KEY(Privacy\_id)

REFERENCES PRIVACY(Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create Table MEMBER\_SIRKET

(

Member\_id int NOT NULL,

Sirket\_id int NOT NULL,

Date\_joined date NOT NULL,

Date\_left date,

CONSTRAINT PK\_MEMBER\_SIRKET PRIMARY KEY (Member\_id,Sirket\_id),

CONSTRAINT FK\_Member\_id\_MEMBER\_SIRKET FOREIGN KEY(Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Sirket\_id FOREIGN KEY(Sirket\_id)

REFERENCES SIRKET(Sirket\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table UNIVERSITE

(

Universite\_id int NOT NULL,

Created\_by\_member\_id int,

Universite\_name varchar(20) NOT NULL,

Universite\_Description varchar(100),

Universite\_date\_started date NOT NULL,

Universite\_date\_ended date,

Universite\_date\_last\_activity datetime,

Privacy\_id int,

CONSTRAINT PK\_UNIVERSITE PRIMARY KEY(Universite\_id),

CONSTRAINT FK\_Universite\_Created\_by\_member\_id FOREIGN KEY(Created\_by\_member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Universite FOREIGN KEY(Privacy\_id)

REFERENCES PRIVACY(Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create Table MEMBER\_UNIVERSITE

(

Member\_id int NOT NULL,

Universite\_id int NOT NULL,

Date\_joined date NOT NULL,

Date\_left date,

CONSTRAINT PK\_MEMBER\_UNIVERSITE PRIMARY KEY (Member\_id,Universite\_id),

CONSTRAINT FK\_Member\_id\_MEMBER\_UNIVERSITE FOREIGN KEY(Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Universite\_id FOREIGN KEY(Universite\_id)

REFERENCES UNIVERSITE(Universite\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table NOTIFICATION

(

Notification\_id int NOT NULL,/\*bildirimin kimden gelecegini dusunmedik.\*/

Msg varchar(50),

Tip int,

Created\_at date NOT NULL,

Member\_id int NOT NULL,

Privacy\_id int,

CONSTRAINT PK\_NOTIFICATION PRIMARY KEY (Notification\_id),

CONSTRAINT FK\_Notification\_Member\_id FOREIGN KEY (Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Notification FOREIGN KEY (Privacy\_id)

REFERENCES PRIVACY(Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create Table BOOKMARK\_CATEGORY

(

Bookmark\_category\_id int NOT NULL,

Adi varchar(20) NOT NULL,

CONSTRAINT PK\_BOOKMARK\_CATEGORY PRIMARY KEY (Bookmark\_category\_id)

);

Create Table BOOKMARK\_SUB\_CATEGORY

(

Bookmark\_sub\_category\_id int NOT NULL,

Adi varchar(20) NOT NULL,

Bookmark\_category\_id int NOT NULL,

CONSTRAINT PK\_BOOKMARK\_SUB\_CATEGORY PRIMARY KEY (Bookmark\_sub\_category\_id),

CONSTRAINT FK\_Bookmark\_category\_id FOREIGN KEY (Bookmark\_category\_id)

REFERENCES BOOKMARK\_CATEGORY (Bookmark\_category\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table BOOKMARK

(

Bookmark\_id int NOT NULL,

Url varchar(255),

Rating int ,

Created\_at date NOT NULL,

Bookmark\_category\_id int ,

Bookmark\_sub\_category\_id int,

Privacy\_id int,

CONSTRAINT PK\_BOOKMARK PRIMARY KEY (Bookmark\_id),

CONSTRAINT FK\_Bookmark\_category\_id\_BOOKMARK FOREIGN KEY (Bookmark\_category\_id)

REFERENCES BOOKMARK\_CATEGORY (Bookmark\_category\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

/\*null oldugunu dusunduk\*/

CONSTRAINT FK\_Bookmark\_sub\_category\_id\_BOOKMARK FOREIGN KEY (Bookmark\_sub\_category\_id)

REFERENCES BOOKMARK\_SUB\_CATEGORY (Bookmark\_sub\_category\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Bookmark FOREIGN KEY (Privacy\_id)

REFERENCES PRIVACY (Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create Table BOOKMARK\_INFO

(

Bookmark\_info\_id int NOT NULL,

Bookmark\_id int NOT NULL,

Member\_id int NOT NULL,

Favourite boolean,

Clicks int,

Privacy\_id int,

CONSTRAINT PK\_BOOKMARK\_INFO PRIMARY KEY (Bookmark\_info\_id),

CONSTRAINT FK\_1\_BOOKMARK\_INFO FOREIGN KEY (Bookmark\_id)

REFERENCES BOOKMARK (Bookmark\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_2\_BOOKMARK\_INFO FOREIGN KEY (Member\_id)

REFERENCES MEMBER (Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Bookmark\_info FOREIGN KEY (Privacy\_id)

REFERENCES PRIVACY (Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create table REF\_PROFILE\_SECTION

(

Profile\_section\_code int not null,/\*burda favori film,kitap,müzik vb. profili olusturan alt basliklar yazılacak.\*/

Profile\_section\_description varchar(20),

CONSTRAINT PK\_REF\_PROFILE\_SECTION primary key(Profile\_section\_code)

);

Create table MEMBER\_PROFILES/\*burada bir profilin kime ait oldugu tutuluyo sadece\*/

(Profile\_id int not null,

Member\_id int not null,

Date\_created date not null,

Date\_last\_updated date not null,

Goruntulenme\_sayisi int ,

Privacy\_id int,

CONSTRAINT PK\_MEMBER\_PROFILES primary key(Profile\_id),

CONSTRAINT FK\_MEMBER\_PROFILES foreign key (Member\_id) REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id foreign key (Privacy\_id) REFERENCES PRIVACY(Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

CREATE TABLE PROFILE\_SECTION (

Profile\_section\_id INT NOT NULL,

Profile\_section\_code INT NOT NULL,

Profile\_id INT NOT NULL,

Date\_created date NOT NULL,

Date\_updated date NOT NULL,

Profile\_section\_text VARCHAR(200),

CONSTRAINT PK\_PROFILE\_SECTION PRIMARY KEY (Profile\_section\_id,Profile\_section\_code,Profile\_id),

CONSTRAINT FK\_Profile\_section\_code\_PROFILE\_SECTION FOREIGN KEY (Profile\_section\_code)

REFERENCES REF\_PROFILE\_SECTION (Profile\_section\_code)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Profile\_id\_PROFILE\_SECTION FOREIGN KEY (Profile\_id)

REFERENCES MEMBER\_PROFILES (Profile\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table POPULER\_MEMBER

(

Email\_address varchar(45)

);

Create Table COUNTRY

(

Country\_id int NOT NULL,

Country varchar(45),

CONSTRAINT PK\_COUNTRY PRIMARY KEY(Country\_id)

);

Create Table CITY

(

City\_id int NOT NULL,

Country\_id int NOT NULL,

City varchar(45),

CONSTRAINT PK\_CITY PRIMARY KEY (City\_id),

CONSTRAINT FK\_Country\_id FOREIGN KEY (Country\_id)

REFERENCES COUNTRY(Country\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table ADDRESS

(

Address\_id int NOT NULL,/\*Address sahasına yazacagı sehir ile city\_id control edilmeli\*/

Address varchar(45),

City\_id int NOT NULL,

Profile\_id int NOT NULL,

Privacy\_id int,

CONSTRAINT PK\_ADDRESS PRIMARY KEY (Address\_id),

CONSTRAINT FK\_City\_id FOREIGN KEY(City\_id)

REFERENCES CITY(City\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Profile\_id FOREIGN KEY (Profile\_id)/\*tam tersi olmsı daha mı mantıklı??\*/

REFERENCES MEMBER\_PROFILES(Profile\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Address FOREIGN KEY (Privacy\_id)/\*tam tersi olmsı daha mı mantıklı??\*/

REFERENCES PRIVACY(Privacy\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table LANG

(

Language\_id int NOT NULL,

Lang varchar(45),

Member\_id int NOT NULL,

CONSTRAINT PK\_LANG PRIMARY KEY(Language\_id),

CONSTRAINT FK\_Member\_id\_LANG FOREIGN KEY(Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table NICK\_NAME

(

Nick\_name\_id int NOT NULL,

Nickname varchar(45),

Member\_id int NOT NULL,

Privacy\_id int,

CONSTRAINT PK\_NICK\_NAME PRIMARY KEY(Nick\_name\_id),

CONSTRAINT FK\_Member\_id\_NICK\_NAME FOREIGN KEY(Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Nick FOREIGN KEY(Privacy\_id)

REFERENCES PRIVACY(Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create Table REF\_CV\_SECTIONS

(

Cv\_section\_code int NOT NULL,

Cv\_section\_descr varchar(45),

CONSTRAINT PK\_REF\_CV\_SECTIONS PRIMARY KEY(Cv\_section\_code)

);

Create Table CVS

(

Cv\_id int NOT NULL,

Member\_id int NOT NULL,

Date\_created date NOT null,

Date\_updated date NOT NULL,

Privacy\_id int,

CONSTRAINT PK\_CVS PRIMARY KEY(Cv\_id),

CONSTRAINT FK\_Member\_id\_CVS FOREIGN KEY(Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Cv FOREIGN KEY(Privacy\_id)

REFERENCES PRIVACY(Privacy\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table CV\_SECTIONS

(

Cv\_section\_id int NOT NULL,/\*aynı mantıkla profili 3 parcali pk dan olusturduk\*/

Cv\_id int NOT NULL,

Cv\_section\_code int NOT NULL,

Date\_created date NOT NULL,

Date\_updated date,

Cv\_section\_text varchar(100),

CONSTRAINT PK\_CV\_SECTIONS PRIMARY KEY(Cv\_section\_id),

CONSTRAINT FK\_Cv\_id\_CV\_SECTIONS FOREIGN KEY(Cv\_id)

REFERENCES CVS(Cv\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Cv\_section\_code\_CV\_SECTIONS FOREIGN KEY(Cv\_section\_code)

REFERENCES REF\_CV\_SECTIONS(Cv\_section\_code)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create table FEED\_CATEGORY (

Feed\_category\_id int not null,

Isim varchar(45),

CONSTRAINT PK\_FEED\_CATEGORY PRIMARY KEY (Feed\_category\_id)

);

Create Table FEED\_SUB\_CATEGORY

(

Feed\_sub\_category\_id int NOT NULL,

Isim varchar(45),

Feed\_category\_id int NOT NULL,

CONSTRAINT PK\_FEED\_SUB\_CATEGORY PRIMARY KEY(Feed\_sub\_category\_id),

CONSTRAINT FK\_Feed\_category\_id\_FEED\_SUB\_CATEGORY FOREIGN KEY(Feed\_category\_id)

REFERENCES FEED\_CATEGORY(Feed\_category\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create table FEED (

Feed\_id int not null,

Feed\_url varchar(255),

Rating int,

created\_at date Not null,

feed\_category\_id int,

feed\_sub\_category\_id int,

Privacy\_id int,

CONSTRAINT PK\_FEED PRIMARY KEY (Feed\_id),

CONSTRAINT FK\_feed\_category\_id\_FEED foreign key (feed\_category\_id)references FEED\_CATEGORY (feed\_category\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_feed\_sub\_category\_id\_FEED foreign key (feed\_sub\_category\_id)references FEED\_SUB\_CATEGORY (feed\_sub\_category\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Feed foreign key (Privacy\_id)references PRIVACY (Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create table FEED\_INFO (

Feed\_info\_id int not null,

Feed\_id int not null,

member\_id int NOT NULL,

favorite boolean,

clicks int ,

Privacy\_id int,

CONSTRAINT PK\_Feed\_info\_id\_FEED\_INFO PRIMARY KEY (Feed\_info\_id),

CONSTRAINT FK\_Feed\_id\_FEED\_INFO foreign key (Feed\_id)references FEED(feed\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_member\_id\_FEED\_INFO foreign key (member\_id)references MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Feed\_Info foreign key (Privacy\_id)references PRIVACY (Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

create table CONNECTIONS

(

Member\_id int not null,

Connection\_member\_id int not null,

Date\_connection date not null,

CONSTRAINT PK\_CONNECTIONS PRIMARY KEY (Member\_id,Connection\_member\_id),

CONSTRAINT FK\_CONNECTIONS FOREIGN KEY (Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK2\_CONNECTIONS FOREIGN KEY (Connection\_member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table CHAT

(

Member\_id int not null,

Chat\_member\_id int NOT NULL,

Msg varchar(100) NOT NULL,

Created\_at date NOT NULL,

CONSTRAINT PK\_CHAT PRIMARY KEY(Member\_id,Chat\_member\_id),

CONSTRAINT FK\_CHAT FOREIGN KEY(Member\_id,Chat\_member\_id)

REFERENCES CONNECTIONS(Member\_id,Connection\_member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

create table PEOPLE\_BEING\_FOLLOWED

(Member\_id int NOT NULL,

Member\_being\_followed int NOT NULL ,

Date\_started\_following date not null,

Date\_stopped\_following date,

CONSTRAINT PK\_PEOPLE\_BEING\_FOLLOWED PRIMARY KEY (Member\_id,Member\_being\_followed),

CONSTRAINT FK\_PEOPLE\_FOLLOWED FOREIGN KEY (Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_PEOPLE\_BEING\_FOLLOWED FOREIGN KEY (Member\_being\_followed)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create table RECOMMENDATIONS

(Member\_recommending\_id int not null,

Member\_being\_recommendend\_id int not null,

Date\_of\_recommendation date not null,

CONSTRAINT PK\_RECOMMENDATIONS PRIMARY KEY (Member\_recommending\_id,Member\_being\_recommendend\_id),

CONSTRAINT FK2\_RECOMMENDATIONS FOREIGN KEY (Member\_recommending\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_RECOMMENDATIONS FOREIGN KEY (Member\_being\_recommendend\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create table STATU

(

Status\_id int not null,

Message varchar(100),

Created\_at date not null,

Thumps\_up int,

Thumps\_down int,

/\*Privacy int,\*/

Is\_reply boolean,

To\_fb boolean,

To\_twitter boolean,

Member\_id int not null,

Privacy\_id int,

CONSTRAINT PK\_STATUS primary key(Status\_id),

CONSTRAINT FK\_STATUS foreign key (Member\_id) REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Privacy\_id\_Statu foreign key (Privacy\_id)references PRIVACY (Privacy\_id)

ON DELETE SET NULL ON UPDATE CASCADE

);

Create table COMMENTS

( comment\_id int not null,

Message varchar(100),

Created\_at date not null,

Status\_id int not null,

Commnet\_member\_id int NOT NULL,

CONSTRAINT PK\_COMMENTS primary key(comment\_id),

CONSTRAINT FK\_Status\_id\_COMMENTS foreign key (Status\_id) REFERENCES STATU(Status\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Commnet\_member\_id\_COMMENTS foreign key (Commnet\_member\_id) REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create table THUMB\_UP\_DOWN

(

Thump\_up\_down\_id int not null,

Flag boolean,

Created\_at date not null,

Status\_id int not null,

Begenen\_member int NOT NULL,

CONSTRAINT PK\_THUMP\_UP\_DOWN primary key(thump\_up\_down\_id),

CONSTRAINT FK\_THUMP\_UP\_DOWN foreign key (Status\_id) REFERENCES STATU(Status\_id)

ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT FK\_Begenen\_member foreign key (Begenen\_member) REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table OYUN

(

Oyun\_id int NOT NULL,

Oyun\_adi varchar(100),

CONSTRAINT PK\_OYUN PRIMARY KEY (Oyun\_id)

);

Create Table DURTME

(

Durtme\_id int NOT NULL,

Durten\_id int NOT NULL,

Durtulen\_id int NOT NULL,

CONSTRAINT PK\_DURTME PRIMARY KEY (Durtme\_id),

CONSTRAINT FK\_Durten\_id\_Durtulen\_id FOREIGN KEY (Durten\_id,Durtulen\_id)

REFERENCES CONNECTIONS(Member\_id,Connection\_member\_id)/\*uygulamalardaki surtme id ile sorun olabilir\*/

ON DELETE CASCADE ON UPDATE CASCADE

);

Create Table ETKINLIK

( Etkinlik\_id int NOT NULL,

Etkinlik\_adi varchar(50),

CONSTRAINT PK\_ETKINLIK PRIMARY KEY (Etkinlik\_id)

);

Create Table UYGULAMALAR

(

Uygulama\_id int NOT NULL,

Member\_id int,

Etkinlik\_id int,

Durtme\_id int,

Oyun\_id int,

CONSTRAINT PK\_UYGULAMALAR PRIMARY KEY (Uygulama\_id),

CONSTRAINT FK\_Etkinlik\_id\_UYGULAMALAR FOREIGN KEY(Etkinlik\_id)

REFERENCES ETKINLIK(Etkinlik\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_Durtme\_id\_UYGULAMALAR FOREIGN KEY(Durtme\_id)

REFERENCES DURTME(Durtme\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_Oyun\_id\_UYGULAMALAR FOREIGN KEY(Oyun\_id)

REFERENCES OYUN(Oyun\_id)

ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT FK\_Member\_id\_UYGULAMALAR FOREIGN KEY(Member\_id)

REFERENCES MEMBER(Member\_id)

ON DELETE CASCADE ON UPDATE CASCADE

);