

T.C.
İZMİR EKONOMİ ÜNİVERSİTESİ

MBP 203
VERİ TABANI VE YÖNETİMİ
ONLINE KURS SİSTEMİ

20212425023
Mustafa Cüneyt Kafes

Öğretim Üyesi
Burak Evrentuğ

İZMİR - 2022

- 1. ÖZET**
- 2. GİRİŞ**
- 3. BÖLÜM 1: MEVCUT SİSTEMİN TANIMI, İNCELENMESİ**
- 4. BÖLÜM 2: VERİ TABANININ OLUŞTURULMASI**
 - **KISIM 1: DDL KODLARI**
 - **KISIM 2: DML KODLARI**
- 5. BÖLÜM 3: OLASI SORUNLARIN TESPİTİ VE ÇÖZÜMÜ**
 - **KISIM 1: DQL KODLARI VE YORDAMLAR**
 - **KISIM 2: TETİKLEYİCİLER**
- 6. SONUÇ VE ÖNERİLER**
- 7. REFERANSLAR**

ÖZET

E-ticaret sitelerinin günümüzde sıkça kullanılıyor olması yanı sıra pandemi etkisiyle birlikte eğitim de sanal mecraaya taşınmış ve e-ticaret sitelerinin bir kısmı akademik içerikler de içermeye başlamıştır. Bu projede online kurs sistemi kullanan e-ticaret sitelerinin veri tabanı simüle edilmiş, işletmenin yaşayabileceği bazı sorunlar ele alınarak uygun çözümler üretilmiştir. Projenin amacı bu sektördeki işletmelerin veri saklama ve işlemeye ilgili yaşayabileceği sorunları önceden öngörebilmek ve çözebilmektir. Buna bağlı olarak mümkün olduğunca az tablo kullanılmış, olası suistimal durumlarına karşı işletmeyi koruyabilmek adına yordamlar kullanılarak log kayıt sistemleri eklenmiş ve veri kaybı en aza indirilmeye çalışılmıştır.

Proje kapsamında MySQL veri tabanı, yordamlar (procedures), tetikleyiciler (triggers), DDL, DML ve DQL bağlamlarında SQL kodları kullanılmıştır. Uygulamaya geçilmesi durumunda, başlangıç için bünyesinde Linux işletim sistemi, 1024MB Ram, 1 Core CPU ve 10GB disk kapasitesi bulunan bir sunucuda veri tabanının kurulması ve işlenmesi yeterli görülmüştür. İşletmenin hizmet verdiği ve bünyesinde çalışan olarak bulundurduğu kişi sayısı arttıkça sunucu donanımının yükseltilmesi ve sürekli güncel tutulması gerektiği unutulmamalıdır.

Anahtar kelimeler: Online Kurs Sistemi, E-ticaret, Veri Tabanı

GİRİŞ

Cep telefonları, internet, bilgisayar gibi elektronik cihazlar günümüzde olmazsa olmazımız haline gelmiştir. Bize çeşitli alanlarda kolaylıklar sağlayan bu teknolojiler üzerinden araştırma, öğrenme ve alışveriş yapma oranı hızla artmaktadır. Buna bağlı olarak, sanal mecralar işletmeler için büyük bir pazar haline gelmiştir.

İnternet kullanımı hakkında bilgi sahibi olmak internet üzerinden satış yapabilmek veya hizmet sunabilmek için yeterli değildir. Bunun yanı sıra internet kullanıcılarının dikkatini çekecek şekilde görsel desteği sağlanmakla birlikte kullanıcıların güven içerisinde alışverişlerini yapması işletme tarafından sağlanması gereken en önemli hususlardan biridir. Sistemde çıkacak herhangi bir sorun veya kötü niyetli kişiler tarafından yararlanılan bir zafiyet işletmeyi büyük zararlara sokabileceği gibi müşterinin güvenini ve memnuniyetini zedeleyerek işletmenin imajını büyük oranda kötü etkileyecektir.

Online Kurs Sistemi projesi sayesinde, Eğitim sektöründe bulunan ve işletmesini e-ticaret ortamına taşımak isteyen bir kurum için bir veri tabanı yapısı simüle edilerek karşılaşılabilecek sorunlar önceden tespit edilmeye çalışılmış ve buna uygun çözümler getirilmiştir. Projenin amacı, bahsi geçen işletmelere farkındalık kazandırarak bazı sorunları önceden öngörmesine yardımcı olmak, uygun çözümler geliştirmektir.

Proje, MySQL tabanlı olup çeşitli yordamlar ve tetikleyiciler ile DDL, DML, DQL bağlamlarında SQL kodları kullanılarak tasarlanmıştır.

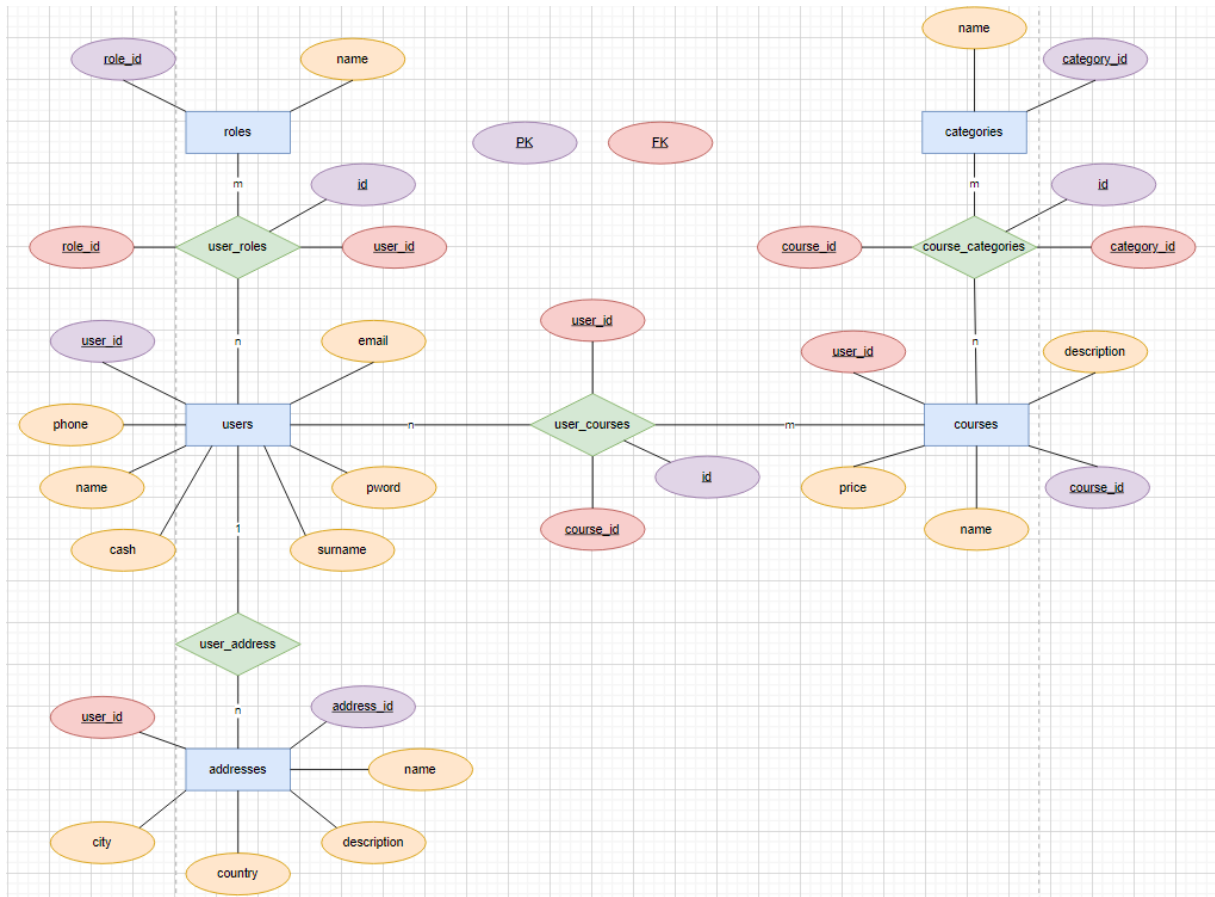
Proje eğitim sektöründe bulunan ve işletmesini e-ticaret ortamına taşımak isteyen her kurum tarafından kullanılabilir.

BÖLÜM 1

MEVCUT SİSTEMİN TANIMI, İNCELENMESİ

Kullanılan veri tabanının yapısını anlaşılabilir hale getirmek için varlık ilişkisi diyagramı (Entity-Relationship Diagram) kullanılmıştır. Varlık ilişkisi diyagramı veri tabanı örneklerinin yapısını tanımlayan, günümüzde sıklıkla kullanılan bir diyagramdır. Chen Notasyonu aktif olarak kullanılan ve alternatif varlık-ilişki modellemelerinden daha fazla detay sunduğu kabul edilen bir gösterim notasyonudur. Projenin anlaşılabilirliği açısından varlık ilişkisi diyagramında Chen Notasyonu kullanılarak ilişki gösterimi yapılmıştır.

Aşağıdaki görselde projenin yapısına ilişkin varlık ilişkisi diyagramı bulunmaktadır.



Diyagramda oval şekiller attribute (nitelik), kare şekiller entity(varlık), elmas şekiller relationship (ilişki) özelliklerini temsil etmektedir. Mor ile gösterilen nitelikler veri tabanı yapısında primary key (tekil anahtar) özelliğini, kırmızı ile gösterilen nitelikler veri tabanı yapısında foreign key (ikincil anahtar) özelliğini temsil etmektedir.

Veri tabanında tüm kullanıcılar, “users” isimli tek bir tabloda yer almaktadır. Bu sistemde; kullanıcıların işletme yöneticisi, eğitmen veya müşteri olduğu “roles” tablosunda ilişkilendirildiği rolden anlaşılabilecektir. Bunun için “user_roles” isiminde bir ilişki tablosu oluşturulması gerekmektedir. Sistemin dinamikliği için bu tablolar arasında çoka çok (many-to-many) ilişki kullanılmıştır. Örneğin; siteye kayıt yaptırmış olan herkes “kullanıcı” rolüne sahip olacak fakat işletme yöneticisinin aynı zamanda “yönetici” rolü de bulunacaktır. Bu sayede sadece siteye kayıtlı kullanıcıların görebileceği bir sistem geliştirmek istiyorsak görsel arayüzde her rol için farklı farklı sorgu yazılmasına gerek kalmadan tek bir sorgu ile işlem halledilecektir. Ek olarak, sistemde kursların yer aldığı “courses” tablosu ve kurs kategorilerinin bulunacağı “categories” tablosu yer almaktadır ve kursların ilgili kategorilerle eşleştirilebilmesi için “course_categories” isimli ek tablo eklenmiştir. Bir kursun birden fazla kategoride bulunabileceği öngörülerek bu tablolar arasında da çoka çok ilişki olduğu belirtilmiştir. “user_courses” isimli tablo kullanıcıların satın aldığı, erişiminin olduğu kursları belirtmektedir, bir kullanıcı birden fazla kursa sahip olabileceği ve bir kursu alan birden fazla kullanıcı olabileceği için bu aşamada da çoka çok ilişki kullanılmıştır. “addresses” tablosu ise, kullanıcıların faturalandırma için kullanabileceği birden fazla adres olabileceği düşünülerek oluşturulmuştur. Bir kullanıcının birden fazla adresi olabileceği fakat bir adres sadece bir kullanıcıya özel olduğu için “addresses” ve “users” tabloları arasında bire çok (one-to-many) ilişki kullanılmıştır.

Veri tabanının oluşturulmuş hali aşağıdaki görselde yer almaktadır.

Tablo	Eylem	Satır	Türü	Karşılaştırma	Boyut	Ek Yük
<input type="checkbox"/> addresses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	32.0 KiB	-
<input type="checkbox"/> categories	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	11	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> course_categories	★ Gözet Yapı Ara Ekle Boşalt Kaldır	14	InnoDB	utf8_turkish_ci	48.0 KiB	-
<input type="checkbox"/> roles	★ Gözet Yapı Ara Ekle Boşalt Kaldır	4	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> users	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	32.0 KiB	-
<input type="checkbox"/> user_courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	11	InnoDB	utf8_turkish_ci	48.0 KiB	-
<input type="checkbox"/> user_roles	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	48.0 KiB	-
8 tabloları	Toplam	80	InnoDB	utf8_turkish_ci	256.0 KiB	0 B

BÖLÜM 2

VERİ TABANININ OLUŞTURULMASI

Veri tabanı, XAMPP uygulamasının kullanımıyla MySQL veri tabanı sunucusunda local (yerel) olarak oluşturulmuştur. Projede sırasıyla DDL, DML, DQL bağlamlarıyla SQL kodları kullanılmıştır. İlgili kodlar aşağıda yer alan 1, 2 ve 3. Kısımlarda gösterilmiştir.

KISIM 1: DDL KODLARI

1- Veri tabanının oluşturulması

Aşağıdaki görselde “online_kurs_sistemi” isimli veri tabanının oluşturulmasına yönelik ilgili kod satırı gösterilmiştir.

```
✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0008 saniye sürdü.)

CREATE DATABASE online_kurs_sistemi;

[ Satır içi düzenle ] [ Düzenle ] [ PHP kodu oluştur ]

⚠ Error: #1046 No database selected
```

2- Users tablosunun oluşturulması

Aşağıdaki görselde “users” tablosu oluşturulmuş, constraint yapısıyla email sütunu Unique Key olarak tanımlanmıştır.

```
✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0003 saniye sürdü.)

CREATE TABLE users( user_id int PRIMARY KEY, name varchar(100) NOT NULL, surname varchar(100) NOT NULL, phone varchar(20), email varchar(255) NOT NULL, pword
varchar(255) NOT NULL, cash decimal(18,2) NOT NULL, CONSTRAINT email_uk UNIQUE (email) );

[ Satır içi düzenle ] [ Düzenle ] [ PHP kodu oluştur ]
```

3- Courses tablosunun oluşturulması

Aşağıdaki görselde “courses” tablosu oluşturulmuş, constraint yapısıyla instructor_id sütunundaki değerlerin users tablosundaki user_id değeriyle ilişkili olduğunu belirten Foreign Key yapısı tanımlanmıştır.

✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0003 saniye sürdü.)

```
CREATE TABLE courses( course_id int PRIMARY KEY, instructor_id int, name varchar(255) NOT NULL, price decimal(18,0), description varchar(255), CONSTRAINT instructor_fk FOREIGN KEY (instructor_id) REFERENCES users(user_id) );
```

[Satır içi düzenle] [Düzenle] [PHP kodu oluştur]

4- Roles tablosunun oluşturulması

Aşağıdaki görselde “roles” tablosu oluşturulmuştur.

Sorgu kutusunu göster

✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0003 saniye sürdü.)

```
CREATE TABLE roles( role_id int PRIMARY KEY, name varchar(255) );
```

[Satır içi düzenle] [Düzenle] [PHP kodu oluştur]

5- Categories tablosunun oluşturulması

Aşağıdaki görselde “categories” tablosu oluşturulmuştur.

Sorgu kutusunu göster

✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0003 saniye sürdü.)

```
CREATE TABLE categories( category_id int PRIMARY KEY, name varchar(255) NOT NULL );
```

[Satır içi düzenle] [Düzenle] [PHP kodu oluştur]

6- Addresses tablosunun oluşturulması

Aşağıdaki görselde “addresses” tablosu oluşturulmuş, constraint yapısıyla user_id değerinin users tablosundaki user_id ile bağlantılı olduğu Foreign Key tanımıyla belirtilmiştir.

Sorgu kutusunu göster

✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0004 saniye sürdü.)

```
CREATE TABLE addresses( address_id int PRIMARY KEY, user_id int, name varchar(255) NOT NULL, country varchar(255) NOT NULL, city varchar(255) NOT NULL, description varchar(255), CONSTRAINT address_fk FOREIGN KEY (user_id) REFERENCES users(user_id) );
```

[Satır içi düzenle] [Düzenle] [PHP kodu oluştur]

7- User_roles ilişki tablosunun oluşturulması

Aşağıdaki görselde user ve roles tablolarındaki elemanların ilişkisini bünyesinde bulunduracak olan user_roles tablosu oluşturulmuş, constraint yapısıyla user_id değerinin users tablosundaki user_id ile, role_id değerinin ise roles tablosundaki role_id ile bağlantılı olduğu Foreign Key tanımıyla belirtilmiştir.

✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0003 saniye sürdü.)

```
CREATE TABLE user_roles( id int PRIMARY KEY AUTO_INCREMENT, user_id int NOT NULL, role_id int NOT NULL, CONSTRAINT uroles_user FOREIGN KEY (user_id) REFERENCES users(user_id), CONSTRAINT uroles_role FOREIGN KEY (role_id) REFERENCES roles(role_id) );
```

[Satır içi düzenle] [Düzenle] [PHP kodu oluştur]

8- User_courses ilişki tablosunun oluşturulması

Aşağıdaki görselde user ve courses tablolarındaki elemanların ilişkisini bünyesinde bulunduracak olan user_courses tablosu oluşturulmuş, constraint yapısıyla user_id değerinin users tablosundaki user_id ile, course_id değerinin ise courses tablosundaki course_id ile bağlantılı olduğu Foreign Key tanımıyla belirtilmiştir. Bu tablo kullanıcıların hangi kurslara erişimi olduğunu belirtecektir.

✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0004 saniye sürdü.)

```
CREATE TABLE user_courses( id int PRIMARY KEY AUTO_INCREMENT, user_id int NOT NULL, course_id int NOT NULL, CONSTRAINT ucourses_user FOREIGN KEY (user_id) REFERENCES users(user_id), CONSTRAINT ucourses_course FOREIGN KEY (course_id) REFERENCES courses(course_id) );
```

[Satır içi düzenle] [Düzenle] [PHP kodu oluştur]

9- Course_categories ilişki tablosunun oluşturulması

Aşağıdaki görselde categories ve courses tablolarındaki elemanların ilişkisini bünyesinde bulunduracak olan course_categories tablosu oluşturulmuş, constraint yapısıyla course_id değerinin course tablosundaki course_id ile, category_id değerinin ise categories tablosundaki category_id ile bağlantılı olduğu Foreign Key tanımıyla belirtilmiştir. Bu tablo kursların hangi kategorilerde yer aldığını belirtecektir.

```
✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0.0003 saniye sürdü.)

CREATE TABLE course_categories( id int PRIMARY KEY AUTO_INCREMENT, course_id int NOT NULL, category_id int NOT NULL, CONSTRAINT ccategories_course FOREIGN KEY (course_id) REFERENCES courses(course_id), CONSTRAINT ccategories_category FOREIGN KEY (category_id) REFERENCES categories(category_id) );

[ Satır içi düzenle ] [ Düzenle ] [ PHP kodu oluştur ]
```

Yukarıdaki görsellerde belirtilen kodların eksiksiz uygulanması sonucu veri tabanındaki tablo görüntüsü aşağıdaki gibi olacaktır.

Tablo	Eylem	Satır	Türü	Karşılaştırma	Boyut	Ek Yük
<input type="checkbox"/> addresses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	32.0 KiB	-
<input type="checkbox"/> categories	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	11	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> course_categories	★ Gözet Yapı Ara Ekle Boşalt Kaldır	14	InnoDB	utf8_turkish_ci	48.0 KiB	-
<input type="checkbox"/> roles	★ Gözet Yapı Ara Ekle Boşalt Kaldır	4	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> users	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	32.0 KiB	-
<input type="checkbox"/> user_courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	11	InnoDB	utf8_turkish_ci	48.0 KiB	-
<input type="checkbox"/> user_roles	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	48.0 KiB	-
8 tabloları	Toplam	80	InnoDB	utf8_turkish_ci	256.0 KiB	0 B

KISIM 2: DML KODLARI













1- Roles ve Users tablolarına veri eklenmesi

Aşağıdaki görselde roles tablosuna işletme yöneticisi, işletme yetkilisi, eğitmen ve kullanıcı olmak üzere 4 rol eklenmiş ve users tablosuna 10 adet farklı verilere sahip test kullanıcısı eklenmiştir.




```
online_kurs_sistemi veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ⓘ

1 INSERT INTO roles (role_id, name) VALUES (1, "owner");
2 INSERT INTO roles (role_id, name) VALUES (2, "admin");
3 INSERT INTO roles (role_id, name) VALUES (3, "instructor");
4 INSERT INTO roles (role_id, name) VALUES (4, "user");
5
6 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (1, 'mustafa cüneyt', 'kafes', '05309050790', 'm.cuneytkafes@gmail.com', '123asd', 100);
7 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (2, 'ahmet ali', 'yılmaz', '05324139007', 'a.yilmaz@gmail.com', '123asd', 2);
8 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (3, 'fatma', 'karabulut', '05321050500', 'f.karabulut@gmail.com', '123asd', 500);
9 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (4, 'kemal', 'karaca', '05302208970', 'k.karaca@gmail.com', '123asd', 70);
10 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (5, 'celal', 'korkmaz', '05327771354', 'celal.korkmaz@gmail.com', '123asd', 50);
11 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (6, 'defne deniz', 'aksoy', '05553280019', 'defne.aksoy@gmail.com', '123asd', 40);
12 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (7, 'berkay', 'karakus', '05527203941', 'b.karakus@gmail.com', '123asd', 30);
13 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (8, 'deniz', 'tunc', '05301850396', 'tunc.deniz@gmail.com', '123asd', 50);
14 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (9, 'aslı', 'gök', '05325070440', 'gok.asli@gmail.com', '123asd', 100);
15 INSERT INTO users (user_id, name, surname, phone, email, pword, cash) VALUES (10, 'zeki can', 'gözcü', '05321136290', 'gozcu.zeki@gmail.com', '123asd', 0);
```

Görseldeki kodun uygulanması durumunda roles tablosunun çıktısı aşağıdaki gibi olacaktır.

<div><div><div><div></div><div></div><div></div></div><div></div></div></div>					role_id	name
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	1	owner	
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	2	admin	
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	3	instructor	
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	4	user	

Görseldeki kodun uygulanması durumunda users tablosunun çıktısı aşağıdaki gibi olacaktır.

		user_id	name	surname	phone	email	pword	cash
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	1	mustafa cüneyt	kafes	05309050790	m.cuneytkafes@gmail.com	123asd	100.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	2	ahmet ali	yilmaz	05324139007	a.yilmaz@gmail.com	123asd	2.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	3	fatma	karabulut	05321050500	f.karabulut@gmail.com	123asd	500.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	4	kemal	karaca	05302208970	k.karaca@gmail.com	123asd	70.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	5	celal	korkmaz	05327771354	celal.korkmaz@gmail.com	123asd	50.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	6	defne deniz	aksoy	05553280019	defne.aksoy@gmail.com	123asd	40.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	7	berkay	karakuş	05527203941	b.karakus@gmail.com	123asd	30.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	8	deniz	tunç	05301850396	tunc.deniz@gmail.com	123asd	50.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	9	aslı	gök	05325070440	gok.asli@gmail.com	123asd	100.00
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	10	zeki can	gözcü	05321136290	gozcu.zeki@gmail.com	123asd	0.00

2- Categories tablosuna veri eklenmesi

Aşağıdaki görselde cateories tablosuna 10 adet farklı verilere sahip test kategorisi eklenmiştir.

```
online_kurs_sistemi veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ⓘ

1 INSERT INTO categories(category_id, name) VALUES (1, "Yazılım Geliştirme");
2 INSERT INTO categories(category_id, name) VALUES (2, "Kişisel Gelişim");
3 INSERT INTO categories(category_id, name) VALUES (3, "Tasarım");
4 INSERT INTO categories(category_id, name) VALUES (4, "Fotoğraf ve Video");
5 INSERT INTO categories(category_id, name) VALUES (5, "Sağlık ve Fitness");
6 INSERT INTO categories(category_id, name) VALUES (6, "Müzik");
7 INSERT INTO categories(category_id, name) VALUES (7, "Sınava Hazırlık");
8 INSERT INTO categories(category_id, name) VALUES (8, "Dil Öğrenimi");
9 INSERT INTO categories(category_id, name) VALUES (9, "El Sanatları");
10 INSERT INTO categories(category_id, name) VALUES (10, "Evcil Hayvan Bakımı");
```

Görseldeki kodun uygulanması durumunda categories tablosunun çıktısı aşağıdaki gibi olacaktır.

<div><div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div></div><div></div></div>						category_id	name	
<input type="checkbox"/>		Düzenle		Kopyala		Sil	1	Yazılım Geliştirme
<input type="checkbox"/>		Düzenle		Kopyala		Sil	2	Kişisel Gelişim
<input type="checkbox"/>		Düzenle		Kopyala		Sil	3	Tasarım
<input type="checkbox"/>		Düzenle		Kopyala		Sil	4	Fotoğraf ve Video
<input type="checkbox"/>		Düzenle		Kopyala		Sil	5	Sağlık ve Fitness
<input type="checkbox"/>		Düzenle		Kopyala		Sil	6	Müzik
<input type="checkbox"/>		Düzenle		Kopyala		Sil	7	Sınava Hazırlık
<input type="checkbox"/>		Düzenle		Kopyala		Sil	8	Dil Öğrenimi
<input type="checkbox"/>		Düzenle		Kopyala		Sil	9	El Sanatları
<input type="checkbox"/>		Düzenle		Kopyala		Sil	10	Evcil Hayvan Bakımı

3- Courses tablosuna veri eklenmesi

Aşağıdaki görselde courses tablosuna 11 adet farklı verilere sahip kurs eklenmiştir. Kurslara eğitmen olarak rastgele kullanıcılar atanmıştır.

```
online_kurs_sistemi veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ⓘ


1 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (1,4, 'Adım Adım Matematik', '34.00', 'Uygulamalarla adım adım matematik');
2 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (2,5, 'Köpek Eğitimi', '32.00', 'En saldırgan köpekleri bile dize getirebileceksiniz');
3 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (3,6, 'Premiere Pro ile Video Editleme', '39.99', 'Premiere Pro CC uygulamasını kullanabileceksiniz');
4 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (4,4, 'İngilizce Kursu', '49.90', 'Ana Dilin Gibi İngilizce Konuş!!');
5 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (5,5, 'Sıfırdan İleri Seviyeye Fransızca', '18.50', 'Gözünde büyüttüğün kadar zor değil');
6 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (6,4, 'TOEFL Kursu', '25.00', 'O sene bu sene! Bu kurs sayesinde artık TOEFL sınavı cepte!');
7 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (7,5, 'Seslendirme Sanatçılığı', '23.00', 'Ses sanatçısı olun');
8 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (8,6, 'Veri Tabanı Sistemleri', '39.90', 'SQL ile işlemler yapmayı öğrenin');
9 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (9,4, 'Web Geliştirme', '49.50', 'Kendi web sitenizi yaratmayı öğrenip, sektörde aranan kişi olacaksınız!');
10 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (10,4, 'Fotoğrafçılık Kursu', '45.00', 'Bu kurs ile bakış açınızı değiştireceksiniz');
11 INSERT INTO courses (course_id, instructor_id, name, price, description) VALUES (11,6, 'Girişimcilik Kursu', '55.00', 'Risk almaktan korkma!');
```

Görseldeki kodun uygulanması durumunda courses tablosunun çıktısı aşağıdaki gibi olacaktır.

		course_id	instructor_id	name	price	description
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	1	4	Adım Adım Matematik	34	Uygulamalarla adım adım matematik
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	2	5	Köpek Eğitimi	32	En saldırgan köpekleri bile dize getirebileceğini...
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	3	6	Premiere Pro ile Video Editleme	40	Premiere Pro CC uygulamasını kullanabileceksiniz
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	4	4	İngilizce Kursu	50	Ana Dilin Gibi İngilizce Konuş!!
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	5	5	Sıfırdan İleri Seviyeye Fransızca	19	Gözünde büyüttüğün kadar zor değil
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	6	4	TOEFL Kursu	25	O sene bu senel Bu kurs sayesinde artık TOEFL sına...
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	7	5	Seslendirme Sanatçılığı	23	Ses sanatçısı olun
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	8	6	Veri Tabanı Sistemleri	40	SQL ile işlemler yapmayı öğrenin
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	9	4	Web Geliştirme	50	Kendi web sitenizi yaratmayı öğrenip, sektörde ara...
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	10	4	Fotoğrafçılık Kursu	45	Bu kurs ile bakış açınızı değiştireceksiniz
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	11	6	Girişimcilik Kursu	55	Risk almaktan korkma!




























4- User_roles ilişki tablosuna veri eklenmesi

Aşağıdaki görselde users ve roles isimli tablolarda bulunan elemanlar arasındaki ilişkiyi temsil eden user_roles tablosuna 10 adet ilişki eklenmiştir. Özetle, kullanıcılara rol ataması yapılmıştır.

```
online_kurs_sistemi veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: 
```

```
1 INSERT INTO user_roles(user_id, role_id) VALUES (1,1);
2 INSERT INTO user_roles(user_id, role_id) VALUES (2,2);
3 INSERT INTO user_roles(user_id, role_id) VALUES (3,2);
4 INSERT INTO user_roles(user_id, role_id) VALUES (4,3);
5 INSERT INTO user_roles(user_id, role_id) VALUES (5,3);
6 INSERT INTO user_roles(user_id, role_id) VALUES (6,3);
7 INSERT INTO user_roles(user_id, role_id) VALUES (7,4);
8 INSERT INTO user_roles(user_id, role_id) VALUES (8,4);
9 INSERT INTO user_roles(user_id, role_id) VALUES (9,4);
10 INSERT INTO user_roles(user_id, role_id) VALUES (10,4);
```

Görseldeki kodun uygulanması durumunda user_roles tablosunun çıktısı aşağıdaki gibi olacaktır.

				id	user_id	role_id
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	1	1	1
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	2	2	2
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	3	3	2
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	4	4	3
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	5	5	3
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	6	6	3
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	7	7	4
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	8	8	4
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	9	9	4
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	10	10	4

5- Course_categories ilişki tablosuna veri eklenmesi





Aşağıdaki görselde courses ve categories isimli tablolarda bulunan elemanlar arasındaki ilişkiyi temsil eden course_categories tablosuna 11 adet ilişki eklenmiştir.

Özetle, kategorilere kurs ataması yapılmıştır.

online_kurs_sistemi veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır:

```
1 INSERT INTO course_categories(course_id, category_id) VALUES (1,7);
2 INSERT INTO course_categories(course_id, category_id) VALUES (2,10);
3 INSERT INTO course_categories(course_id, category_id) VALUES (3,3);
4 INSERT INTO course_categories(course_id, category_id) VALUES (3,4);
5 INSERT INTO course_categories(course_id, category_id) VALUES (4,7);
6 INSERT INTO course_categories(course_id, category_id) VALUES (4,8);
7 INSERT INTO course_categories(course_id, category_id) VALUES (5,7);
8 INSERT INTO course_categories(course_id, category_id) VALUES (5,8);
9 INSERT INTO course_categories(course_id, category_id) VALUES (6,7);
10 INSERT INTO course_categories(course_id, category_id) VALUES (7,6);
11 INSERT INTO course_categories(course_id, category_id) VALUES (8,1);
12 INSERT INTO course_categories(course_id, category_id) VALUES (9,1);
13 INSERT INTO course_categories(course_id, category_id) VALUES (10,4);
14 INSERT INTO course_categories(course_id, category_id) VALUES (11,2);
```


Görseldeki kodun uygulanması durumunda course_categories tablosunun çıktısı aşağıdaki gibi olacaktır.

				id	course_id	category_id
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	1	1	7
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	2	2	10
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	3	3	3
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	4	3	4
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	5	4	7
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	6	4	8
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	7	5	7
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	8	5	8
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	9	6	7
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	10	7	6
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	11	8	1
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	12	9	1
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	13	10	4
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	14	11	2

6- Addresses tablosuna veri eklenmesi


Aşağıdaki görselde addresses isimli tabloya 10 adet test verisi eklenmiştir.

Özetle, kullanıcılara adres ataması yapılmıştır.

```
online_kurs_sistemi veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ⓘ

1 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (1,1,"ev","Türkiye","Manisa","Yunusemre 1503 Sokak");
2 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (2,1,"eski ev","Türkiye","İzmir","Mansuroğlu Mah.");
3 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (3,2,"okul","Türkiye","İzmir","Balçova, İzmir Ekonomi Üniversitesi");
4 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (4,3,"ev","Türkiye","İzmir","Poligon, 1307 Sokak");
5 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (5,4,"deneme","Türkiye","Antalya","");
6 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (6,5,"universite","Türkiye","Manisa","Celal Bayar Üniversitesi");
7 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (7,6,"ev","Türkiye","Manisa","Yunusemre 1503 Sokak");
8 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (8,7,"iş","Türkiye","Antalya","Alanya");
9 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (9,8,"ev","Türkiye","İstanbul","Taksim");
10 INSERT INTO addresses(address_id, user_id, name, country, city, description) VALUES (10,9,"asddafs","Türkiye","Ankara","");
```

Görseldeki kodun uygulanması durumunda addresses tablosunun çıktısı aşağıdaki gibi olacaktır.

			address_id	user_id	name	country	city	description
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	1	1 ev	Türkiye	Manisa	Yunusemre 1503 Sokak
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	2	1 eski ev	Türkiye	İzmir	Mansuroğlu Mah.
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	3	2 okul	Türkiye	İzmir	Balçova, İzmir Ekonomi Üniversitesi
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	4	3 ev	Türkiye	İzmir	Poligon, 1307 Sokak
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	5	4 deneme	Türkiye	Antalya	
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	6	5 universite	Türkiye	Manisa	Celal Bayar Üniversitesi
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	7	6 ev	Türkiye	Manisa	Yunusemre 1503 Sokak
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	8	7 iş	Türkiye	Antalya	Alanya
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	9	8 ev	Türkiye	İstanbul	Taksim
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	10	9 asddafs	Türkiye	Ankara	














7- User_courses ilişki tablosuna veri eklenmesi

Aşağıdaki görselde users ve courses isimli tablolarda bulunan elemanlar arasındaki ilişkiyi temsil eden user_courses tablosuna 10 adet ilişki eklenmiştir. Özetle, kullanıcıların erişebileceği kurslar belirlenmiştir.

online_kurs_sistemi veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır:

```
1 INSERT INTO user_courses(user_id, course_id) VALUES (1,8);
2 INSERT INTO user_courses(user_id, course_id) VALUES (2,3);
3 INSERT INTO user_courses(user_id, course_id) VALUES (3,5);
4 INSERT INTO user_courses(user_id, course_id) VALUES (4,2);
5 INSERT INTO user_courses(user_id, course_id) VALUES (5,1);
6 INSERT INTO user_courses(user_id, course_id) VALUES (5,6);
7 INSERT INTO user_courses(user_id, course_id) VALUES (6,6);
8 INSERT INTO user_courses(user_id, course_id) VALUES (7,7);
9 INSERT INTO user_courses(user_id, course_id) VALUES (8,4);
10 INSERT INTO user_courses(user_id, course_id) VALUES (9,11);
11 INSERT INTO user_courses(user_id, course_id) VALUES (9,10);
```

Görseldeki kodun uygulanması durumunda user_courses tablosunun çıktısı aşağıdaki gibi olacaktır.

				id	user_id	course_id
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	1	1	8
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	2	2	3
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	3	3	5
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	4	4	2
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	5	5	1
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	6	5	6
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	7	6	6
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	8	7	7
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	9	8	4
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	10	9	11
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	11	9	10

BÖLÜM 3

OLASI SORUNLARIN TESPİTİ VE ÇÖZÜMÜ

Projede eğitim sektöründe bulunup işletmesini e-ticaret mecrasına taşımak isteyen bir kurumun yapması gerektiği sistemi ve karşılaşılabileceği hataları simüle ettiğimiz için bu aşamada kendimizi işletme sahibi yerine koyarak ne gibi ihtiyaçlarımız olabileceğini düşünmemiz gerekmektedir.

Bu aşamada işletmenin; yetkili listesini, adresi ekli olmayan kullanıcıları, kurs içermeyen kategorileri, bir eğitmene ait kursların fiyat ortalamasını, bir eğitmene ait kursları, en popüler kursu, belirli fiyat aralığındaki kursları, belirli kategorideki kursları ve belirli kursların sayısını, kullanıcıların hangi kurslara erişimi olduğunu, belirli bir kullanıcının adreslerini ve tüm kursları görebilmek isteyeceği, bunlara yönelik görsel arayüz kısmında bazı işlemler yapmak isteyebileceği öngörülmüştür. Bu sebeple, ilgili işlemleri gerçekleştirebilmek üzere yordamlar kullanılarak çeşitli DQL kodları yazılmıştır.

Ek olarak, etkili bir kullanıcının herhangi bir tabloda silme veya değiştirme işlemi yapmasıyla usulsüzlüğe sebebiyet vermesi gibi olası suistimal durumlarının önlenmesi ve işletmeyi yasal kanıtlarla rahatlatabilmesi adına tetikleyiciler kullanılması gerektiği düşünülmüştür. Bu aşamayla ilgili durumlar 2. Kısımda bahsedilmiştir.

KISIM 1: DQL KODLARI VE YORDAMLAR

1- Admin listesini getirmek

Aşağıdaki görselde admin_listesi isminde bir yordam oluşturularak. rol_id değeri 3'ten küçük olan (işletme yöneticisi ve işletme yetkilisi) kişilerin listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

admin_listesi

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü	Uzunluk/Değerler	Seçenekler
-----	-----	------	------------------	------------

Tanım

Parametre ekle

```
1 SELECT users.user_id, CONCAT(users.name," ",users.surname) AS
full_name, users.phone, users.email, users.cash, roles.role_id,
roles.name AS role
2 FROM users, user_roles, roles
3 WHERE users.user_id = user_roles.user_id
4 AND roles.role_id = user_roles.role_id
5 GROUP BY users.user_id HAVING roles.role_id < 3
```

Belirleyici

☐

Yetkileri

☒

Git

Kapat

2- Adresi olmayan kullanıcıları getirmek

Aşağıdaki görselde adresi_olmayan_kullanici isminde bir yordam oluşturularak addresses tablosunda id'si bulunmayan kullanıcıların listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

adres_i_olmayan_kullanici

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü	Uzunluk/Değerler	Seçenekler
-----	-----	------	------------------	------------

Tanım

Parametre ekle

```
1 SELECT users.user_id, CONCAT(users.name, " ",users.surname) AS  
kullanici, users.phone AS tel_no, users.email  
2 FROM users, roles, user_roles  
3 WHERE users.user_id NOT IN (SELECT addresses.user_id FROM  
addresses)  
4 AND roles.name LIKE "%user%"  
5 AND user_roles.role_id = roles.role_id  
6 AND user_roles.user_id = users.user_id
```

Belirleyici

☐

Yetkileri

☒

Git

Kapat

3- Kurs bulunmayan kategorileri getirmek

Aşağıdaki görselde bos_kategoriler isminde bir yordam oluşturularak course_categories tablosunda id'si bulunmayan kategorilerin listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

bos_kategoriler

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü	Uzunluk/Değerler	Seçenekler
-----	-----	------	------------------	------------

Tanım

Parametre ekle

```
1 SELECT categories.name AS kurs_eklenmemis_kategoriler
2 FROM categories
3 WHERE categories.name
4 NOT IN (
5     SELECT categories.name
6     FROM courses, categories, course_categories
7     WHERE courses.course_id = course_categories.course_id
8     AND course_categories.category_id =
9     categories.category_id
10 )
```

Belirleyici

☐

Yetkileri ayarla

☒

Git

Kapat

4- Eğitime ait kursların fiyat ortalamasını getirmek

Aşağıdaki görselde eğitmenin_fiyat_ortalamasi isminde bir yordam oluşturularak, ismi verilen bir eğitmenin vermiş olduğu kursların fiyat ortalamasının hesaplanması sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

egitmenin_fiyat_ortalamasi

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü
IN	egitmen_adi	VARCHAR

Tanım

Parametre ekle

```
1 SELECT users.name AS egitmen, ROUND(AVG(courses.price),2)
2 AS kurslarinin_fiyat_ortalamasi
3 FROM users, courses
4 WHERE users.user_id = courses.instructor_id
5 AND (
6     (users.name LIKE CONCAT("%",egitmen_adi,"%"))
7     OR
8     (users.surname LIKE CONCAT("%",egitmen_adi,"%"))
9 )
```

Git

Kapat

5- Eğitime ait kursları getirmek

Aşağıdaki görselde eğitmenin_kurslari isminde bir yordam oluşturularak, ismi verilen bir eğitmenin vermiş olduğu tüm kursların listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

egitmenin_kurslari

Türü

PROCEDURE

Parametreler

Yön

Adı

Türü

IN

egitmen_adi

VARCHAR

Tanım

Parametre ekle

```
1 SELECT CONCAT(users.name," ",users.surname) AS egitmen, courses.name
2 FROM users, courses, roles, user_roles
3 WHERE roles.name LIKE "%instructor%"
4 AND user_roles.role_id = roles.role_id
5 AND user_roles.user_id = users.user_id
6 AND (
7     (users.name LIKE CONCAT("%",egitmen_adi,"%"))
8     OR
9     (users.surname LIKE CONCAT("%",egitmen_adi,"%")))
10 AND users.user_id = courses.instructor_id
```

Git

Kapat

6- En popüler kursu getirmek

Aşağıdaki görselde en_populer_kurs isminde bir yordam oluşturularak, kursa erişimi olan kişi sayısı tespit edilmiş ve buna bağlı olarak en popüler olan kursun belirlenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

en_populer_kurs

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü	Uzunluk/Değerler	Seçenekler
-----	-----	------	------------------	------------

Tanım

Parametre ekle

```
1 SELECT courses.name AS course, categories.name AS category,
2 COUNT(users.user_id) AS students, courses.price,
3 courses.description
4 FROM categories, courses, course_categories, users,
5 user_courses
6 WHERE courses.course_id = course_categories.course_id
7 AND course_categories.category_id = categories.category_id
8 AND user_courses.course_id = courses.course_id
9 AND user_courses.user_id = users.user_id
10 GROUP BY users.user_id ORDER BY students
11 DESC LIMIT 1
```

Belirleyici

☐

Yetkileri

☒

Git

Kapat

7- Belirli fiyat aralığına göre kursları getirmek

Aşağıdaki görselde fiyat_filtrelemesi isminde bir yordam oluşturularak, verilen fiyat değerleri arasında bulunan kursların listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

fiyat_filtrelemesi

Türü

PROCEDURE

Parametreler

	Yön	Adı	Türü
↑	IN	minimum_fiyat	INT
↑	IN	maximum_fiyat	INT

Tanım

Parametre ek

```
1 SELECT courses.name, categories.name AS category,
2 CONCAT(users.name, " ",users.surname) AS instructor,
3 courses.price, courses.description
4 FROM courses, users, categories, course_categories
5 WHERE courses.price BETWEEN minimum_fiyat AND maximum_fiyat
6 AND courses.instructor_id = users.user_id
7 AND courses.course_id = course_categories.course_id
8 AND course_categories.category_id = categories.category_id
```

Git

Kapat

8- Kategorideki kurs sayısı

Aşağıdaki görselde kategorideki_kurs_sayisi isminde bir yordam oluşturularak, id'si verilen kategorideki kurs sayısının hesaplanması sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

kategorideki_kurs_sayisi

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü
IN	category_id	INT

Tanım

Parametre ek

```
1 SELECT categories.name AS kategori, COUNT(courses.course_id)
2 AS kategoride_bulunan_toplam_kurs_sayisi
3 FROM categories, courses, course_categories
4 WHERE course_categories.course_id = courses.course_id
5 AND course_categories.category_id = categories.category_id
6 AND categories.category_id = category_id
```

Git

Kapat

9- Kategorideki kursları getirmek

Aşağıdaki görselde kategorideki_kurslar isiminde bir yordam oluşturularak, ismi verilen kategorideki tüm kursların listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

kategorideki_kurslar

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü
IN	kategori	VARCHAR

Tanım

Parametre ekle

```
1 SELECT categories.name AS category, courses.name
2 AS course, CONCAT(users.name," ",users.surname)
3 AS instructor, courses.price, courses.description
4 FROM categories, courses, course_categories, users
5 WHERE course_categories.course_id = courses.course_id
6 AND course_categories.category_id = categories.category_id
7 AND categories.name LIKE CONCAT("%",kategori,"%")
8 AND users.user_id = courses.instructor_id
```

Git

Kapat

10- Kullanıcıların kurslarını getirmek

Aşağıdaki görselde `kullanici_larin_kurslari` isminde bir yordam oluşturularak, kullanıcıların hangi kurslara erişimi olduğunu tespit edilip listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

kullanici_larin_kurslari

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü	Uzunluk/Değerler	Seçenekler
-----	-----	------	------------------	------------

Tanım

Parametre ekle

```
1 SELECT CONCAT(users.name," ",users.surname) AS kullanıcı,  
  courses.name AS kurs FROM users, courses, user_courses  
2 WHERE users.user_id = user_courses.user_id  
3 AND user_courses.course_id = courses.course_id
```

Belirleyici

☐

Yetkileri

☒

Git

Kapat

11- Kullanıcının adreslerini getirmek

Aşağıdaki görselde kullanıcının_adresleri isminde bir yordam oluşturularak, ismi verilen kullanıcının sisteme kaydettiği tüm adreslerinin listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

kullanıcının_adresleri

Türü

PROCEDURE

Parametreler

Yön

Adı

Türü

IN

kullanici

VARCHAR

Tanım

Parametre ekle

```
1 SELECT CONCAT(users.name, " ",users.surname)
2 AS full_name, addresses.name, addresses.country,
3 addresses.city, addresses.description
4 FROM users, addresses
5 WHERE users.user_id = addresses.user_id
6 AND (
7     (users.name LIKE CONCAT("%",kullanici,"%"))
8     OR
9     (users.surname LIKE CONCAT("%",kullanici,"%"))
10 )
```

Git

Kapat

12- Kursları getirmek

Aşağıdaki görselde kurslari_getir isminde bir yordam oluşturularak, sistemde bulunan tüm kursların listelenmesi sağlanmıştır.

Düzenle

Ayrıntılar

Yordam adı

kurslari_getir

Türü

PROCEDURE

Parametreler

Yön	Adı	Türü	Uzunluk/Değerler	Seçenekler
-----	-----	------	------------------	------------

Tanım

Parametre ekle

```
1 SELECT courses.name AS kurs, CONCAT(users.name,"
",users.surname) AS egitmen, categories.name AS kategori,
courses.price AS fiyat, courses.description AS aciklama
2 FROM courses, categories, course_categories, users
3 WHERE courses.course_id = course_categories.course_id
4 AND course_categories.category_id = categories.category_id
5 AND courses.instructor_id = users.user_id GROUP BY kurs ORDER
BY kurs ASC
```

Belirleyici

☐

Yetkileri

☒

Git

Kapat

KISIM 2: TETİKLEYİCİLER

Önceden de bahsedildiği gibi, projede yetkili bir kullanıcının herhangi bir tabloda silme veya değiştirme işlemi yapmasıyla usulsüzlüğe sebebiyet vermesi gibi olası suistimal durumlarının önlenmesi ve işletmeyi yasal kanıtlarla rahatlatabilmesi adına tetikleyiciler kullanılması gerektiği düşünülmüştür.

Bunun için DDL kodları kullanılarak projeye ek olarak 4 tablo daha eklenmiştir. Veri kaybını önleyebilmek için silinen kursların tutulduğu “deleted_courses” tablosu, veri tabanında hangi kullanıcının ne işlemler yaptığını görebilmek ve işletmeye olası suistimal durumunda kanıt sağlayabilmek için “log_records” tablosu projeye dahil edilmiş; aynı zamanda işletmenin yıllık planlama vb. süreçlerde realist verilerle istatistiksel işlemler yapabilmesini sağlamak için, sistemde kayıtlı olan kursların toplam fiyat ortalamasını tutan “price_average_of_courses” tablosu ve sistemdeki toplam kurs sayısını tutan “total_courses” tablosu oluşturulmuştur.

1- Silinen kursların tutulmasını sağlamak

Aşağıdaki görselde veri kaybını önleyebilmek adına silinen kursların tutulacağı “deleted_courses” isimli tablosu oluşturulmuştur.

```
✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0003 saniye sürdü.)

CREATE TABLE deleted_courses( course_id int PRIMARY KEY, instructor_id int, name varchar(255), price decimal(18,2), description varchar(255) );

[ Satır içi düzenle ] [ Düzenle ] [ PHP kodu oluştur ]
```

Aşağıdaki görselde kurs_silindiginde_kaydet isimli tetikleyici oluşturulmuş, kurs tablosunda veri silme işlemi gerçekleştirildiğinde “deleted_courses” tablosuna silinen verinin kaydedilmesi sağlanmıştır.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_silindiginde_kaydet

Tablo

courses

Zaman

AFTER

Olay

DELETE

Tanım

```
1 INSERT INTO deleted_courses(course_id, instructor_id, name, price, description) VALUES (old.course_id, old.instructor_id, old.name, old.price, old.description)
```

Tanımlayıcı

root@localhost

Git

Kapat

2- Silinen kursların tutulmasını sağlamak

Aşağıdaki görselde veri tabanında hangi kullanıcının ne işlemler yaptığını görebilmek ve işletmeye olası suistimal durumunda kanıt sağlayabilmek için “log_records” tablosu oluşturulmuştur. Veri tabanında herhangi bir işlem yapıldığında bu tabloya kayıt düşmesi beklenmektedir.

```
✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0.0005 saniye sürdü.)

CREATE TABLE log_records( log_id int PRIMARY KEY AUTO_INCREMENT, description varchar(255) NOT NULL, actualized_by int, actualized_at timestamp NOT NULL, CONSTRAINT act_fk FOREIGN KEY (actualized_by) REFERENCES users(user_id) );

[ Satır içi düzenle ] [ Düzenle ] [ PHP kodu oluştur ]
```

Aşağıdaki görselde kurs silindiğinde “log_records” tablosuna veri düşmesi için “kurs_silindiğinde_log” isminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_silindiğinde_log

Tablo

courses

Zaman

AFTER

Olay

DELETE

Tanım

1 INSERT INTO log_records(description, actualized_by, actualized_at) VALUES (CONCAT(old.name," isimli kurs silindi. ID: ",old.course_id," - InstructorID", old.instructor_id),NULL,Now())

Tanımlayıcı

root@localhost

Git

Kapat

Aşağıdaki görselde yeni bir kullanıcı sisteme kaydolduğunda “log_records” tablosuna veri düşmesi için “kullanici_eklendiginde” isiminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kullanici_eklendiginde

Tablo

users

Zaman

AFTER

Olay

INSERT

Tanım

```
1 INSERT INTO log_records(description, actualized_by,
2   CONCAT(new.name," ",new.surname," isimli kullanıcı sisteme
3   kaydedildi. ID: ",new.user_id)
   ,NULL,Now())
```

Tanımlayıcı

root@localhost

Git

Kapat

Aşağıdaki görselde bir kullanıcının bilgileri değiştirildiğinde “log_records” tablosuna veri düşmesi için “kullanici_guncellendiginde” isminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kullanici_guncellendiginde

Tablo

users

Zaman

AFTER

Olay

UPDATE

Tanım

```
1 INSERT INTO log_records(description, actualized_by,
2   CONCAT(new.name, " ",new.surname, " isimli kullanıcının
3   bilgileri değiştirildi. ID: ",new.user_id)
   ,NULL,Now())
```

Tanımlayıcı

root@localhost

Git

Kapat

Aşağıdaki görselde sisteme yeni bir kurs eklendiğinde “log_records” tablosuna veri düşmesi için “kurs_eklendiğinde_log” isimde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_eklendiğinde_log

Tablo

courses

Zaman

AFTER

Olay

INSERT

Tanım

```
1 INSERT INTO log_records(description, actualized_by,
2   CONCAT(new.name," isimli kurs eklendi. ID: ",new.course_id,"
3   - InstructorID: ",new.instructor_id)
   ,NULL,Now())
```

Tanımlayıcı

root@localhost

Git

Kapat

Aşağıdaki görselde bir kurs bilgileri değiştirildiğinde “log_records” tablosuna veri düşmesi için “kurs_guncellendiginde_log” isminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_guncellendiginde_log

Tablo

courses

Zaman

BEFORE

Olay

UPDATE

Tanım

```
1 INSERT INTO log_records(description, actualized_by,
2   CONCAT(new.name," isimli kurs güncellendi. ID:
3   ",new.course_id," - InstructorID: ",new.instructor_id)
   ,NULL,Now())
```

Tanımlayıcı

root@localhost

Git

Kapat

3- Kursların toplam fiyat ortalamasını tutmak

Aşağıdaki görselde işletmenin yıllık planlama vb. süreçlerde realist verilerle istatistiksel işlemler yapabilmesini sağlamak için, sistemde kayıtlı olan kursların toplam fiyat ortalamasını tutmak adına “price_average_of_courses” tablosu oluşturulmuştur.

```
✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0003 saniye sürdü.)  
  
CREATE TABLE price_average_of_courses( id int PRIMARY KEY AUTO_INCREMENT, price_average int NOT NULL, last_checked_at timestamp NOT NULL );  
[ Satır içi düzenle ] [ Düzenle ] [ PHP kodu oluştur ]
```

Aşağıdaki görselde kurs bilgileri değiştirildiğinde “price_average_of_courses” tablosuna veri düşmesi için “kurs_guncellendiginde_ortalama” isminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_guncellendiginde_orte

Tablo

courses

Zaman

AFTER

Olay

UPDATE

Tanım

```
1 INSERT INTO price_average_of_courses(price_average,  
last_checked_at)  
2 VALUES ((SELECT AVG(courses.price) FROM courses), Now())
```

Tanımlayıcı

root@localhost

Git

Kapat

Aşağıdaki görselde bir kurs eklendiğinde “price_average_of_courses” tablosuna veri düşmesi için “kurs eklendiğinde_ortalama” isiminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_eklendiğinde_ortalama

Tablo

courses

Zaman

AFTER

Olay

INSERT

Tanım

```
1 INSERT INTO price_average_of_courses(price_average,
last_checked_at)
2 VALUES ((SELECT AVG(courses.price) FROM courses), Now())
```

Tanımlayıcı

root@localhost

Git

Kapat

Aşağıdaki görselde bir kurs silindiğinde “price_average_of_courses” tablosuna veri düşmesi için “kurs_silindiginde _ortalama” isminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_silindiginde_ortalama

Tablo

courses

Zaman

AFTER

Olay

DELETE

Tanım

```
1 INSERT INTO price_average_of_courses(price_average,
last_checked_at)
2 VALUES ((SELECT AVG(courses.price) FROM courses), Now())
```

Tanımlayıcı

root@localhost

Git

Kapat

4- Kursların toplam sayısını tutmak

Aşağıdaki görselde işletmenin yıllık planlama vb. süreçlerde realist verilerle istatistiksel işlemler yapabilmesini sağlamak için, sistemde kayıtlı olan kursların toplam sayısını tutmak adına “total_courses” tablosu oluşturulmuştur.

```
MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0,0003 saniye sürdü.)  
  
CREATE TABLE total_courses( id int PRIMARY KEY AUTO_INCREMENT, total_course_count int NOT NULL, last_checked_at timestamp NOT NULL );  
  
[ Satır içi düzenle ] [ Düzenle ] [ PHP kodu oluştur ]
```

Aşağıdaki görselde bir kurs eklendiğinde “total_courses” tablosuna veri düşmesi için “kurs_eklendiğinde_sayi” isminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_eklendiğinde_sayi

Tablo

courses

Zaman

AFTER

Olay

INSERT

1

INSERT INTO total_courses(total_course_count, last_checked_at)

2

VALUES ((SELECT COUNT(*) FROM courses), Now())

Tanım

Tanımlayıcı

root@localhost

Git

Kapat

Aşağıdaki görselde bir kurs silindiğinde “total_courses” tablosuna veri düşmesi için “kurs_silindiğinde_sayı” isminde tetikleyici eklenmiştir.

Düzenle

Ayrıntılar

Tetikleyici adı

kurs_silindiğinde_sayı

Tablo

courses

Zaman

AFTER

Olay

DELETE

Tanım

```
1 INSERT INTO total_courses(total_course_count, last_checked_at)
2 VALUES ((SELECT COUNT(*) FROM courses), Now())
```

Tanımlayıcı

root@localhost

Git

Kapat

SONUÇ VE ÖNERİLER

Tüm işlemler tamamlandıktan sonra proje veri tabanına ait tablo çıktısı aşağıdaki gibidir.

Tablo	Eylem	Satır	Türü	Karşılaştırma	Boyut	Ek Yük
<input type="checkbox"/> addresses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	32.0 KiB	-
<input type="checkbox"/> categories	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	11	InnoDB	utf8_turkish_ci	32.0 KiB	-
<input type="checkbox"/> course_categories	★ Gözet Yapı Ara Ekle Boşalt Kaldır	14	InnoDB	utf8_turkish_ci	48.0 KiB	-
<input type="checkbox"/> deleted_courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	3	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> log_records	★ Gözet Yapı Ara Ekle Boşalt Kaldır	9	InnoDB	utf8_turkish_ci	32.0 KiB	-
<input type="checkbox"/> price_average_of_courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	3	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> roles	★ Gözet Yapı Ara Ekle Boşalt Kaldır	4	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> total_courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	3	InnoDB	utf8_turkish_ci	16.0 KiB	-
<input type="checkbox"/> users	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	32.0 KiB	-
<input type="checkbox"/> user_courses	★ Gözet Yapı Ara Ekle Boşalt Kaldır	11	InnoDB	utf8_turkish_ci	48.0 KiB	-
<input type="checkbox"/> user_roles	★ Gözet Yapı Ara Ekle Boşalt Kaldır	10	InnoDB	utf8_turkish_ci	48.0 KiB	-
12 tabloları	Toplam	98	InnoDB	utf8_turkish_ci	352.0 KiB	0 B

Proje kapsamında eğitim sektöründe bulunup e-ticaret mecrasına geçiş yaparak işletmesinin sanal yüzünü oluşturmak isteyen bir kurumun, kurulumunu tamamlaması gereken bir veri tabanı sistem simüle edilmiş, karşılaşılabileceği hatalar öngörülerek düzeltilmeye çalışılmış aynı zamanda kurumun ihtiyaçları karşılanmıştır.

Projede çeşitli suistimal potansiyellerinin önüne geçilmiş olması, bazı tablolarda veri kaybının önlenmiş olması avantajlar arasında gösterilebilir.

Tetikleyicilerin artırılmasıyla diğer tablolara da kontrollerin eklenmesi, users tablosundaki şifre sütunda görsel programlama araçlarıyla kriptoloji sistemine uygun işlemler yapılması, rollerin sahip olduğu yetkileri tutacak farklı bir tablo yaratılarak arayüzde yetki kısıtlamalarının yapılması projeyi daha güvenli ve efektif hale getirmiş olacaktır.

REFERANSLAR

22 Aralık 2022 tarihinde:

[https://aktif.net/chen-notasyonu-ile-veritabani-tasarimi/#:~:text=VARLIK%20İLİŞKİSİ%20DİYAGRAMI%20\(ER%20DİYAGRAMI,örneklerinin%20yapısını%20tanımlayan%20bir%20diyagramdır](https://aktif.net/chen-notasyonu-ile-veritabani-tasarimi/#:~:text=VARLIK%20İLİŞKİSİ%20DİYAGRAMI%20(ER%20DİYAGRAMI,örneklerinin%20yapısını%20tanımlayan%20bir%20diyagramdır)

23 Aralık 2022 tarihinde:

https://www.w3schools.com/sql/sql_constraints.asp

23 Aralık 2022 tarihinde:

https://www.w3schools.com/sql/sql_ref_drop_constraint.asp

23 Aralık 2022 tarihinde:

<https://sebhasian.com/mysql-error-1452/>

23 Aralık 2022 tarihinde:

<https://www.tutorialspoint.com/mysql-error-1452-cannot-add-or-a-child-row-a-foreign-key-constraint-fails>