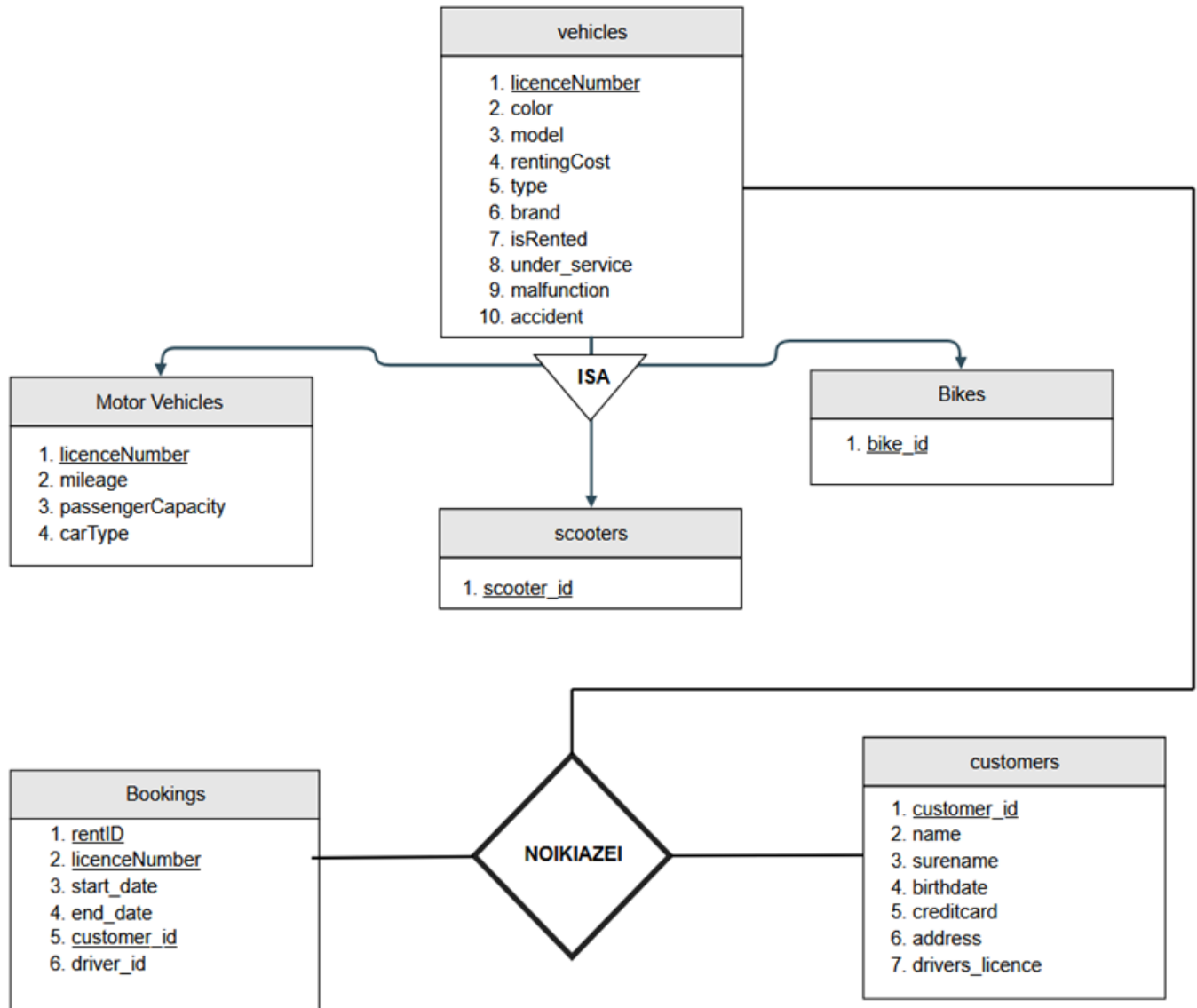


Διάγραμμα οντοτήτων σχέσεων



Πίνακες Σχεσιακού Μοντέλου

- Customers

#	Name	Type	Attributes	Null
1	customer_id	int(11)	None	No
2	name	varchar(30)	None	No
3	surname	varchar(30)	None	No
4	birthdate	date	None	No
5	creditcard	varchar(30)	None	No
6	address	varchar(30)	None	No
7	drivers_licence	varchar(30)	None	No

- Bikes

#	Name	Type	Attributes	Null
1	bike_id	int(11)	None	No

- Bookings

#	Name	Type	Attributes	Null
1	rentID	int(11)	None	No
2	licenceNumber	int(11)	None	No
3	start_date	date	None	No
4	end_date	date	None	No
5	customer_id	int(11)	None	No
6	driver_id	int(11)	None	Yes

- motor\_vehicles

#	Name	Type	Attributes	Null
1	licenceNumber	int(11)	None	No
2	mileage	varchar(30)	None	No
3	passengerCapacity	int(11)	None	No
4	carType	varchar(30)	None	No

- Scooters

#	Name	Type	Attributes	Null
1	scooter_id	int(11)	None	No

- Vehicles

#	Name	Type	Attributes	Null
1	licenceNumber	int(11)	None	No
2	color	varchar(30)	None	No
3	model	varchar(30)	None	No
4	rentingCost	bigint(20)	None	No
5	type	varchar(30)	None	No
6	brand	varchar(30)	None	No
7	isRented	tinyint(1)	None	No
8	under_service	tinyint(1)	None	No
9	malfunction	varchar(150)	None	No
10	accident	varchar(150)	None	No

### Πρωτεύοντα Κλειδιά

Στον πίνακα bikes, primary key→bike\_id

Στον πίνακα customers, primary key →customer\_id(auto increment)

Στον πίνακα bookings,primary key → rentID(auto increment)

Στους πίνακες vehicles και motor vehicles, primary key → licenceNumber

Στον πίνακα scooters, primary key→ scooter\_id

### Περιορισμοί πλυθικότητας

Όλα τα γνωρίσματα είναι απλά, δεν έχουμε σύνθετα η πλειότιμα γνωρίσματα.

Η σχέση isA είναι (1,1) (ένα vehicle αντιστοιχεί σε ένα type)

Η σχέση νοικιάζει είναι (N,N)(πολλές ενοικιάσεις από πολλούς χρήστες/πελάτες).

### Συναρτησιακές εξαρτήσεις

customer\_id→name,surname,birthdate,creditcard,address,drivers\_licence

rent\_id→licenceNumber,start\_date,end\_date,customer\_id,driver\_id

licenceNumber→color,model,rentingCost,type,brand,isRented,under\_service,malfunction,accident

licenceNumber→mileage,passengerCapacity,carType

(επίσης θεωρητικά τα bike\_id, scooter\_id βρίσκονται σε αριστερά μέλη συναρτησιακών εξαρτήσεων ωστόσο οι πίνακες bikes και scooters δεν διαθέτουν παραπάνω στοιχεία για να μπουν στο δεξί μέλος)

### 3η Κανονική Μορφή

Μια σχέση R βρίσκεται σε τρίτη κανονική μορφή αν :

1) Αν είναι σε 1-NF δηλαδή δεν έχει σύνθετα και πλειότιμα γνωρίσματα.

2) Αν είναι σε 2-NF δηλαδή αν κάθε συναρτησιακή εξάρτηση  $X \rightarrow Y$  που υπάρχει στην R είναι full functional

dependency. Μία συναρτησιακή εξάρτηση  $X \rightarrow Y$  είναι full functional dependency αν η συναρτησιακή εξάρτηση

παύει να ισχύει αν αφαιρέσουμε οποιοδήποτε πεδίο από το X .

3) Αν δεν υπάρχουν transitive dependencies (μεταβατικές εξαρτήσεις). Μία συναρτησιακή εξάρτηση  $X \rightarrow Y$  είναι

transitive dependencies αν ισχύει  $X \rightarrow Z$  και  $Z \rightarrow Y$  για κάποιο σύνολο από πεδία που δεν ανήκουν σε κανένα

κλειδί.

Σύμφωνα με τα παραπάνω η βάση είναι σε 3η Κανονική Μορφή.

### Εντολές SQL για την δημιουργία των πινάκων

#### customers:

```
CREATE TABLE `customers` (  
  `customer_id` int(11) NOT NULL AUTO_INCREMENT,  
  `name` varchar(30) NOT NULL,  
  `surname` varchar(30) NOT NULL,  
  `birthdate` date NOT NULL,  
  `creditcard` varchar(30) NOT NULL,  
  `address` varchar(30) NOT NULL,  
  `drivers_licence` varchar(30) NOT NULL,  
  PRIMARY KEY (`customer_id`)  
);
```

#### Παραδείγματα:

```
INSERT INTO `customers` (`name`, `surname`, `birthdate`, `creditcard`, `address`, `drivers_licence`)  
VALUES  
  ('Δημήτρης', 'Παπαδόπουλος', '1988-09-20', '1234-5678-9012-3456', 'Λεωφόρος Αθηνών 123',  
  'DL123456GR'),  
  ('Σοφία', 'Κωνσταντίνου', '1992-04-15', '9876-5432-1098-7654', 'Οδός Θεσσαλονίκης 456',  
  'DL789012GR'),  
  ('Γεώργιος', 'Νικολαΐδης', '1980-11-03', '5678-9012-3456-7890', 'Πλατεία Ηρακλείου 789',  
  'DL345678GR'),  
  ('Ελένη', 'Ανδρέου', '1995-07-28', '3456-7890-1234-5678', 'Οδός Πάτρας 101', 'DL901234GR'),  
  ('Αντώνης', 'Καραγιάννης', '1983-02-10', '7890-1234-5678-9012', 'Πλατεία Λάρισας 202',  
  'DL567890GR');
```

#### bookings:

```
CREATE TABLE `bookings` (  
  `rentID` int(11) NOT NULL AUTO_INCREMENT,  
  `licenceNumber` int(11) NOT NULL,  
  `start_date` date NOT NULL,  
  `end_date` date NOT NULL,  
  `customer_id` int(11) NOT NULL,  
  `driver_id` int(11) DEFAULT NULL,  
  PRIMARY KEY (`rentID`)  
);
```

Παραδείγματα:

```

INSERT INTO `bookings` (`rentID`, `licenceNumber`, `start_date`, `end_date`, `customer_id`,
`driver_id`)
VALUES
(1, 12345, '2024-02-01', '2024-02-07', 1, 101),
(2, 67890, '2024-03-15', '2024-03-22', 2, 102),
(3, 13579, '2024-04-10', '2024-04-15', 3, NULL),
(4, 24680, '2024-05-05', '2024-05-12', 4, 103),
(5, 98765, '2024-06-20', '2024-06-25', 5, NULL);

```

Σημείωση : το customer\_id, licenceNumber,driver\_id υπάρχουν στους πίνακες bookings,customers.

vehicles:

```

CREATE TABLE `vehicles` (
  `licenceNumber` int(11) NOT NULL,
  `color` varchar(30) NOT NULL,
  `model` varchar(30) NOT NULL,
  `rentingCost` bigint(20) NOT NULL,
  `type` varchar(30) NOT NULL,
  `brand` varchar(30) NOT NULL,
  `isRented` tinyint(1) NOT NULL,
  `under_service` tinyint(1) NOT NULL,
  `malfunction` varchar(150) NOT NULL,
  `accident` varchar(150) NOT NULL,
  PRIMARY KEY (`licenceNumber`)
);

```

**Παραδείγματα:**

```

INSERT INTO `vehicles` (`licenceNumber`, `color`, `model`, `rentingCost`, `type`, `brand`,
`isRented`, `under_service`, `malfunction`, `accident`)
VALUES
(12345, 'Blue', 'Sedan', 5000, 'Compact', 'Toyota', 0, 0, 'No issues', 'No accidents'),
(67890, 'Red', 'SUV', 8000, 'SUV', 'Ford', 1, 1, 'Needs oil change', 'Minor dent on rear bumper'),
(13579, 'Silver', 'Hatchback', 4500, 'Compact', 'Honda', 0, 0, 'Regular maintenance', 'No accidents'),
(24680, 'Black', 'Truck', 10000, 'Truck', 'Chevrolet', 0, 1, 'Brake inspection needed', 'No accidents'),
(98765, 'White', 'Convertible', 7000, 'Sports Car', 'BMW', 1, 0, 'Battery replacement required', 'Minor
scratch on driver's side');

```

motor vehicles:

```

CREATE TABLE `motor_vehicles` (
  `licenceNumber` int(11) NOT NULL,
  `mileage` varchar(30) NOT NULL,
  `passengerCapacity` int(11) NOT NULL,
  `carType` varchar(30) NOT NULL,
  PRIMARY KEY (`licenceNumber`)
);

```

scooters:

```
CREATE TABLE `scooter` (  
  `scooter_id` int(11) NOT NULL,  
  PRIMARY KEY (`scooter_id`)  
);
```

bikes:

```
CREATE TABLE `bike` (  
  `bike_id` int(11) NOT NULL,  
  PRIMARY KEY (`bike_id`)  
);
```

Σημείωση: Οι πίνακες motor\_vehicles, scooters και bikes είναι συμπληρωματικοί, δεν έχει νόημα να δείξουμε ξεχωριστά παραδείγματα insert.

Ερωτήσεις προς την βάση δεδομένων

Κατάσταση διαθέσιμων ή ενοικιασμένων οχημάτων ανά κατηγορία

Επιστρέφουμε όλα τα οχήματα, και την κατάσταση τους (αν είναι ενοικιασμένα ή όχι)

```
SELECT IsRented, type  
FROM Vehicles
```



	IsRented	type
<input type="checkbox"/> Edit Copy Delete	0	Car
<input type="checkbox"/> Edit Copy Delete	0	Car
<input type="checkbox"/> Edit Copy Delete	0	Scooter

Κατάσταση ενοικιάσεων ανά χρονική περίοδο

Ουσιαστικά επιστρέφουμε τα αυτοκίνητα που είναι νοικιασμένα την δεδομένη χρονική στιγμή

```
SELECT color, brand  
FROM Vehicles  
WHERE licenceNumber IN (SELECT licenceNumber  
                        FROM bookings)
```

```
SELECT color,brand FROM Vehicles WHERE licenceNumber IN (SELECT licenceNumber FROM bookings);
```

☐ Profiling [ [Edit inline](#) ] [ [Edit](#) ] [ [Explain SQL](#) ] [ [Create PHP code](#) ] [ [Refresh](#) ]

☐ Show all | Number of rows: 25 | Filter rows:

Extra options

← T →      ▼ color    brand  
☐ Edit Copy Delete black audi

Μέγιστη, ελάχιστη και μέση διάρκεια ενοικίασης ανά κατηγορία οχήματος

```
SELECT DATEDIFF(end_date,start_date)
FROM bookings
HAVING MAX(DATEDIFF(end_date,start_date));
```

```
SELECT DATEDIFF(end_date,start_date) FROM bookings HAVING MAX(DATEDIFF(end_date,start_date));
```

☐ Profiling [ [Edit inline](#) ] [ [Edit](#) ] [ [Explain SQL](#) ] [ [Create PHP code](#) ] [ [Refresh](#) ]

☐ Show all | Number of rows: 25 | Filter rows:

Extra options

DATEDIFF(end\_date,start\_date)  
7

```
SELECT DATEDIFF(end_date,start_date)
FROM bookings
HAVING MIN(DATEDIFF(end_date,start_date));
```

```
SELECT DATEDIFF(end_date,start_date) FROM bookings HAVING MIN(DATEDIFF(end_date,start_date));
```

☐ Profiling [ [Edit inline](#) ] [ [Edit](#) ] [ [Explain SQL](#) ] [ [Create PHP code](#) ] [ [Refresh](#) ]

☐ Show all | Number of rows: 25 | Filter rows:

Extra options

DATEDIFF(end\_date,start\_date)  
7

```
SELECT AVG(DATEDIFF(end_date,start_date)) FROM bookings;
```

```
SELECT AVG(DATEDIFF(end_date,start_date)) FROM bookings;
```

☐ Profiling [Edit inline](#) [Edit](#) [Explain SQL](#) [Create PHP code](#) [Refresh](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

```
AVG(DATEDIFF(end_date,start_date))
```

-11.0000

### Έσοδα από ενοικίαση ανά χρονική περίοδο και ανά κατηγορία οχήματος

SELECT rentingCost\*DATEDIFF(end\_date,start\_date) as kostos ,type,model,color  
FROM bookings,vehicles;

```
SELECT rentingCost*DATEDIFF(end_date,start_date) as esoda ,type,model,color FROM bookings,vehicles;
```

☐ Profiling [Edit inline](#) [Edit](#) [Explain SQL](#) [Create PHP code](#) [Refresh](#)

1 > >> | ☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

esoda	type	model	color
350	Car	q7	black
350	Car	q7	black
350	Scooter	q7	black
-700	Car	q7	black
-700	Car	q7	black
-700	Scooter	q7	black
-700	Car	q7	black
-700	Car	q7	black
-700	Scooter	q7	black
-700	Car	q7	black
-700	Car	q7	black
-700	Scooter	q7	black
-700	Car	q7	black
-700	Car	q7	black
-700	Scooter	q7	black
-700	Car	q7	black
-700	Car	q7	black
-700	Scooter	q7	black
-700	Car	q7	black
-700	Car	q7	black
-700	Scooter	q7	black
-700	Car	q7	black
-700	Car	q7	black
-700	Scooter	q7	black
-700	Car	q7	black

### Δημοφιλέστερο όχημα ανά κατηγορία

WITH RankedVehicles AS (  
 SELECT v.type,  
 v.model,  
 COUNT(\*) as rental\_count,  
 ROW\_NUMBER() OVER (PARTITION BY v.type ORDER BY COUNT(\*) DESC) as rank



```

FROM Vehicles v
JOIN Bookings b ON v.licenceNumber = b.licenceNumber
GROUP BY v.type, v.model
)
SELECT type, model, rental_count
FROM RankedVehicles
WHERE rank = 1;

```

✓ Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)

```

SELECT v.type, v.brand, v.licenceNumber, SUM(DATEDIFF(b.end_date, b.start_date) * v.rentingCost) as total_revenue
FROM Bookings b JOIN Vehicles v ON b.licenceNumber = v.licenceNumber GROUP BY v.type, v.brand, v.licenceNumber
ORDER BY total_revenue DESC;

```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

type	brand	licenceNumber	total_revenue
Car	audi	1	2027615000
Car	ferrari	123	252000

Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations

Print Copy to clipboard Export Display chart Create view

### Περιγραφή των περιορισμών της υλοποίησής και των δυνατοτήτων βελτίωσής του.

Στο αρχείο ajax.js έχουν υλοποιηθεί πολλές γραμμές κώδικα, δηλαδή πολλά ajax πρωτόκολλα που θα μπορούσαν να είχαν αποφευχθεί με σωστή χρήση των συναρτήσεων της js. Επιπλέον, μέσω των servlets θα μπορούσε να επιτευχθεί η επικοινωνία με την βάση με λιγότερες γραμμές κώδικα. Τέλος, το project μας είναι πλήρως λειτουργικό και τυχόν πλεονάζοντα αντικείμενα υπάρχουν λόγω της ταυτόχρονης υλοποίησης και σχεδίασης του κώδικα, τα οποία πλέον μας είναι ευδιάκριτα για την επόμενη φορά.