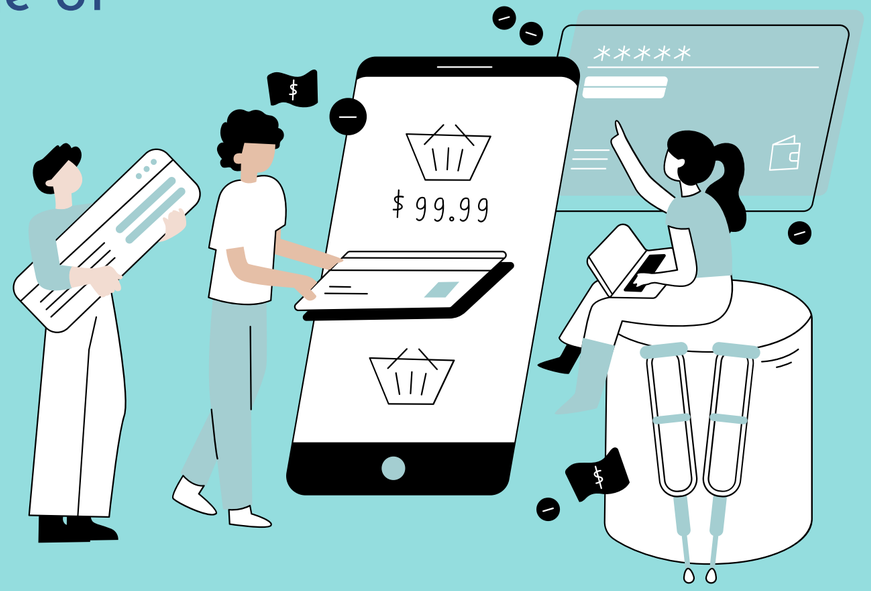
Creating Database

E-Commerse Database

The purpose of this database is to efficiently and seamlessly store the data in an e-commerce platform. Its goal is to ensure the efficient and smooth storage of data from an e-commerce platform.



Description of the Database

The Database has 5 entities;

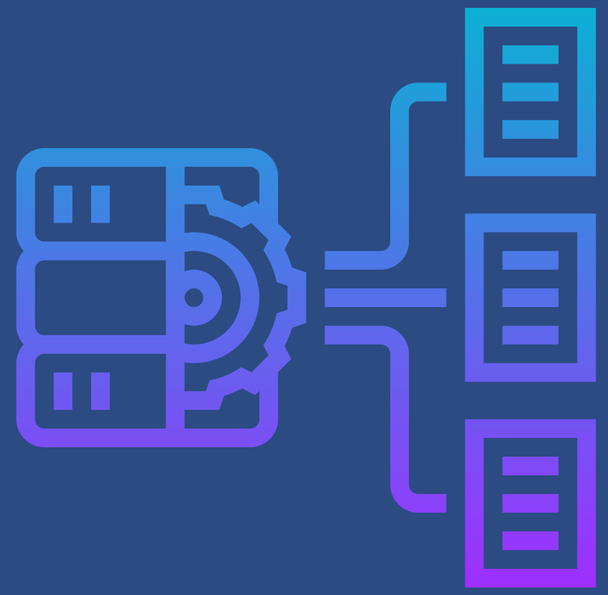
-Orders

-Pay Options

-Customer

-Storage

-Products



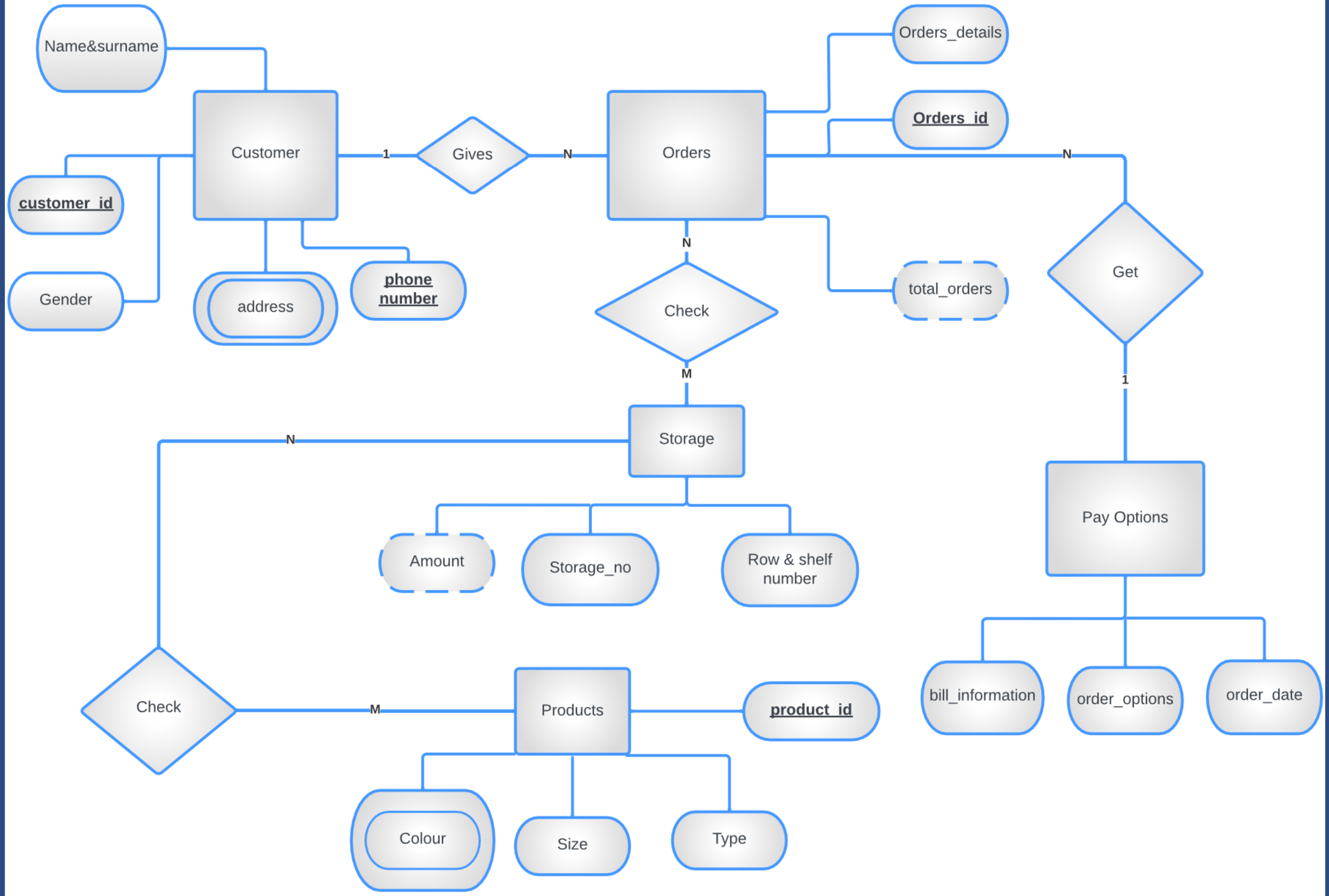
The Database has 4 relationship

-Gives

-Get

-Check

-Check


Name&surname
Customer
customer_id
Gender
address
phone number
1
Gives
N
Orders
Orders_details
Orders_id
total_orders
N
N
Check
M
Storage
Amount
Storage_no
Row & shelf number
N
Check
M
Products
product_id
Colour
Size
Type
Get
1
Pay Options
bill_information
order_options
order_date

Tables

-Orders

-Pay Options

-Customer

-Storage

-Products

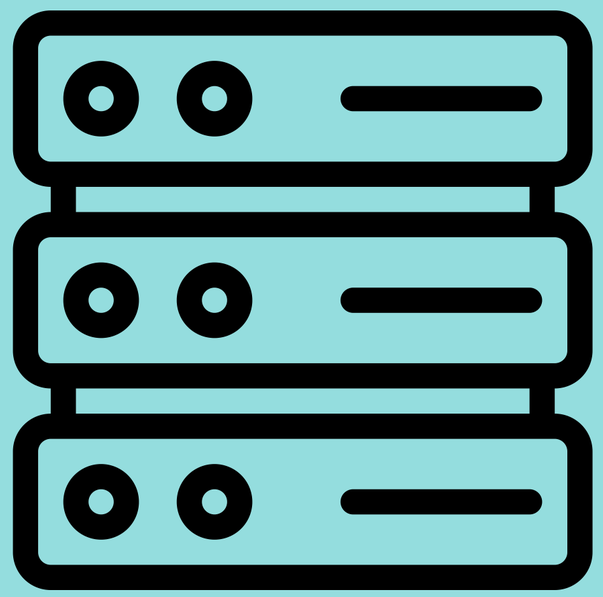
```
CREATE TABLE check1 (
  Storage_no varchar(45) NOT NULL,
  product_id varchar(45) NOT NULL,
  PRIMARY KEY (Storage_no,product_id),
  UNIQUE KEY Storage_no_UNIQUE (Storage_no),
  UNIQUE KEY product_id_UNIQUE (product_id),
  CONSTRAINT product_id FOREIGN KEY (product_id) REFERENCES products (product_id),
  CONSTRAINT Storage_no FOREIGN KEY (Storage_no) REFERENCES storage (Storage_no)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

CREATE TABLE check2 (
  Orders_id varchar(45) NOT NULL,
  Storage_no varchar(45) NOT NULL,
  PRIMARY KEY (Orders_id,Storage_no),
  UNIQUE KEY Storage_no_UNIQUE (Storage_no),
  UNIQUE KEY Orders_id_UNIQUE (Orders_id),
  CONSTRAINT pk2orders_id FOREIGN KEY (Orders_id) REFERENCES orders (Orders_id),
  CONSTRAINT pk2storage_no FOREIGN KEY (Storage_no) REFERENCES storage (Storage_no)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

CREATE TABLE customer (
  Customer_id varchar(45) NOT NULL,
  name&surname varchar(45) NOT NULL,
  Gender varchar(45) NOT NULL,
  adress varchar(45) NOT NULL,
  phone_number varchar(45) NOT NULL,
  PRIMARY KEY (Customer_id),
  UNIQUE KEY Customer_id_UNIQUE (Customer_id),
  UNIQUE KEY phone_number_UNIQUE (phone_number),
  CONSTRAINT check_Gender CHECK (((Gender = _utf8mb4'F') or (Gender = _utf8mb4'M')))
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

```sql
CREATE TABLE orders (
  Orders_id varchar(45) NOT NULL,
  Orders_details varchar(45) NOT NULL,
  total_orders varchar(45) NOT NULL,
  customer_id varchar(45) NOT NULL,
  bill_information varchar(45) NOT NULL,
  PRIMARY KEY (Orders_id),
  UNIQUE KEY Orders_id_UNIQUE (Orders_id),
  KEY customer_id_idx (customer_id),
  KEY bill_information_idx (bill_information),
  CONSTRAINT bill_information FOREIGN KEY (bill_information) REFERENCES pay_options (bill_information),
  CONSTRAINT customer_id FOREIGN KEY (customer_id) REFERENCES customer (Customer_id),
  CONSTRAINT check_total_orders CHECK ((total_orders > 0))
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

CREATE TABLE pay_options (
  bill_information varchar(45) NOT NULL,
  order_options varchar(45) NOT NULL,
  order_date varchar(45) NOT NULL,
  orders_details varchar(45) NOT NULL,
  PRIMARY KEY (bill_information),
  UNIQUE KEY bill_information_UNIQUE (bill_information),
  CONSTRAINT check_order_options CHECK (((order_options = _utf8mb4'Credit Card') or (0 <> _utf8mb4'Cash')))
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```



```sql
CREATE TABLE products (
  product_id varchar(45) NOT NULL,
  Colour varchar(45) NOT NULL,
  Size varchar(45) NOT NULL,
  Type varchar(45) NOT NULL,
  PRIMARY KEY (product_id),
  UNIQUE KEY product_id_UNIQUE (product_id),
  CONSTRAINT check_Colour CHECK (((Colour = _utf8mb4'Black') or (0 <> _utf8mb4'White') or (0 <> _utf8mb4'Yellow') or (0 <> _utf8mb4'Blue'))),
  CONSTRAINT check_products CHECK (((Size = _utf8mb4'Small') or (0 <> _utf8mb4'Medium') or (0 <> _utf8mb4'Large'))),
  CONSTRAINT check_size CHECK (((Size = _utf8mb4'Small') or (0 <> _utf8mb4'Medium') or (0 <> _utf8mb4'Large')))
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

CREATE TABLE storage (
  Storage_no varchar(45) NOT NULL,
  Amount varchar(45) NOT NULL,
  Row&shelf number varchar(45) NOT NULL,
  PRIMARY KEY (Storage_no),
  UNIQUE KEY Storage_no_UNIQUE (Storage_no)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

Data Input

As seen in the example, data entry has been made for all tables.

```
LOCK TABLES `orders` WRITE;
/*!40000 ALTER TABLE `orders` DISABLE KEYS */;
INSERT INTO `orders` VALUES ('ORD21','On the way','1','YRK36','FTR1'),
('ORD22','Prepared','1','YRK12','FTR2'),('ORD23','On the way','4','YRK9','FTR3'),
('ORD24','Prepared','2','YRK25','FTR4'),('ORD25','Arrived','5','YRK23','FTR5'),
('ORD26','On the way','2','YRK31','FTR6'),('ORD27','Prepared','4','YRK5','FTR7'),
('ORD28','Arrived','1','YRK21','FTR8');
UNLOCK TABLES;
```

Some SQL Queries with Outputs

```
USE homeworkdatabase;
SELECT pay_options.orders_options, pay_options.orders_details, orders.orders_id
FROM pay_options
LEFT JOIN orders
on pay_options.bill_information = orders.bill_information
WHERE orders_options = "Cash"
and pay_options.orders_details like "Pre%"
order by orders.orders_id desc
```

orders_options	orders_details	orders_id
Cash	Prepared	ORD72
Cash	Prepared	ORD56
Cash	Prepared	ORD51
Cash	Prepared	ORD46
Cash	Prepared	ORD34
Cash	Prepared	ORD27
Cash	Prepared	ORD24

Some SQL Queries with Outputs

```
SELECT  size, colour, count(*) as "toplam miktar"
FROM homeworkdatabase.products
where colour = "Black"
group by size
HAVING size in ("Medium","Large")
```

	size	colour	toplam miktar
▶	Large	Black	12
	Medium	Black	3

Some SQL Queries with Outputs

```
SELECT sum(total_orders) as "Toplam Sipariş", max(total_orders) as "Tek Seferde En Çok Verilen Sipariş"
FROM homeworkdatabase.orders
```

	Toplam Sipariş	Tek Seferde En Çok Verilen Sipariş
▶	158	5

Teşekkürler!

Ömer Türk
Ahmet Erdem Cesur
Metin Vatansever
Mehmet Demirbilek