

# Datenbankdesign / Datenbankanalyse

## Inhalt

Aufgabe 1: Datenbankdesign .....	2
Xampp Control Panel.....	2
Erstellen einer neuen „Connection“ in MySql.....	2
ER-Modell Handyvergleich .....	2
Relationen: .....	2
Erstellen der Datenbank mit „Forward Engineering“ .....	3
Befüllen der Datenbank mit den Informationen aus dem CSV-File .....	4
Ort Table:.....	4
Kunde Table:.....	4
Handy Table:.....	4
Properties Table: .....	5
Aufgabe 2: Datenbank Analyse .....	5
1. Category .....	5
2. Customer .....	5
3. Manufacturer .....	5
4. Product .....	5
5. Weborder .....	5
6. Weborderline .....	5
Relationen / Beziehungen .....	5
Aufgabe / Query zur Webshop Datenbank .....	7
SQL Skript „lap_handyverleih“ .....	8
SQL Skript „lap_handyverleih“ Insert Into .....	10

## Aufgabe 1: Datenbankdesign

### Xampp Control Panel

Zu aller erst öffnen wir das Xampp Control Panel & starten den MySQL Server. (Siehe Abbild)

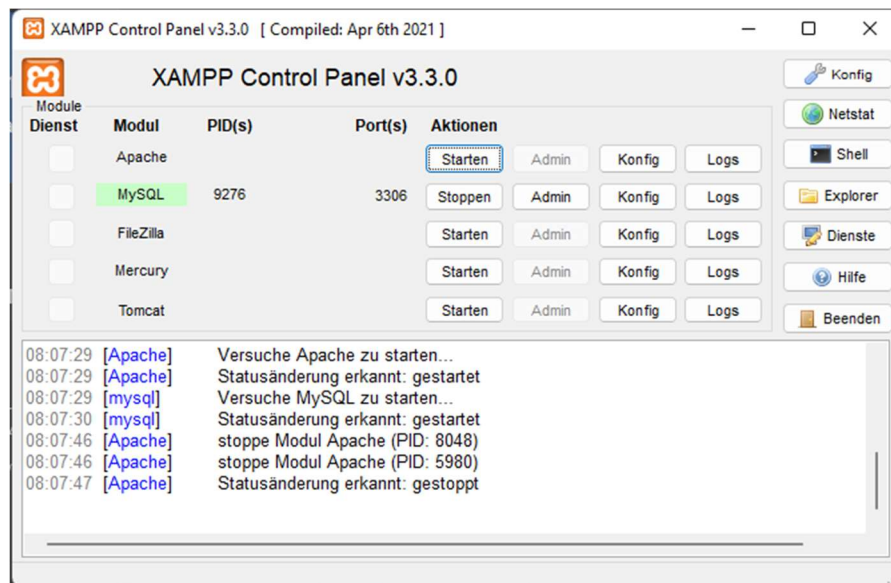


Abbildung 1: Xampp Control Panel

### Erstellen einer neuen „Connection“ in MySQL

Folgende Konfigurationen wurden für die MySQL Connection getätigt:

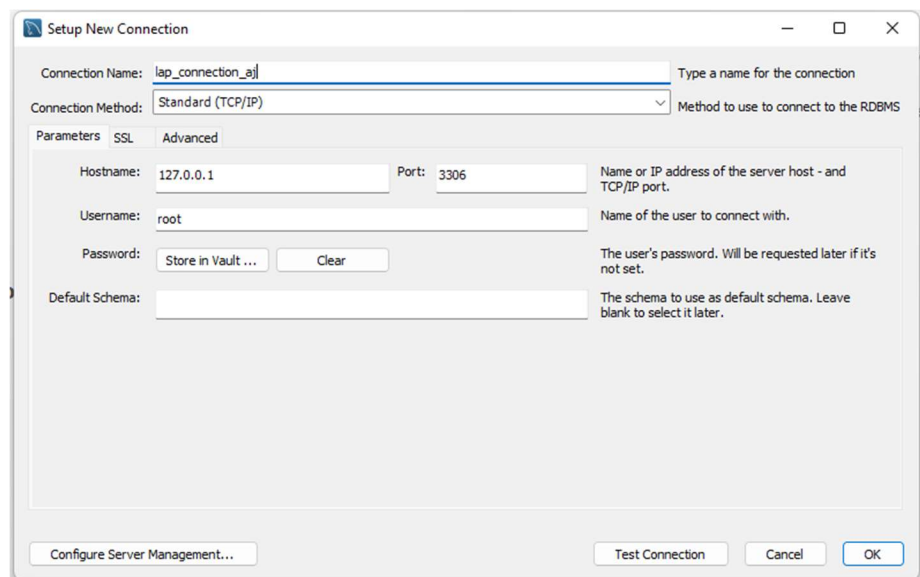


Abbildung 2: MySql Connection

### ER-Modell Handyvergleich

#### Relationen:

1. Ort & Kunde (1:N)
  - a. Ein Kunde, kann an einem Ort gemeldet sein.

- b. Ein Ort, kann mehrere Kunden haben.
- 2. Kunde & Handy (N:1)
  - a. Ein Kunde, kann mehrere Handys besitzen.
  - b. Ein Handy, kann nur einen Kunden gehören.
- 3. Handy & Properties (1:N)
  - a. Ein Handy, kann ein Set an Eigenschaften haben.
  - b. Eine Eigenschaft, kann auf mehrere Handys zutreffen.

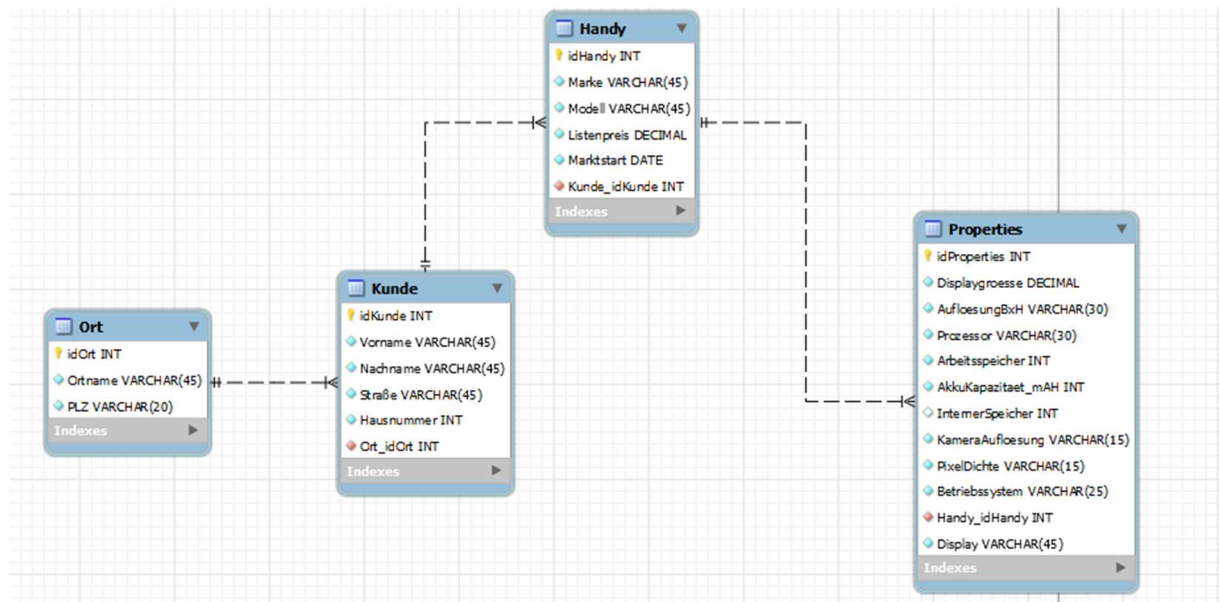


Abbildung 3: ER-Modell

### Erstellen der Datenbank mit „Forward Engineering“

1. Navigieren Sie auf „Database“ und wählen sie „Forward Engineering“
2. Wählen Sie die oben erstellte Connection aus. (lap\_connection\_aj).
3. Löschen Sie im SQL-Skript alle Einträge des Wortes „Visible“.



lap\_handyverleih\_aj  
.sql

Befüllen der Datenbank mit den Informationen aus dem CSV-File

Anbei liegt das SQL Skript mit allen Inserts in die Datenbank



insert\_into\_handyv  
erleih\_aj.sql

Ort Table:

Result Grid			
Filter Rows:			
	idOrt	Ortname	PLZ
▶	1		
	2	Schwarzach im Pongau	5620
	3	St.Johann im Pongau	5600
*	NULL	NULL	NULL

Abbildung 4: Ort Query Output

Kunde Table:

Result Grid						
Filter Rows:						
Edit: Export/Import						
	idKunde	Vorname	Nachname	Straße	Hausnummer	Ort_idOrt
▶	1					1
	2	Max	Mustermann	Hauptplatz		1
	3	Martina	Musterfrau	Markusplatz		2
*	NULL	NULL	NULL	NULL	NULL	NULL

Abbildung 5: Kunde Query Output

Handy Table:

Result Grid						
Filter Rows:						
Edit: Export/Import						
	idHandy	Marke	Modell	Listenpreis	Marktstart	Kunde_idKunde
▶	1	Xiaomi	11T Pro	650	2021-10-01	1
	2	Samsung	Galaxy S20 FE	633	2020-10-01	1
	3	Samsung	Galaxy S21	846	2021-01-01	1
	4	Apple	iPhone 12 mini	779	2020-11-01	1
	5	Apple	iPhone 12	876	2020-10-01	1
	6	Google	Pixel 6a	459	2022-07-01	1
	7	Apple	iPhone 13 Pro Max	1089	2021-09-01	1
	8	Apple	iPhone 13	770	2021-09-01	1
	9	Xiaomi	Redmi 10 5G	230	2022-06-01	1
	10	Xiaomi	12 Pro	1050	2022-03-01	1
	11	Vivo	Y72 5G	299	2021-07-01	1
*	NULL	NULL	NULL	NULL	NULL	NULL

Abbildung 6: Handy Query Output

## Properties Table:

	idProperties	Displaygroesse	AufloesungBxH	Prozessor	Arbeitsspeicher	AkkuKapazitaet_mAH	InternerSpeicher	KameraAufloesung	PixelDichte	Betriebssystem	Handy_idHa
1	1	6.67	1080x2400	Qualcomm Snapdragon 888	8	5000	128	12032x9024	395 ppi	Android 11	1
2	2	6.50	1080x2400	Samsung Exynos 990	6	4500	128	4272x2848	405 ppi	Android 10	2
3	3	6.20	1080x2400	Samsung Exynos 2100	8	4000	128	4272x2848	442 ppi	Android 11	3
4	4	5.40	1080x2340	Apple A14 Bionic	4	2227	256	4272x2848	477 ppi	iOS 14.1	4
5	5	6.10	1170x2532	Apple A14 Bionic	4	2775	256	4272x2848	457 ppi	iOS 14.1	5
6	6	6.10	1080x2400	Google Tensor	6	4410	128	4272x2848	431 ppi	Android 12	6
7	7	6.70	1284x2778	Apple A15 Bioni	6	4373	128	4272x2848	457 ppi	iOS 15	7
8	8	6.10	1170x2532	Apple A15	4	3240	128	4272x2848	457 ppi	iOS 15	8
9	9	6.58	1080x2408	MediaTek Dimensity 700	4	5000	64	8688x5792	401 ppi	Android 12	9
10	10	6.73	1440x3200	Snapdragon 8 Gen 1	12	4600	256	8688x5792	521 ppi	Android 12	10
11	11	6.58	1080x2408	MediaTek Dimensity 700	8	5000	128	9248x6936	401 ppi	Android 11	11

Abbildung 7: Properties Query Output

## Aufgabe 2: Datenbank Analyse

Datenbank: lap\_webshop

### 1. Category

Der Table Category enthält Kategorien, denen Produkte zugeordnet werden können  
Bsp: Ein Handy wäre der Technik Kategorie zugeordnet.

### 2. Customer

Der Table Customer enthält personen bezogene Daten wie z.B: (Vorname, Nachname, Adresse, etc...) vom Kunden.

### 3. Manufacturer

Der Table Manufacturer steht für den Hersteller. Hier wird der Name des Herstellers gespeichert und ihm Produkte zugeordnet.

Bsp: Das Produkt „Iphone12“ kommt von dem Hersteller „Apple“

### 4. Product

Der Table Product enthält alle im Webshop enthaltene Produkte. Dabei sind verschiedene Eigenschaften angegeben, wie z.B der Name des Produktes.

### 5. Weborder

Der Table Weborder enthält alle vom Kunden in Auftrag gegebenen Bestellungen und dementsprechend Bestelldetails wie z.B: die OrderID.

Bsp: Ein Kunde bestellt ein Produkt, Daten wie Adresse, Bestelldatum, etc... werden im Weborder festgehalten.

### 6. Weborderline

Der Table Weborderline enthält weitere Details zum Order wie z.B die Menge, Steuerrate, etc... des Produktes.

## Relationen / Beziehungen

### 1. Product & Manufacturer (1:N)

- Ein Product hat einen Manufacturer.
- Ein Manufacturer kann mehrere Products haben.

### 2. Product & Category (1:N)

- Ein Product hat eine Category.
- Eine Category kann mehrere Product haben.

3. Product & Weborder (1:N)
  - a. Ein Product kann in einem Weborder enthalten sein.
  - b. Ein Weborder kann mehrere Products enthalten.
4. Weborder & Weborderline (1:1)
  - a. Ein Weborder kann eine Weborderline, bzw Details zum Weborder enthalten.
  - b. Eine Weborderline kann Details zu einem Weborder enthalten.
5. Product & Weborderline (1:N)
  - a. Ein Product kann in einer Weborderline enthalten sein.
  - b. Eine Weborderline kann mehrere Products enthalten.

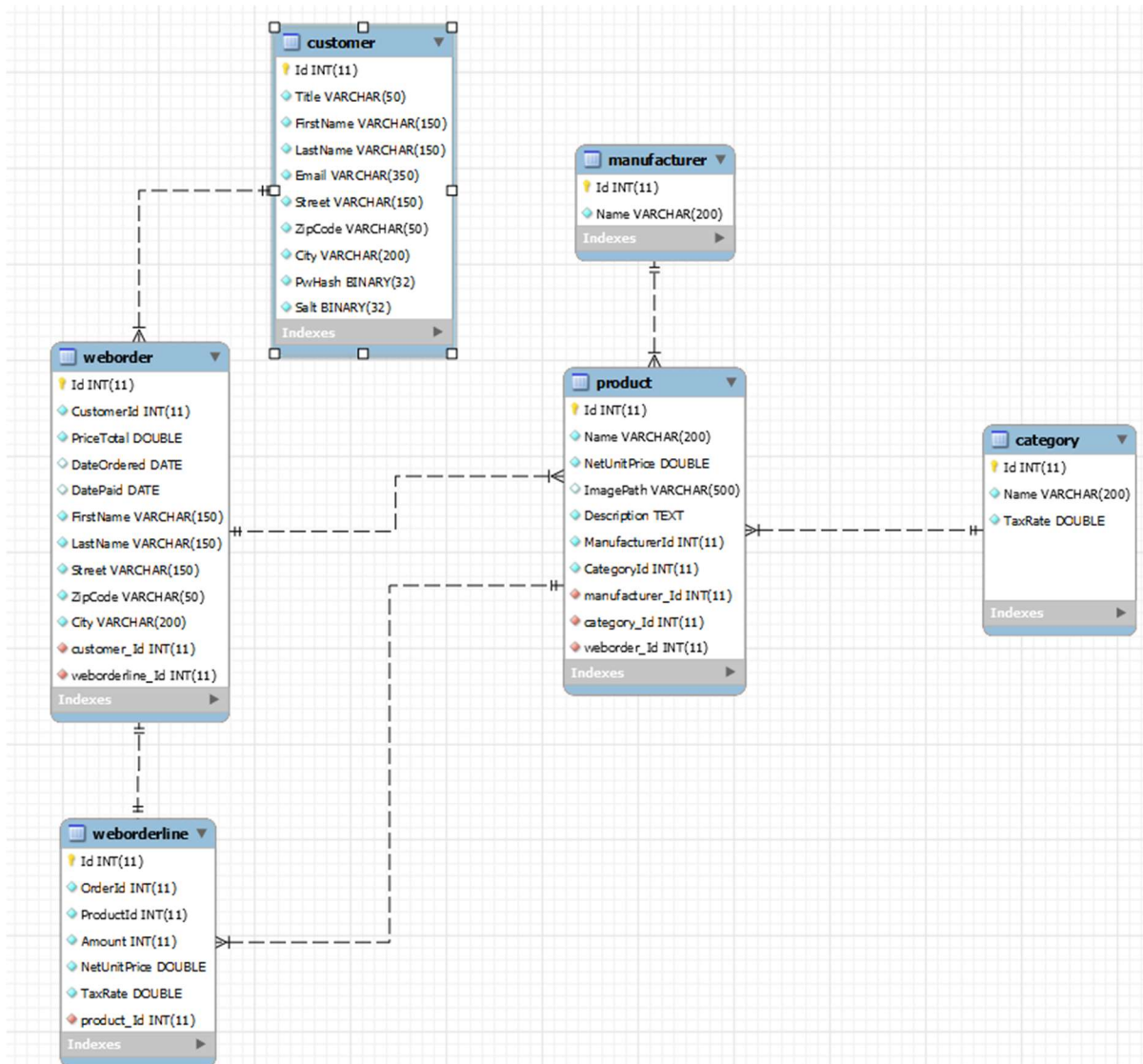


Abbildung 8: ER-Modell der lap\_webshop Datenbank

## Aufgabe / Query zur Webshop Datenbank

Wenn die Beziehungen gesetzt sind, muss man unter „Database“ den Punkt „Synchronize Model“ auswählen. Dabei wird wieder ein fertiges Skript angezeigt. Man beachte, dass man auch hier das Wort „Visible“ entfernen muss.

Warum muss man Visible entfernen?

- A. Visible wurde in MariaDB noch nicht eingeführt, in MySQL jedoch schon.

*Skript zum Updaten nach dem Setzen der Beziehungen:*



update\_ermodel\_ch  
anges\_aj.sql

Aufgabe: Query welche alle Hersteller auflistet mit der Anzahl der Produkte.

#Sql Query zur Aufgabe:

```
SELECT m.Id, m.Name, COUNT(p.Id) AS Produktanzahl  
FROM product as p  
INNER JOIN manufacturer as m  
ON p.ManufacturerId = m.Id  
GROUP BY m.Id, m.Name
```

	Id	Name	Produktanzahl
▶	1	Pegasus Spiele	11
	2	KOSMOS	6
	3	Wizards of the Coast	9
	4	Nintendo	11
	5	Sony	4
	6	Microsoft	3

Abbildung 9: Query Aufgabe Output



## SQL Skript „lap\_handyverleih“

```
-- MySQL Workbench Forward Engineering
--
-- Schema lap_handyvergleich
--
--
-- Schema lap_handyvergleich
--
CREATE SCHEMA IF NOT EXISTS `lap_handyvergleich` DEFAULT CHARACTER SET utf8 ;
USE `lap_handyvergleich` ;

--
-- Table `lap_handyvergleich`.`Ort`
--
CREATE TABLE IF NOT EXISTS `lap_handyvergleich`.`Ort` (
  `idOrt` INT NOT NULL AUTO_INCREMENT,
  `Ortname` VARCHAR(45) NOT NULL,
  `PLZ` VARCHAR(20) NOT NULL,
  PRIMARY KEY (`idOrt`),
  UNIQUE INDEX `idOrt_UNIQUE` (`idOrt` ASC) )
ENGINE = InnoDB;

--
-- Table `lap_handyvergleich`.`Kunde`
--
CREATE TABLE IF NOT EXISTS `lap_handyvergleich`.`Kunde` (
  `idKunde` INT NOT NULL AUTO_INCREMENT,
  `Vorname` VARCHAR(45) NOT NULL,
  `Nachname` VARCHAR(45) NOT NULL,
  `Straße` VARCHAR(45) NOT NULL,
  `Hausnummer` INT NOT NULL,
  `Ort_idOrt` INT NOT NULL,
  PRIMARY KEY (`idKunde`),
  UNIQUE INDEX `idKunde_UNIQUE` (`idKunde` ASC) ,
  INDEX `fk_Kunde_Ort_idx` (`Ort_idOrt` ASC) ,
  CONSTRAINT `fk_Kunde_Ort`
    FOREIGN KEY (`Ort_idOrt`)
    REFERENCES `lap_handyvergleich`.`Ort` (`idOrt`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```



```
ENGINE = InnoDB;
```

```
-- Table `lap_handyvergleich`.`Handy`
```

```
CREATE TABLE IF NOT EXISTS `lap_handyvergleich`.`Handy` (  
  `idHandy` INT NOT NULL AUTO_INCREMENT,  
  `Marke` VARCHAR(45) NOT NULL,  
  `Modell` VARCHAR(45) NOT NULL,  
  `Listenpreis` DECIMAL NOT NULL,  
  `Marktstart` DATE NOT NULL,  
  `Kunde_idKunde` INT NOT NULL,  
  PRIMARY KEY (`idHandy`),  
  UNIQUE INDEX `idHandy_UNIQUE` (`idHandy` ASC) ,  
  INDEX `fk_Handy_Kunde1_idx` (`Kunde_idKunde` ASC) ,  
  CONSTRAINT `fk_Handy_Kunde1`  
    FOREIGN KEY (`Kunde_idKunde`)  
    REFERENCES `lap_handyvergleich`.`Kunde` (`idKunde`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-- Table `lap_handyvergleich`.`Properties`
```

```
CREATE TABLE IF NOT EXISTS `lap_handyvergleich`.`Properties` (  
  `idProperties` INT NOT NULL AUTO_INCREMENT,  
  `Displaygroesse` DECIMAL NOT NULL,  
  `AufloesungBxH` VARCHAR(30) NOT NULL,  
  `Prozessor` VARCHAR(30) NOT NULL,  
  `Arbeitsspeicher` INT NOT NULL,  
  `AkkuKapazitaet_mAH` INT NOT NULL,  
  `InternerSpeicher` INT,  
  `KameraAufloesung` VARCHAR(15) NOT NULL,  
  `PixelDichte` VARCHAR(15) NOT NULL,  
  `Betriebssystem` VARCHAR(25) NOT NULL,  
  `Handy_idHandy` INT NOT NULL,  
  `Display` VARCHAR(45) NOT NULL,  
  `Gewicht` VARCHAR(20) NOT NULL,  
  PRIMARY KEY (`idProperties`),  
  UNIQUE INDEX `idEigenschaften_UNIQUE` (`idProperties` ASC) ,  
  INDEX `fk_Properties_Handy1_idx` (`Handy_idHandy` ASC) ,  
  CONSTRAINT `fk_Properties_Handy1`  
    FOREIGN KEY (`Handy_idHandy`)  
    REFERENCES `lap_handyvergleich`.`Handy` (`idHandy`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

## SQL Skript „lap\_handyverleih“ Insert Into

```
#Insert Into ORT  
INSERT INTO Ort (Ortname, PLZ)  
VALUES  
(  
(  
'Schwarzach im Pongau', 5620),  
'St.Johann im Pongau', 5600);  
  
#Insert Into Kunde  
INSERT INTO Kunde (Vorname, Nachname, Straße, Hausnummer, Ort_idOrt)  
VALUES  
(  
(  
'Max', 'Mustermann', 'Hauptplatz', 13, 1),  
'Martina', 'Musterfrau', 'Markusplatz', 12, 2);  
  
#Insert Into Handy  
INSERT INTO Handy  
(Marke, Modell, Listenpreis, Marktstart, Kunde_idKunde)  
VALUES  
(  
'Xiaomi', '11T Pro', 650.00, '2021-10-1', 1),  
'Samsung', 'Galaxy S20 FE', 633.00, '2020-10-1', 1),  
'Samsung', 'Galaxy S21', 846.00, '2021-1-1', 1),  
'Apple', 'iPhone 12 mini', 779.00, '2020-11-1', 1),  
'Apple', 'iPhone 12', 876.00, '2020-10-1', 1),  
'Google', 'Pixel 6a', 459.00, '2022-07-1', 1),  
'Apple', 'iPhone 13 Pro Max', 1089.00, '2021-09-1', 1),  
'Apple', 'iPhone 13', 770.00, '2021-09-1', 1),  
'Xiaomi', 'Redmi 10 5G', 230.00, '2022-06-1', 1),  
'Xiaomi', '12 Pro', 1050.00, '2022-03-1', 1),  
'Vivo', 'Y72 5G', 299.00, '2021-07-1', 1)  
  
#Insert Into Properties  
INSERT INTO `lap_handyvergleich`.`properties`  
(`Displaygroesse`, `AufloesungBxH`, `Prozessor`, `Arbeitsspeicher`,  
`AkkuKapazitaet_mAH`, `InternerSpeicher`,
```

```
`KameraAufloesung`, `PixelDichte`, `Betriebssystem`, `Display`,  
`Handy_idHandy`, `Gewicht`)
```

#### Values

```
(6.67, '1080x2400', 'Qualcomm Snapdragon 888', 8, 5000, '128GB', '12032x9024',  
'395 ppi', 'Android 11', 'AMOLED', 1, '204 g'),  
(6.50, '1080x2400', 'Samsung Exynos 990 ', 6, 4500, 128, '4272x2848', '405  
ppi', 'Android 10', 'Super-AMOLED', 2, '190 g' ),  
(6.20, '1080x2400', 'Samsung Exynos 2100', 8, 4000, 128, '4272x2848', '442  
ppi', 'Android 11', 'AMOLED', 3, '169 g'),  
(5.40, '1080x2340', 'Apple A14 Bionic', 4, 2227, 256, '4272x2848', '477 ppi',  
'iOS 14.1', 'OLED', 4, '133 g'),  
(6.10, '1170x2532', 'Apple A14 Bionic', 4, 2775, 256, '4272x2848', '457 ppi',  
'iOS 14.1', 'OLED', 5, '162 g'),  
(6.10, '1080x2400', 'Google Tensor ', 6, 4410, NULL, '4272x2848', '431 ppi',  
'Android 12', 'OLED', 6, '178 g'),  
(6.70, '1284x2778', 'Apple A15 Bioni', 6, 4373, 128, '4272x2848', '457 ppi',  
'iOS 15', 'AMOLED', 7, '238 g'),  
(6.10, '1170x2532', 'Apple A15', 4, 3240, 128, '4272x2848', '457 ppi', 'iOS  
15', 'AMOLED', 8, '140 g'),  
(6.58, '1080x2408', 'MediaTek Dimensity 700', 4, 5000, 64, '8688x5792', '401  
ppi', 'Android 12', 'LCD', 9, '190 g'),  
(6.73, '1440x3200', 'Snapdragon 8 Gen 1', 12, 4600, 256, '8688x5792', '521  
ppi', 'Android 12', 'OLED', 10, '205 g'),  
(6.58, '1080x2408', 'MediaTek Dimensity 700', 8, 5000, 128, '9248x6936', '401  
ppi', 'Android 11', 'LCD', 11, '193 g')
```