

Phase 2:

Aufgabe 3 Datenbank Grundgerüst

```
CREATE TABLE Dienstleistung (
```

```
    ServiceID INT PRIMARY KEY,
```

```
    Bezeichnung VARCHAR(50) CHECK (Bezeichnung IN ('Spa', 'Restaurant',  
'Unterhaltung')),
```

```
    Preis DECIMAL(5, 2) CHECK (Preis >= 0),
```

```
    Beschreibung CLOB
```

```
);
```

```
CREATE TABLE Zimmer (
```

```
    Zimmernummer INT PRIMARY KEY,
```

```
    PreisProNacht DECIMAL(5, 2) CHECK (PreisProNacht >= 0),
```

```
    Verfuegbarkeit CHAR(1) CHECK (Verfuegbarkeit IN ('Y', 'N')) NOT NULL,
```

```
    Kategorie VARCHAR(50) CHECK (Kategorie IN ('Einzelzimmer', 'Doppelzimmer', 'Suite'))
```

```
);
```

```
CREATE TABLE Reservierung (
```

```
    ReservierungsID INT PRIMARY KEY,
```

```
    Zimmernummer INT, -- Fremdschlüssel zur Tabelle Zimmer
```

```
    Gesamtsumme DECIMAL(5, 2) CHECK (Gesamtsumme >= 0),
```

```
    Buchungsstatus VARCHAR(50) NOT NULL,
```

```
    Check_out DATE NOT NULL,
```

```
    Check_in DATE NOT NULL,
```

```
    CONSTRAINT FK_Zimmer FOREIGN KEY (Zimmernummer) REFERENCES  
    Zimmer(Zimmernummer)
```

```
);
```

```
CREATE TABLE Gast (  
    GastID INT PRIMARY KEY,  
    ReservierungsID INT, -- Fremdschlüssel zur Tabelle Reservierung  
    Vorname VARCHAR(50) NOT NULL,  
    Nachname VARCHAR(50) NOT NULL,  
    Geburtsdatum DATE NOT NULL,  
    Telefonnummer VARCHAR(20),  
    Email VARCHAR(100) NOT NULL,  
    Adresse VARCHAR(100) NOT NULL,  
    CONSTRAINT FK_Reservierung FOREIGN KEY (ReservierungsID) REFERENCES  
    Reservierung(ReservierungsID)  
);
```

```
CREATE TABLE Rechnung (  
    RechnungsID INT PRIMARY KEY,  
    GastID INT, -- Fremdschlüssel zur Tabelle Gast  
    Zahlungsart VARCHAR(50) NOT NULL,  
    Zahlungsdatum DATE NOT NULL,  
    Betrag DECIMAL(10, 2) CHECK (Betrag >= 0),  
    CONSTRAINT FK_Gast_Rechnung FOREIGN KEY (GastID) REFERENCES Gast(GastID)  
);
```

```
CREATE TABLE Feedback (  
    FeedbackID INT PRIMARY KEY,  
    GastID INT, -- Fremdschlüssel zur Tabelle Gast  
    Bewertung INT CHECK (Bewertung >= 1 AND Bewertung <= 5),  
    Datum DATE NOT NULL,  
    Kommentar CLOB,  
    CONSTRAINT FK_Gast FOREIGN KEY (GastID) REFERENCES Gast(GastID)
```

);

CREATE TABLE Personal (

MitarbeiterID *INT* PRIMARY KEY,
Gehalt *INT* CHECK (Gehalt >= 0),
Arbeitszeiten *VARCHAR*(50) NOT NULL,
Position *VARCHAR*(50) NOT NULL,
Vorname *VARCHAR*(50) NOT NULL,
Nachname *VARCHAR*(50) NOT NULL

);

CREATE TABLE Spa (

ServiceID *INT*, -- Fremdschlüssel zur Tabelle Dienstleistung
Oeffnungszeiten *VARCHAR*(50) NOT NULL,
Ausstattung *CLOB*,
Kapazitaet *INT* CHECK (Kapazitaet >= 0),
CONSTRAINT FK_Spa_Dienstleistung FOREIGN KEY (ServiceID) REFERENCES
Dienstleistung(ServiceID)

);

CREATE TABLE Unterhaltung (

ServiceID *INT*, -- Fremdschlüssel zur Tabelle Dienstleistung
Uhrzeit *VARCHAR*(50) NOT NULL,
Dauer *INT* CHECK (Dauer >= 0),
Datum *DATE* NOT NULL,
Art *VARCHAR*(50) CHECK (Art IN ('Konzert', 'Theater', 'Film')),
Beschreibung *CLOB*,
CONSTRAINT FK_Unterhaltung_Dienstleistung FOREIGN KEY (ServiceID)
REFERENCES Dienstleistung(ServiceID)

);

CREATE TABLE Restaurant (

ServiceID *INT*, -- Fremdschlüssel zur Tabelle Dienstleistung

Sitze *INT* CHECK (Sitze >= 0),

Oeffnungszeiten *VARCHAR*(50) NOT NULL,

Speisen *CLOB*,

Getraenke *CLOB*,

CONSTRAINT FK_Restaurant_Dienstleistung FOREIGN KEY (ServiceID) REFERENCES
Dienstleistung(ServiceID)

);

COMMIT;

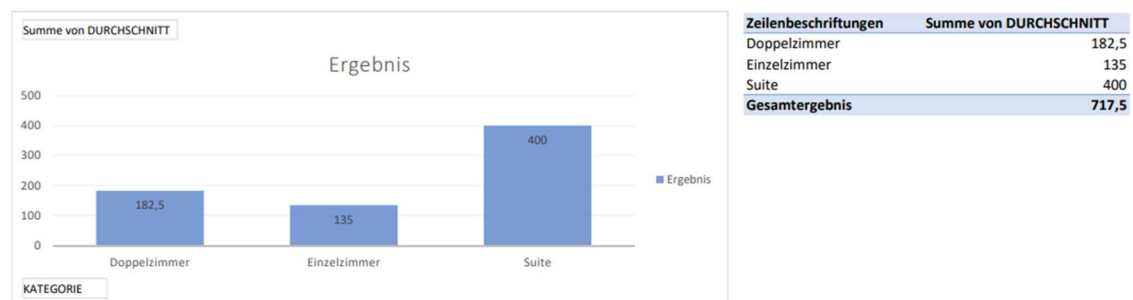
Phase 3:

Aufgabe 1

Bericht und Auswertung:

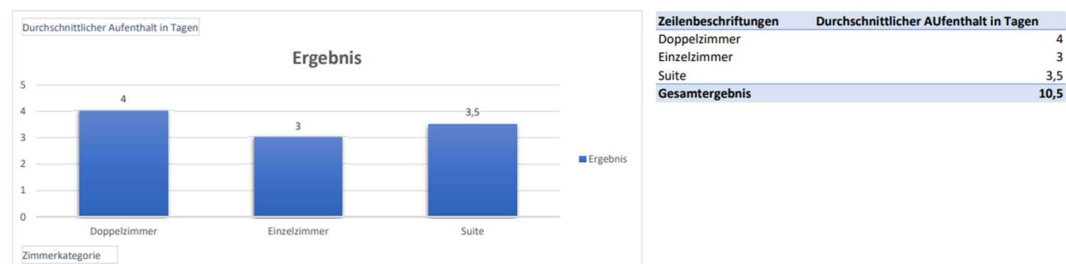
1. Durchschnittlicher Preis pro Nacht je Zimmerkategorie

```
SELECT kategorie, AVG(preispronacht) AS durchschnitt
FROM zimmer
GROUP BY kategorie;
```



2. Durchschnittlicher Aufenthalt in Tagen je Zimmerkategorie

```
SELECT z.kategorie, AVG(r.check_out - r.check_in) AS durchschnittlicher_aufenthalt
FROM reservierung r
JOIN zimmer z ON r.zimmernummer = z.zimmernummer
WHERE r.check_in IS NOT NULL AND r.check_out IS NOT NULL
GROUP BY z.kategorie;
```



Aufgabe 2 : Trigger

--Trigger zur Berechnung des Gesamtpreises einer Reservierung

CREATE OR REPLACE TRIGGER BerechneGesamtsumme

BEFORE INSERT OR UPDATE ON Reservierung

FOR EACH ROW

DECLARE

v_preis *DECIMAL*(10, 2);

BEGIN

-- Hole den Preis pro Nacht aus der Zimmer-Tabelle

SELECT PreisProNacht

INTO v_preis

FROM Zimmer

WHERE Zimmernummer = :NEW.Zimmernummer;

-- Berechne die Gesamtsumme basierend auf der Anzahl der Nächte

:NEW.Gesamtsumme := v_preis * (TRUNC(:NEW.Check_out) - TRUNC(:NEW.Check_in));

-- Falls keine Gültigkeit für Check_in oder Check_out, gebe Fehlermeldung zurück

IF :NEW.Check_out <= :NEW.Check_in THEN

RAISE_APPLICATION_ERROR(-20001, 'Das Check-out-Datum muss nach dem
Check-in-Datum liegen.');

END IF;

END;

/

-- Beispiel: Aktualisieren einer bestehenden Reservierung

UPDATE Reservierung

SET Check_in = TO_DATE('2025-01-03', 'YYYY-MM-DD'), Check_out = TO_DATE('2025-01-10', 'YYYY-MM-DD')

WHERE ReservierungsID = 1;

Aufgabe 3 : Testdaten

--Daten in Zimmer Tabelle einfügen

```
INSERT INTO Zimmer VALUES (101, 120.00, 'Y', 'Einzelzimmer');  
INSERT INTO Zimmer VALUES (102, 150.00, 'Y', 'Doppelzimmer');  
INSERT INTO Zimmer VALUES (103, 300.00, 'Y', 'Suite');  
INSERT INTO Zimmer VALUES (104, 200.00, 'N', 'Doppelzimmer');  
INSERT INTO Zimmer VALUES (105, 400.00, 'N', 'Suite');  
INSERT INTO Zimmer VALUES (106, 180.00, 'Y', 'Einzelzimmer');  
INSERT INTO Zimmer VALUES (107, 250.00, 'Y', 'Doppelzimmer');  
INSERT INTO Zimmer VALUES (108, 500.00, 'N', 'Suite');  
INSERT INTO Zimmer VALUES (109, 100.00, 'Y', 'Einzelzimmer');  
INSERT INTO Zimmer VALUES (110, 130.00, 'Y', 'Doppelzimmer');
```

--Daten in Dienstleistung Tabelle einfügen

```
INSERT INTO Dienstleistung VALUES (1, 'Spa', 50.00, 'Wellnessbereich mit Sauna und Pool');  
INSERT INTO Dienstleistung VALUES (2, 'Restaurant', 0.00, 'Gehobene Küche mit regionalen Spezialitäten');  
INSERT INTO Dienstleistung VALUES (3, 'Unterhaltung', 20.00, 'Live-Konzert jeden Samstagabend');  
INSERT INTO Dienstleistung VALUES (4, 'Spa', 70.00, 'Premium-Spa mit Massagen');  
INSERT INTO Dienstleistung VALUES (5, 'Unterhaltung', 15.00, 'Karaoke-Abend');  
INSERT INTO Dienstleistung VALUES (6, 'Restaurant', 0.00, 'Buffet-Frühstück');  
INSERT INTO Dienstleistung VALUES (7, 'Unterhaltung', 30.00, 'Comedy-Nacht');  
INSERT INTO Dienstleistung VALUES (8, 'Spa', 100.00, 'Privater Whirlpool-Zugang');  
INSERT INTO Dienstleistung VALUES (9, 'Restaurant', 0.00, 'Vegetarische Menüoptionen');  
INSERT INTO Dienstleistung VALUES (10, 'Unterhaltung', 10.00, 'Filmabend im hoteleigenen Kino');
```


--Daten in Reservierung Tabelle einfügen

```
INSERT INTO Reservierung VALUES (1, 101, 240.00, 'Bestätigt', TO_DATE('2025-01-01', 'YYYY-MM-DD'), TO_DATE('2025-01-03', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (2, 102, 450.00, 'Storniert', TO_DATE('2025-01-05', 'YYYY-MM-DD'), TO_DATE('2025-01-10', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (3, 103, 600.00, 'Bestätigt', TO_DATE('2025-01-02', 'YYYY-MM-DD'), TO_DATE('2025-01-04', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (4, 104, 1000.00, 'Storniert', TO_DATE('2025-01-15', 'YYYY-MM-DD'), TO_DATE('2025-01-20', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (5, 105, 2000.00, 'Bestätigt', TO_DATE('2025-01-10', 'YYYY-MM-DD'), TO_DATE('2025-01-15', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (6, 106, 540.00, 'Bestätigt', TO_DATE('2025-01-06', 'YYYY-MM-DD'), TO_DATE('2025-01-11', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (7, 107, 750.00, 'Bestätigt', TO_DATE('2025-01-08', 'YYYY-MM-DD'), TO_DATE('2025-01-12', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (8, 108, 2500.00, 'Storniert', TO_DATE('2025-01-09', 'YYYY-MM-DD'), TO_DATE('2025-01-14', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (9, 109, 300.00, 'Bestätigt', TO_DATE('2025-01-03', 'YYYY-MM-DD'), TO_DATE('2025-01-05', 'YYYY-MM-DD'));
```

```
INSERT INTO Reservierung VALUES (10, 110, 260.00, 'Bestätigt', TO_DATE('2025-01-07', 'YYYY-MM-DD'), TO_DATE('2025-01-09', 'YYYY-MM-DD'));
```

--Daten in Gast Tabelle einfügen

```
INSERT INTO Gast VALUES (1, 1, 'Max', 'Mustermann', TO_DATE('1985-03-15', 'YYYY-MM-DD'), '1234567890', 'max@hotel.de', 'Musterstraße 1, 12345 Musterstadt');
```

```
INSERT INTO Gast VALUES (2, 2, 'Erika', 'Musterfrau', TO_DATE('1990-07-22', 'YYYY-MM-DD'), '0987654321', 'erika@hotel.de', 'Beispielweg 2, 54321 Beispielstadt');
```

```
INSERT INTO Gast VALUES (3, 3, 'Anna', 'Schmidt', TO_DATE('1988-11-02', 'YYYY-MM-DD'), '1231231234', 'anna@hotel.de', 'Hotelstraße 5, 11223 Hotelstadt');
```

```
INSERT INTO Gast VALUES (4, 4, 'Peter', 'Müller', TO_DATE('1982-05-08', 'YYYY-MM-DD'), '0981234567', 'peter@hotel.de', 'Gaststraße 6, 65432 Gaststadt');
```

```
INSERT INTO Gast VALUES (5, 5, 'Julia', 'Schneider', TO_DATE('1995-06-18', 'YYYY-MM-DD'), '5678901234', 'julia@hotel.de', 'Urlaubsweg 9, 87654 Urlaubsstadt');
```

```
INSERT INTO Gast VALUES (6, 6, 'Paul', 'Fischer', TO_DATE('1992-12-25', 'YYYY-MM-DD'), '2345678901', 'paul@hotel.de', 'Zimmerplatz 8, 23456 Zimmerstadt');
```

```
INSERT INTO Gast VALUES (7, 7, 'Laura', 'Weber', TO_DATE('1989-09-11', 'YYYY-MM-DD'), '3456789012', 'laura@hotel.de', 'Straße 7, 12321 Irgendstadt');
```

```
INSERT INTO Gast VALUES (8, 8, 'Tom', 'Hofmann', TO_DATE('1987-01-01', 'YYYY-MM-DD'), '4567890123', 'tom@hotel.de', 'Stadtstraße 1, 54321 Anderstadt');
```

```
INSERT INTO Gast VALUES (9, 9, 'Lisa', 'Bauer', TO_DATE('1993-03-14', 'YYYY-MM-DD'), '5678901234', 'lisa@hotel.de', 'Urlaubsplatz 10, 98765 Ferienstadt');
```

```
INSERT INTO Gast VALUES (10, 10, 'David', 'Lehmann', TO_DATE('1991-08-19', 'YYYY-MM-DD'), '6789012345', 'david@hotel.de', 'Traumstraße 11, 13579 Traumstadt');
```

--Daten in Rechnung Tabelle einfügen

```
INSERT INTO Rechnung VALUES (1, 1, 'Kreditkarte', TO_DATE('2025-01-03', 'YYYY-MM-DD'), 240.00);
```

```
INSERT INTO Rechnung VALUES (2, 3, 'PayPal', TO_DATE('2025-01-04', 'YYYY-MM-DD'), 600.00);
```

```
INSERT INTO Rechnung VALUES (3, 5, 'Kreditkarte', TO_DATE('2025-01-15', 'YYYY-MM-DD'), 2000.00);
```

```
INSERT INTO Rechnung VALUES (4, 6, 'Bargeld', TO_DATE('2025-01-11', 'YYYY-MM-DD'), 540.00);
```

```
INSERT INTO Rechnung VALUES (5, 7, 'Kreditkarte', TO_DATE('2025-01-12', 'YYYY-MM-DD'), 750.00);
```

```
INSERT INTO Rechnung VALUES (6, 9, 'Überweisung', TO_DATE('2025-01-05', 'YYYY-MM-DD'), 300.00);
```

```
INSERT INTO Rechnung VALUES (7, 10, 'PayPal', TO_DATE('2025-01-09', 'YYYY-MM-DD'), 260.00);
```

--Daten in Feedback Tabelle einfügen

```
INSERT INTO Feedback VALUES (1, 1, 5, TO_DATE('2025-01-03', 'YYYY-MM-DD'), 'Wundervoller Aufenthalt!');
```

```
INSERT INTO Feedback VALUES (2, 3, 4, TO_DATE('2025-01-04', 'YYYY-MM-DD'), 'Sehr gutes Zimmer, Frühstück könnte besser sein.');
```

```
INSERT INTO Feedback VALUES (3, 5, 5, TO_DATE('2025-01-15', 'YYYY-MM-DD'),  
'Perfekte Erfahrung.');
```

```
INSERT INTO Feedback VALUES (4, 6, 3, TO_DATE('2025-01-11', 'YYYY-MM-DD'),  
'Sauber, aber etwas laut.');
```

```
INSERT INTO Feedback VALUES (5, 7, 4, TO_DATE('2025-01-12', 'YYYY-MM-DD'), 'Sehr  
freundliches Personal.');
```

```
INSERT INTO Feedback VALUES (6, 9, 2, TO_DATE('2025-01-05', 'YYYY-MM-DD'),  
'Zimmer war nicht sauber.');
```

```
COMMIT;
```

Quellen: Testdaten wurden mit einer KI generiert [ChatGPT]