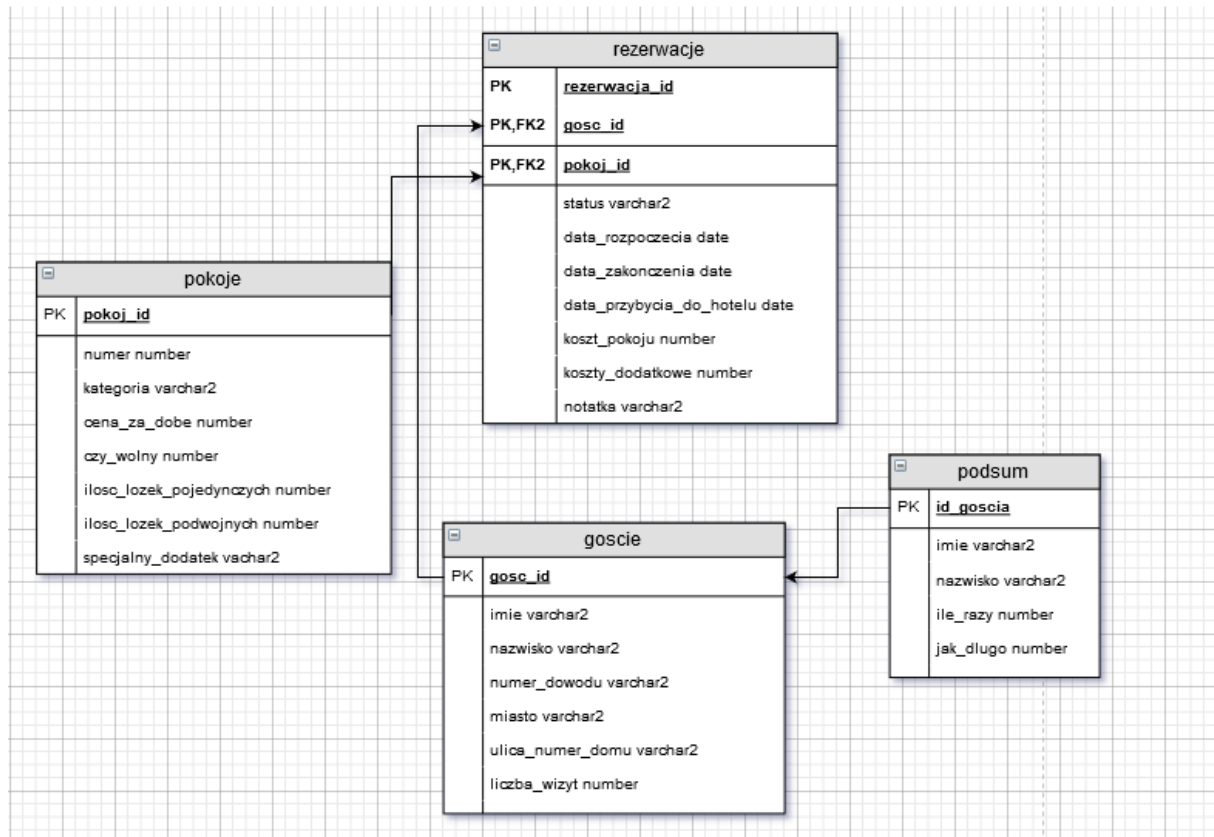


# TAYSIR HOTEL



## 1. add\_guest

```
CREATE OR REPLACE PROCEDURE add_guest (guest_id number, fname
varchar2, lname varchar2, id_number varchar2, city
varchar2, street_house_number varchar2)
```

### Parameters:

guest\_id number,  
 fname varchar2,  
 lname varchar2,  
 id\_number varchar2,  
 city varchar2,  
 ulica\_nr\_domu varchar2

The procedure used to add a guest to the database.

## 2. write\_free\_rooms

```
CREATE OR REPLACE PROCEDURE write_free_rooms (d1 DATE, d2 DATE)
```

### Parameters:

d1 DATE,  
d2 DATE

**Procedure for listing empty rooms between dates d\ and dY.**

## 3. find\_guest\_by\_id

```
CREATE OR REPLACE FUNCTION find_guest_by_id(fid_number  
varchar2)
```

### Parameters:

fid\_number **varchar2**

**An helpful function to find the guest by ID number returning the guest id.**

## 4. add\_reservations

```
CREATE OR REPLACE PROCEDURE add_reservations(fid_number  
varchar2, id_room number, status varchar2, start_date  
DATE, end_date DATE, note varchar2)
```

### Parameters:

add\_reservations **varchar2**,  
fid\_number **number**,  
status **varchar2**,  
start\_date **DATE**,  
end\_date **DATE**,  
note **varchar2**

**The procedure used to add reservations to the database.**

hotel\_request

```
CREATE OR REPLACE PROCEDURE
hotel_request(fid_number varchar2,
arrival_date date)
```

**Parameters:**

fid\_number varchar2,  
arrival\_date date

**Procedure for entering the date of the guest's arrival at the hotel.**

## 6. cancel\_reservations

```
CREATE OR REPLACE PROCEDURE cancel_reservations(id_number
varchar2)
```

**Parameters:**

id\_number varchar2

**Procedure for deleting reservations from the database.**

## 7. diff\_date

```
CREATE OR REPLACE FUNCTION diff_date(guest_id\ number)
```

**Parameters:**

guest\_id\ number

**This function returns the number of days the guest stayed at the hotel.**

## 8. bill

```
CREATE OR REPLACE PROCEDURE bill(fid_number varchar2,
additional_cost number)
```

**Parameters:**

fid\_number varchar2,  
additional\_cost numer

**Procedure for calculating the invoice for the guest.**

## 9. late\_person

```
CREATE OR REPLACE PROCEDURE late_person
```

**Procedure for returning all persons who have made reservations and have not yet reported to the hotel.**

10.       staying\_at\_hotel

```
CREATE OR REPLACE PROCEDURE staying_at_hotel
```

**Procedure for inserting data into the Summary table.**

11.       delete\_old\_reservations

```
CREATE OR REPLACE PROCEDURE delete_old_reservations
```

**Procedure removing all reservations older than 0 years.**

12.       is\_the\_date\_correct

```
CREATE OR REPLACE TRIGGER is_the_date_correct
```

**A trigger that monitors the correctness of the date entered.**

13.       field\_update

```
CREATE OR REPLACE TRIGGER field_update
```

**Trigger updating fields: how many times and how long in the Summary table**

14.       writing\_out\_free\_vip\_rooms

```
CREATE OR REPLACE PROCEDURE writing_out_free_vip_rooms(d1 DATE,  
requirement varchar2)
```

**Parameters**

d1 **DATE**,

requirement **varchar2**

**An auxiliary procedure listing available rooms on a given date, which have the designation vip.**

15.       vip\_reservation

```
CREATE OR REPLACE PROCEDURE vip_reservation(day date,  
requirement varchar2, number_single_beds number,  
number_double_beds number, id_number varchar2)
```

**Parameters:**

day **date**,  
requirement **varchar2**,  
number\_single\_beds **number**,  
number\_double\_beds **number**,  
id\_number **varchar2**

**Procedure for booking a vip room for one night.**