

Normalization

Heap

- Room

Room	Type	Amenities	ADA Accessible	Standard Occupancy	Maximum Occupancy	Base Price	Extra Person
201	Double	Microwave, Jacuzzi	No	2	4	\$199.99	\$10
202	Double	Refrigerator	Yes	2	4	\$174.99	\$10

- Guest

Name	Address	City	State	Zip	Phone
Your Name	Your Address	City	State	Zip	Phone
Mack Simmer	379 Old Shore Street	Council Bluffs	IA	51501	(291) 553-0508

- Reservation

Room Number	Name	Adults	Children	Start Date	End Date	Total Room Cost
308	Mack Simmer	1	0	2/2/2023	2/4/2023	\$299.98
203	Bettyann Seery	2	1	2/5/2023	2/10/2023	\$999.95

First Normal Form (1NF)

To achieve 1NF, the table must satisfy the following conditions:

- There is no top-to-bottom ordering to the rows.
 - There is no left-to-right ordering to the columns.
 - Every row can be uniquely identified.
 - Every row/column intersection (field) contains only one value.
-
- Room: Amenities (Microwave, Refrigerator) | *Every row/column intersection (field) contains only one value*
 - Guest: Primary Key/Surrogate Key | *Every row can be uniquely identified*
 - Reservation: Primary Key/Surrogate Key | *Every row can be uniquely identified*

Room	Type	Amenities	ADA Accessible	Standard Occupancy	Maximum Occupancy	Base Price	Extra Person
201	Double	Microwave	No	2	4	\$199.99	\$10
201	Double	Jacuzzi	No	2	4	\$199.99	\$10

GuestID	Name	Address	City	State	Zip	Phone
1	Mack Simmer	379 Old Shore Street	Council Bluffs	IA	51501	(291) 553-0508

ReservationID	Room Number	Name	Adults	Children	Start Date	End Date	Total Room Cost
1	308	Mack Simmer	1	0	2/2/2023	2/4/2023	\$299.98

Second Normal Form (2NF)

In academic terms, a table is in 2NF if and only if it is in 1NF and every non-primary-key column is functionally dependent on the entire primary key but not functionally dependent on any proper subset of the primary key.

In plain English, you must already be in 1NF and then all of the columns except the primary key need to be strictly related to each other. Additionally, we should not repeat data across rows.

- **Room**

Room has *One-to-Many relationship* with Type, Amenity, Occupancy, BasePrice, and ExtraPersonPrice

RoomNumber	TypeID	AmenityID	ADA Accessible	OccupancyID	BasePriceID	ExtraPersonPriceID
201	2	5	No	2	3	1
202	2	3	Yes	2	2	1

AmenityID	AmenityName
1	Microwave
2	Jacuzzi
3	Refrigerator
4	Oven
5	Microwave, Jacuzzi
6	Microwave, Refrigerator
7	Microwave, Jacuzzi, Refrigerator
8	Microwave, Refrigerator, Oven

TypeID	Type
1	Single
2	Double
3	Suite

OccupancyID	StandardOccupancy	MaximumOccupancy
1	2	2
2	2	4
3	3	8

BasePriceID	Price
1	\$149
2	\$174.99
3	\$199.99
4	\$399.99

ExtraPersonPriceID	Price
1	\$10
2	\$20

- Guest (I didn't make another table with City, State, and Zip bc there are 41,683 zip codes in USA and I don't think it is worth it to store that big fixed-size table yet, maybe later when there are many traffic - clients)

GuestID	FirstName	LastName	Address	City	State	Zip	Phone
1	Mack	Simmer	379 Old Shore Street	Council Bluffs	IA	51501	(291) 553-0508
2	Bettyann	Seery	750 Wintergreen Dr.	Wasilla	AK	99654	(478) 277-9632

- Reservation

Reservation has **Many-to-Many (One-to-Many?)** relationship with Guest and Room

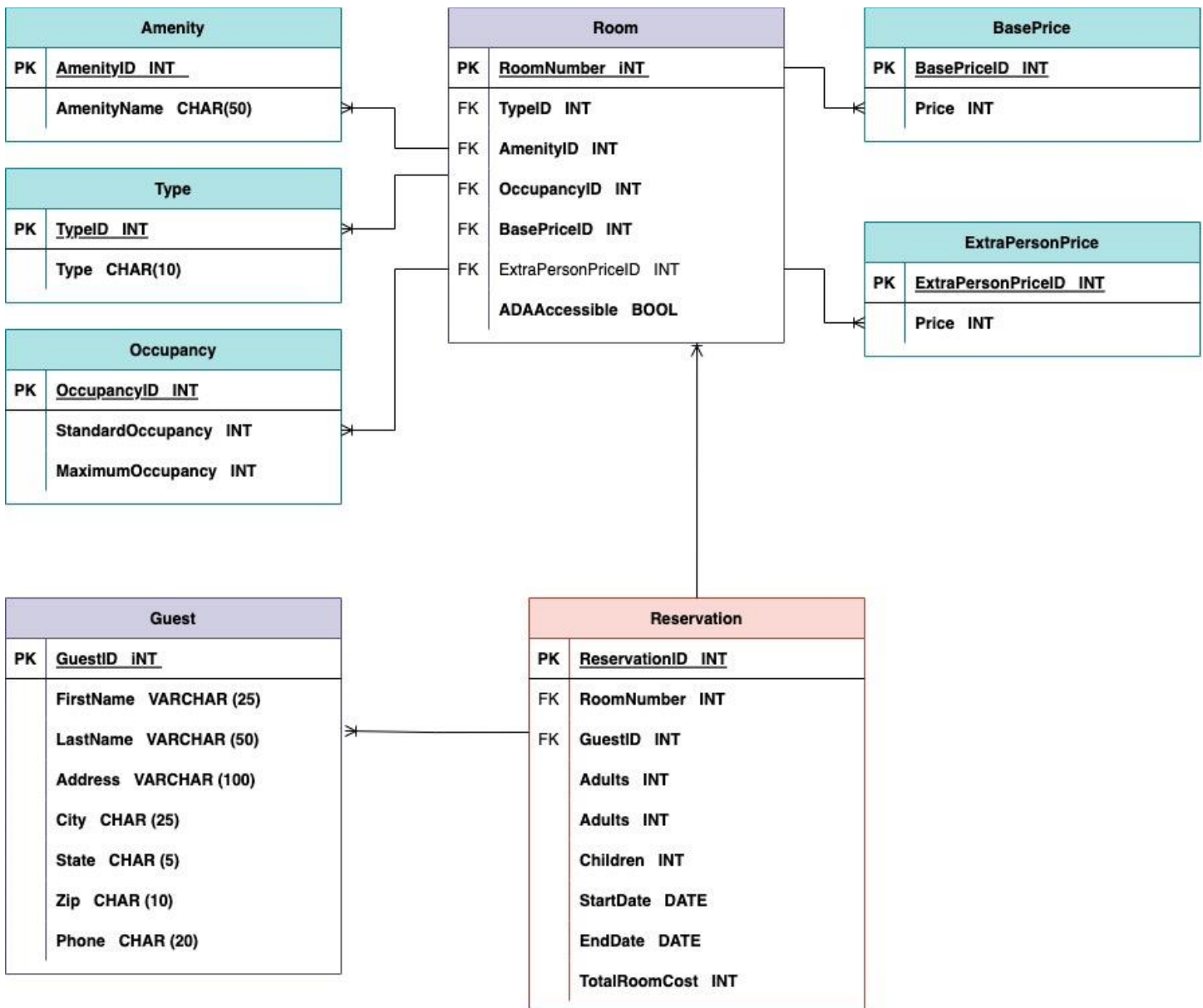
ReservationID	RoomNumber	GuestID	Adults	Children	Start Date	End Date	Total Room Cost
1	308	1	1	0	2/2/2023	2/4/2023	\$299.98
2	203	2	2	1	2/5/2023	2/10/2023	\$999.95

- Bridge Table (Many-to-Many)

- ReservationRoom
- ReservationGuest

Ver.1

- Total **8** tables
 - **Room** has 5 tables with *One-to-Many* relationship
 - **Reservation** has *One-to-Many* relationship with 2 tables - **Room** and **Guest**
- Confused what relationship to use between Reservation, Room and Guest
- One-to-Many or Many-to-Many?
- Confused where to put Foreign Key
- Confused which table should have three lines branching out (a chicken foot)



Ver.2

- Total **10** tables (8 tables and 2 bridge tables)
- 3 main tables - Room, Guest, Reservation
- **Room** has *One-to-Many relationship* with 5 tables - Amenity, Type, Occupancy, BasePrice, ExtraPersonPrice
- **Reservation** has *Many-to-Many relationship* with 2 tables - **Room** and **Guest**

