**Travel agency Tracking Database Normalization**

1st Normal Form:

* Each attribute should contain atomic value.
* A column should contain value from the same domain.
* Each column should have a unique name.
* No ordering to rows and columns.
* No duplicate rows.

2ND Normal Form

* It must be 1st Normal Form.
* No Patial dependency in the relation (Partial dependency occurs when the left-hand side of a candidate key points non-prime attributes).

3rd Normal Form

* It must be 2nd Normal Form.
* No transitive dependency for non-prime attributes (To be non-transitive and 3NF at least one of these must be true: Either the left-hand side of functional dependency is super key or the right-hand side points to a prime attribute).

BCNF

* if it is 3NF.
* For each functional dependency there must be a super key.

Preserving Dependency Decomposition

* is another property of decomposed relational database schema D in which each functional dependency X -> Y specified in F either appeared directly in one of the relation schemas Ri in the decomposed D or could be inferred from the dependencies that appear in some Ri.

Lossless Join Decomposition

* Lossless join decomposition is a decomposition of a relation R into relations R1, R2 such that if we perform a natural join of relation R1 and R2, it will return the original relation R. This is effective in removing redundancies from databases while preserving the original data.

**Database Relations**

R:{

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |
| 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 |  |

}

FD {1 -> 2 to 10, 10-> 11 to 21, 22 ->23 to 30, 1 32 -> 31, 32 -> 33 to 37, 38->32, 39 to 48, 40 -> 41 to 43, 44 -> 45 to 48, 50,-> 1 10 22 32 38 51}

**Personal Details**

R:{

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |

}

FD :{ 1 -> 2 to 10} Highest Normal Form(BCNF) CK:{1}

CREATE TABLE `Personal Data` (

`PK\_PersonalData` INT PK,

`FirstName` NVARCHAR(20),

`MiddleName` NVARCHAR(20) PK,

`LastName` NVARCHAR(20),

`BirthDate` NVARCHAR(20),

`PassportNumber` NVARCHAR(20),

`PassportExpriration` NVARCHAR(20),

`Email` Type,

`ContactNumber` Type,

`Address` Type

);

Accommodation Details

R:{

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |

}

FD:{11-> 11 to 21} Highest Normal Form(BCNF) CK:{11}

CREATE TABLE `Accomodation Details` (

`AccomodationRef` NVARCHAR(20) PK,

`ConfirmationNO` NVARCHAR(20),

`HotelName` NVARCHAR(20),

`HotelRate` NVARCHAR(20),

`NIghts` INT,

`RoomsCount` INT,

`HotelPhone` NVARCHAR(15),

`Room Description` NVARCHAR(20),

`CheckIn` Datetime,

`CheckOut` Datetime,

`PaymentRate` NVARCHAR(20),

`AdditionalDescription` NVARCHAR(MAX)

);

Tours Details

R:{

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 22 |  |  |

}

FD:{22 ->23 to 30} Highest Normal Form(BCNF) CK:{22}

CREATE TABLE `Tours Details` (

`TourRef` NVARCHAR(20) PK,

`ConfirmationNO` NVARCHAR(20),

`Country` NVARCHAR(20),

`Area` NVARCHAR(20),

`DateTime` DateTime,

`Email` INT,

`mode` NVARCHAR(15),

`Modeoftour` NVARCHAR(20),

`additionalDescription` Datetime

);

Guest Details

R:{

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 32 | 33 |  |  |  |  |  |  |  |  |

}

FD:{1 32 -> 33} Highest Normal Form(BCNF) CK:{22}

CREATE TABLE `Guest Details` (

`FK\_PersonalInformation` INT FK,

`Addons` NVARCHAR(50),

`FllightBookReference` NVARCHAR(20) FK,

FOREIGN KEY (`FllightBookReference`) REFERENCES `Trip Details`(`FlightBookReference`),

FOREIGN KEY (`FK\_PersonalInformation`) REFERENCES `Personal Data`(`PK\_PersonalData`) );

Trip Details

R:{

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 32 | 33 | 34 | 35 | 36 | 37 |  |  |  |  |  |

}

FD:{ 32 -> 33 to 37} Highest Normal Form{BCNF} CK:{32}

CREATE TABLE `Trip Details` (

`FlightBookReference` NVARCHAR(20) PK,

`Booking Date` Date,

`Mode` NVARCHAR(20),

`Modeoftravel` NVARCHAR(20),

`Embarkation` NVARCHAR(20),

`Disembarkation` NVARCHAR(20)

);

Flight Details

R:{

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 |

}

FD:{ 38->32, 39 to 48, 40 -> 41 to 43, 44 -> 45 to 48 } Highest Normal Form{2NF} CK:{38}

CREATE TABLE `Fliigh Details` (

`FlightID` INT PK,

`Flight No.` NVARCHAR(10),

`Airline` NVARCHAR(10),

`FlightBookReference` NVARCHAR(20) FK,

`DepartCountry` NVARCHAR(5),

`DepartAirport` NVARCHAR(15),

`DepartTERMINAL` NVARCHAR(15),

`DepartDate` DateTime,

`ArriCountry` NVARCHAR(5),

`ArriAirport` NVARCHAR(15),

`ArrivTERMINAL` NVARCHAR(15),

`ArrivDate` DateTime,

FOREIGN KEY (`FlightBookReference`) REFERENCES `Trip Details`(`FlightBookReference`)

);

Booking Details

R:{

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 50 | 1 | 10 | 22 | 32 | 38 | 51 |  |  |  |  |

}

FD:{ 50,-> 1 10 22 32 38 51 } Highest Normal Form{BCNF} CK:{50}

CREATE TABLE `Booking Details` (

`PK\_Book` INT PK,

`FK\_PersonalData` INT FK,

`AccomodationRef` NVARCHAR(20) PK,

`BookingDate` NVARCHAR(20),

`Status` NVARCHAR(20),

`FlightBookReference` NVARCHAR(20),

`TourRef` NVARCHAR(20),

FOREIGN KEY (`FlightBookReference`) REFERENCES `Trip Details`(`FlightBookReference`),

FOREIGN KEY (`TourRef`) REFERENCES `Tours Details`(`TourRef`),

FOREIGN KEY (`FK\_PersonalData`) REFERENCES `Personal Data`(`PK\_PersonalData`),

FOREIGN KEY (`AccomodationRef`) REFERENCES `Accomodation Details`(`AccomodationRef`)

);

A screenshot of a computer

Description automatically generated