NAME: ARVIND YOGESH ATTUR RAMESH
Id : 29825660

Monash Cabin Normalization

This document provides detailed explanation on normalization between contract cleaner and contract history.

DOCUMENT- A: Monash Cabins Cabin Cleaning Running Sheet:

UNF:

{res_id, res_name, res_address {cabin_no, std_cabin_clean_time, clean_date_time, contr_no, contr_fname, contr_lname, contr_rate, actual_cabin_clean_time, cleaning_charge}}

1NF:

To move from UNF to 1NF, identify unique identifier for the repeating group and remove along with primary key of the main relation, therefore,

RESORT*{**res_id(PK)**,* res_name, res_address*}*

CABIN_CLEANER *{res_id(FK)* ***, contr_no(PK), clean_date_time(PK)****, cabin_no, std_cabin_clean_time , contr_fname, contr_lname, contr_rate, actual_cabin_clean_time, cleaning_charge}*

contr_no -> contr_fname, contr_lname, contr_rate PARTIAL

cabin_no -> std_cabin_clean_time TRANSITIVE

2NF:

2NF is possible only if it satisfies 1NF and there should be no partial dependency.

Here {contr_fname, contr_lname, contr_rate} depends only on the **contr_no,** therefore here partial dependency exist because contractor number(contr_no) is the composite primary key.

RESORT*{**res_id(PK)**,* res_name, res_address*}*

CABIN_CLEANER *{res_id(FK)* ***, contr_no(PFK), clean_date_time(PK)****, cabin_no, std_cabin_clean_time , actual_cabin_clean_time, cleaning_charge}*

CLEANER_DETAIL {**contr_no(PK),** contr_fname, contr_lname, contr_rate }

3NF:

3NF is possible only if it satisfies 2NF and there should be no transitive dependency. In CABIN_CLEANER, std_cabin_clean_time depends on the cabin number (cabin_no) and resort id(res_id). As cabin number is non-key attribute , transitive dependency exists.

RESORT*{**res_id(PK)**,* res_name, res_address*}*

CLEANER_DETAIL{**contr_no(PK),** contr_fname, contr_lname, contr_rate }

CABIN_CLEANER *{res_id(FK)* ***, contr_no(PFK), clean_date_time(PK)****, cabin_no, actual_cabin_clean_time, cleaning_charge}*

CABIN{ ***cabin_no(PK), res_id(PFK),*** *std_cabin_clean_time }*

NAME: ARVIND YOGESH ATTUR RAMESH
Id : 29825660

Monash Cabins Cleaning Contractor Rates History

DOCUMENT B and C:

UNF:

{contr_no, contr_fname, contr_lname, contr_post_addr, contr_phone {contr_start_date, contr_end_date, contr_hourly_rate, contr_type}}

1NF:

Repeating group have to be removed and assign primary key from the main relation.

CLEANER_DETAIL {**contr_no(PK)**, contr_fname, contr_lname, contr_post_addr, contr_phone }

CONTRACTOR_HISTORY {**contr_no(PFK)**, **contr_start_date(PK)**, contr_end_date, contr_hourly_rate, contr_type}

2NF:

2NF satisfies only if it is 1NF and there should be no partial dependency.

Partial dependency exists only if non key attribute depends on any of the composite primary key. Here, all non-key attribute depends only on the primary key, therefore here no partial dependency exists.

CLEANER_DETAIL {**contr_no(PK)**, contr_fname, contr_lname, contr_post_addr, contr_phone }

CONTRACTOR_HISTORY {**contr_no(PFK)**, **contr_start_date(PK)**, contr_end_date, contr_hourly_rate, contr_type}

3NF:

3NF satisfies only if it is 2NF, and there should be no transitive dependency.

Transitive dependency exists only if any non-key attributes depends on other non-key attribute. Here, all non-key attributes depends only on the primary key, therefore no transitive dependency exists.

CLEANER_DETAIL {**contr_no(PK)**, contr_fname, contr_lname, contr_post_addr, contr_phone }

CONTRACTOR_HISTORY {**contr_no(PFK)**, **contr_start_date(PK)**, contr_end_date, contr_hourly_rate, contr_type}

Here 1NF is 2NF and 3NF:

FINAL Normalized Form:

RESORT*{**res_id(PK)**,* res_name, res_address*}*

CABIN_CLEANER *{res_id(FK)* ***, contr_no(PFK), clean_date_time(PK)****, cabin_no, actual_cabin_clean_time, cleaning_charge}*

CABIN{ ***cabin_no(PK), res_id(PFK),*** *std_cabin_clean_time }*

CLEANER_DETAIL {**contr_no(PK)**, contr_fname, contr_lname, contr_rate , contr_post_addr, contr_phone }

NAME: ARVIND YOGESH ATTUR RAMESH
Id : 29825660

CONTRACTOR_HISTORY {**contr_no(PFK)**, **contr_start_date(PK)**, contr_end_date, contr_hourly_rate, contr_type}