

Terceira forma normal (3NF)

Concepto: Debe estar en 2NF y no debe de haber dependencias transitivas entre atributos no clave.

Ejemplo:

Tabla employee_roles

EMPLOYEE_ID	JOB_CODE
E001	J01
E001	J02
E002	J02
E002	J03
E003	J01

Tabla employees

EMPLOYEE_ID	NAME	STATE_CODE
E001	Alice	26
E002	Bob	56
E003	Alice	56

Tabla jobs

JOB_CODE	JOB
J01	Chef
J02	Waster
J03	Bartender

Tabla states

STATE_CODE	HOME_STATE
26	Michigan
56	Wyoming

Nombre: Valery Sinar Trujillo

Profesor: Jhon Corredor

Institución:

Fecha:

Materia:

Curso:

Notas:

ACTIVIDAD

1. Escribir el concepto de normalización en base de datos, dar un ejemplo, escribir el concepto de las 3 formas de normalización en BD, escribir un ejemplo de cada una y escribirlo 3 veces

PRIMERO

- ¿Que es la normalización en BD?

La normalización es un proceso que organiza los datos en una base de datos para reducir la redundancia y mejorar la integridad de los datos. Este proceso se divide en tablas grandes en otras más pequeñas y establecer relaciones entre ellas.

Primera forma normal (1FN):

Concepto: Elimina los grupos repetitivos. Cada campo debe contener un solo valor, no listas.

Ejemplo:

EMPLOYEE_ID	NAME	JOB_CODE	JOB	STATE_CODE	HOME_STATE
E001	Alice	Chef	Chef	26	Michigan
E001	Alice	JO2	Waiter	26	Michigan
E002	Bob	JO2	Waiter	56	Wyoming
E002	Bob	JO3	Bartender	56	Wyoming
E003	Alice	JO1	Chef	56	Wyoming

Todas las entradas son atómicas y hay una clave primaria compuesta (employee-id, job-code) así que la tabla está en la 1FN.

Pero inclusive si solamente sabes el employee-id de alguien, entonces puedes determinar su name, home-state, y state-code (porque deberían ser la misma persona). Esto significa que name, home-state, y state-code son dependientes de employee-id. Así que, la tabla no está en 2NF. Deberíamos separarlos a una tabla distinta para hacerlo 2NF.

segunda forma normal (2NF)

Conceptos: debe estar en 1NF y todos los atributos no clave debe depender completamente de la clave primaria.

Ejemplo:

Tabla employee - roles

EMPLOYEE_ID	JOB_CODE
E001	J01
E001	J02
E002	J02
E002	J03
E003	J01

Tabla employees

EMPLOYEE_ID	NAME	STATE_CODE	HOME_STATE
E001	Alice	26	Michigan
E002	Bob	56	Wyoming
E003	Alice	56	Wyoming

Tabla jobs

JOB_CODE	JOB
J01	Chef
J02	Waiter
J03	Bartender

home-state ahora es dependiente de state-code. Así que, si sabes el state-code, entonces encontrar el de home-state. para llevar esto a un paso más allá, deberíamos separarlos otra vez a una tabla distinta para hacerlo 3NF

Segundo

- ¿Que es la normalización en BD?

La normalización es un proceso que organiza los datos en una base de datos para reducir la redundancia y mejorar integridad de los datos. Este proceso se divide en tablas grandes en otras más pequeñas y establecer relaciones entre ellas.

Primera forma normal (1^{FN}):

Concepto: Elimina los grupos repetitivos. Cada campo debe contener un solo valor, no listas.

Ejemplo:

EMPLOYEE_ID	NAME	JOB_CODE	JOB	STATE_CODE	HOME_STATE
E001	Alice	J01	Chef	26	Michigan
E001	Alice	J02	Waiter	26	Michigan
E002	Bob	J02	Waiter	56	Wyoming
E002	Bob	J03	Bartender	56	Wyoming
E003	Alice	J01	Chef	56	Wyoming

Todas las encontradas son atómicas y hay una clave primaria compuesta (employee_id, job-code) así que la tabla está en la 1^{FN}.

pero inclusive si solamente sabes el employee_id de alguien, entonces puedes determinar su name, home-state, y state-code (porque deberían ser la misma persona). Esto significa que name, home-state, y state-code son dependientes de employee_id. Así que, la tabla no está en 2^{FN}. Debemos separarlos a una tabla distinta para hacerlo 2^{FN}.

Segunda forma normal (2NF)

Concepto: Debe estar en 1NF y todos los atributos no clave debe depender completamente de la clave primaria

Ejemplo:

Tabla employee_roles

EMPLOYEE_ID	JOB_CODE
E001	J01
E001	J02
E002	J02
E002	J03
E003	J01

Tabla employees

EMPLOYEE_ID	NAME	STATE_CODE	HOME_STATE
E001	Alice	26	Michigan
E002	Bob	56	Wyoming
E003	Alice	56	Wyoming

Tabla jobs

JOB_CODE	JOB
J01	Chef
J02	Waiter
J03	Bartender

home_state ahora es dependiente de state_code. Así que, si sabes es state_code, entonces encontrar el de home_state.

Para llevar esto a un paso más allá, deberíamos separarlos otra vez a una tabla distinta para hacerlo 1NF.

TerCera Normal (3NF)

Concepto: Debe estar en 2NF y no debe de haber dependencias transitivas entre atributos no clave.

Ejemplo:

Tabla employee_roles

EMPLOYEE_ID	JOB_CODE
E001	J01
E001	J02
E002	J02
E002	J03
E003	J01

Tabla employees

EMPLOYEE_ID	NAME	STATE_CODE
E001	Alice	26
E002	Bob	36
E003	Alice	36

Table jobs

JOB_CODE	JOB
J01	Chef
J02	Waiter
J03	Bartender

Tabla states

STATE_CODE	HOME_STATE
26	Michigan
56	wyoming

Ahora nuestra BD está en 3NF

Nombre: Valery Sinar Trujillo

Fecha:

Profesor: Jhon Corredor

Materia:

Institución:

Curso:

Nota:

TerCero

- ¿Qué es la normalización en BD?

La normalización es un proceso que organiza los datos en una base de datos para reducir la redundancia y mejorar la integridad de los datos. Este proceso se divide en tablas grandes en otras más pequeñas y establecer relaciones entre ellas.

Primera Forma Normal (1FN)

Concepto: Elimina los grupos repetitivos. Cada campo debe contener un solo valor, no listas.

Ejemplo:

EMPLOYEE_ID	NAME	JOB_CODE	STATE_CODE	STATE_CODE	HOME_STATE
E001	Alice	J01	Chef	26	Michigan
E001	Alice	J02	Waiter	26	Michigan
E002	Bob	J02	Waiter	56	Wyoming
E002	Bob	J03	Bartender	56	Wyoming
E003	Alice	J01	chef	56	Wyoming

Todas las entradas son atómicas y hay una clave primaria compuesta (employee-id, job-code) así que la tabla está en la 1FN.

Pero inclusive se solamente sabes que el employee-id de alguien, entonces puedes determinar su name, home-state, y state-code (porque deberían ser la misma persona). Esto significa que name, home-state y state-code son dependientes de employee-id. Así que, la tabla no está en 2NF. Deberíamos separarlos a una tabla distinta para hacerlo 2NF.

Segunda Forma Normal (2NF)

Concepto: Debe estar en 1NF y todos los atributos no clave debe depender completamente de la clave primaria.

Ejemplo:

Tabla employee_roles

employee_id	job_code
E001	J01
E001	J02
E002	J02
E002	J03
E003	J01

Tabla employees

employee_id	name	state_code	home_state
E001	Alice	26	Michigan
E002	Bob	56	Wyoming
E003	Alice	56	Wyoming

Tabla jobs

job_code	job
J01	Chef
J02	waiter
J03	Bartender

home-state ahora es dependiente de state-code. Así que, si sabes el state-code, entonces encontrar el de home-state.

para llevar esto a un paso más allá, deberíamos separarlos otra vez a una tabla distinta para hacerlo 3NF.

Tercera Forma Normal (3NF)

Concepto: Debe estar en 2NF y no debe de haber dependencias transitivas entre atributos no clave.

Ejemplo:

Tabla employee_roles

EMPLOYEE_ID	JOB_CODE
E001	J01
E001	J02
E002	J02
E002	J03
E003	J01

Tabla employees

EMPLOYEE_ID	NAME	STATE_CODE
E001	Alice	26
E002	Bob	56
E003	Alice	56

Tabla jobs

JOB_CODE	JOB
J01	Chef
J02	Waster
J03	Bar tender

Tabla states

STATE_CODE	HOME_STATE
26	Michigan
56	Wyoming

Ahora nuestra BD está en 3NF