

BASES DE DATOS SQL

Constraints (Restricciones)

Constraints

Las restricciones son usadas para mantener o garantizar la integridad de la información que se almacena en la base de datos

- Null / Not null
- Unique
- Primary Key
- Check
- References
- Default

Constraints

Definición de Constraints






- ***CREATE TABLE***
- ***ALTER TABLE***

Constraints

Primera restricción

- Restricción del dominio de valores (tipo de dato)

Create table prueba (uno int, dos float, tres varchar);

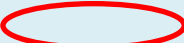
Valores	uno	dos	tres
Enteros			
Punto flotante			
Carateres			

Constraints – Null / Not Null

Null / Not Null

El valor especial NULL es parte de todos los dominios de cualquier tipo de datos.

ESTUDIANTE	ASIGNATURA	NOTA 1	NOTA 2	NOTA 3
ESTUDIANTE 1	ASIGNATURA 1	3,5	4,0	2,5
ESTUDIANTE 1	ASIGNATURA 3	2,0	4,5	4,0
ESTUDIANTE 1	ASIGNATURA 4	4,0		
ESTUDIANTE 1	ASIGNATURA 6	2,5	3,5	4,5
ESTUDIANTE 2	ASIGNATURA 1	3,0	4,0	3,5
ESTUDIANTE 2	ASIGNATURA 2		2,5	3,0
ESTUDIANTE 2	ASIGNATURA 5	4,5	2,0	2,0
ESTUDIANTE 3	ASIGNATURA 2	1,5	2,5	4,0
ESTUDIANTE 3	ASIGNATURA 5	3,0	3,0	3,0
ESTUDIANTE 4	ASIGNATURA 3	2,0	3,0	
ESTUDIANTE 4	ASIGNATURA 4	4,5	4,0	4,5
ESTUDIANTE 4	ASIGNATURA 5	3,0	4,0	5,0
ESTUDIANTE 5	ASIGNATURA 1	5,0	4,0	3,0
ESTUDIANTE 5	ASIGNATURA 2	2,5	4,5	3,5

 Valor NULL

Permitido: NULL

No permitido: NOT NULL

Constraints – Null / Not Null

Null / Not Null

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar not null,  
    tel_e bigint null,  
    dir_e varchar not null,  
    e_mail varchar not null,  
    fech_nac date not null);
```

Constraints – Null / Not Null

Null / Not Null (definición con alter table)

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date);
```

```
Alter table estudiantes alter column nom_e set not null,  
alter column dir_e set not null,  
alter column e_mail set not null,  
alter column fech_nac set not null;
```

Constraints – Null / Not Null

Null / Not Null

Algunos DBMS tiene por defecto que si no se expresa restricción de NULL o NOT NULL, el sistema asume que los campos serán NULL, como el caso de Oracle o PostgreSQL. Pero otros DBMS como SyBase asume que los campos serán NOT NULL.

Como recomendación, siempre se debe ser explícito en la definición de NULL o NOT NULL en cada campo que se defina en la base de datos.

Constraints – Null / Not Null

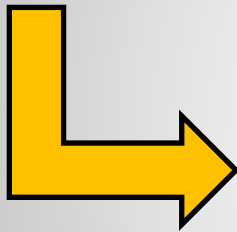


Null / Not Null

Como recomendación, siempre se debe ser explícito en la definición de NULL o NOT NULL en cada campo que se defina en la base de datos.

Constraints – Unique

Información

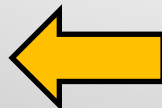


T1

---	---	---
---	---	---
---	---	---
---	---	---



SI



Rechazada

¿Ya existe?

NO



Constraints – Unique

Unique (definición a nivel de columna)

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar,  
    tel_e bigint unique,  
    dir_e varchar,  
    e_mail varchar constraint uq_email unique,  
    fech_nac date);
```

Constraints – Unique

Unique

Declaración de un campo como unique:

tel_e bigint unique

Declaración de un campo como unique, asignándole un nombre a la restricción:

e_mail varchar constraint uq_email unique

En este caso el nombre asignado a la restricción es *uq_email*

Constraints – Unique

Unique (definición a nivel de tabla)

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    unique (tel_e),  
    constraint uq_email unique (e-mail));
```

Constraints – Unique

Unique (definición con alter table)

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail,  
    fech_nac date,  
    cod_carr);
```

```
Alter table estudiantes add unique(tel_e),  
    add constraint uq_email unique (e-mail);
```

Constraints – Primary Key

Llave Candidata

Es un campo, o conjunto de campos, que permiten distinguir un registro, dentro de todo el conjunto de registros en la misma tabla.

Con la relación:

Estudiantes = (cod_e, nom_e, tel_e, dir_e, e-mail, fech_nac, cod_carr)

Identifique las llaves candidatas

Constraints – Primary Key

Con la relación:

Estudiantes = (cod_e, nom_e, tel_e, dir_e, e-mail, fech_nac, cod_carr)

Identifique las llaves candidatas:

- cod_e
- nom_e, fech_nac
- e_mail
- tel_e (si es un número de celular)
- nom_e, dir_e

Constraints – Primary Key

Llave Primaria

La llave primaria es la selección de la mejor llave candidata que identifique de forma única un registro dentro de todos los registros de la relación.

Criterios para escoger la mejor llave candidata:

- Debe ser lo mas corta posible
- Ocupar el menor espacio posible
- De ser posible, que tenga la menor cantidad de campos
- Que sea representativa en la relación

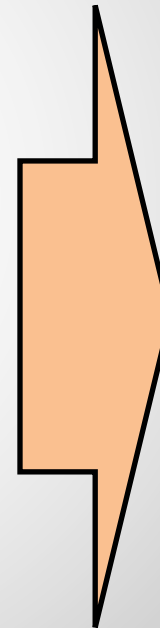
Estudiantes = (cod_e, nom_e, tel_e, dir_e, e_mail, fech_nac, cod_carr)

Constraints – Primary Key

Estudiantes

**UNIQUE
NOT NULL**

<u>Nom_e</u>	<u>Cod_e</u>	<u>Tel_e</u>	<u>Dir_e</u>	<u>Fech_nac</u>
ESTUDIANTE 1	20180111001	3111111111	Direccion 1	2000-01-01
ESTUDIANTE 2	20180111002	3112222222	Direccion 2	2001-02-02
ESTUDIANTE 3	20180111003	3113333333	Direccion 3	2002-10-11
ESTUDIANTE 4	20180111004	3114444444	Direccion 4	2003-03-01
ESTUDIANTE 5	20180111005	3115555555	Direccion 5	2004-03-02
ESTUDIANTE 6	20180121001		Direccion 6	2001-05-19
ESTUDIANTE 7	20180121002	3118888888	Direccion 7	2002-10-21
ESTUDIANTE 8	20180121003	3119999999	Direccion 9	2003-06-06
ESTUDIANTE 9	20180121004	3101111111	Direccion 10	2004-12-02
ESTUDIANTE 10	20180121005	3102222222	Direccion 11	2015-12-21
ESTUDIANTE 11	20180131001	3103333333	Direccion 12	2001-10-21
ESTUDIANTE 12	20180131002	3104444444	Direccion 13	2002-10-10
ESTUDIANTE 13	20180131003	3105555555	Direccion 14	2003-07-23
ESTUDIANTE 14	20180131004	3109999999	Direccion 15	2014-08-19



Solo una llave
primaria en la
tabla

Toda tabla debería tener una llave primaria

Constraints – Primary Key

Primary Key. (definición a nivel de columna)

```
Create table estudiantes (  
    cod_e int primary key,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date);
```

Constraints – Primary Key

Primary Key. (definición a nivel de columna)

```
Create table estudiantes (  
    cod_e int constraint pk_cod_e primary key,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date);
```

Constraints – Primary Key

Primary Key. (definición a nivel de tabla)

```
Create table estudiantes (  
    cod_e int ,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    primary key (cod_e));
```

Constraints – Primary Key

Primary Key. (definición a nivel de tabla)

```
Create table estudiantes (  
    cod_e int ,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    constraint pk_cod_e primary key (cod_e));
```

Constraints – Primary Key

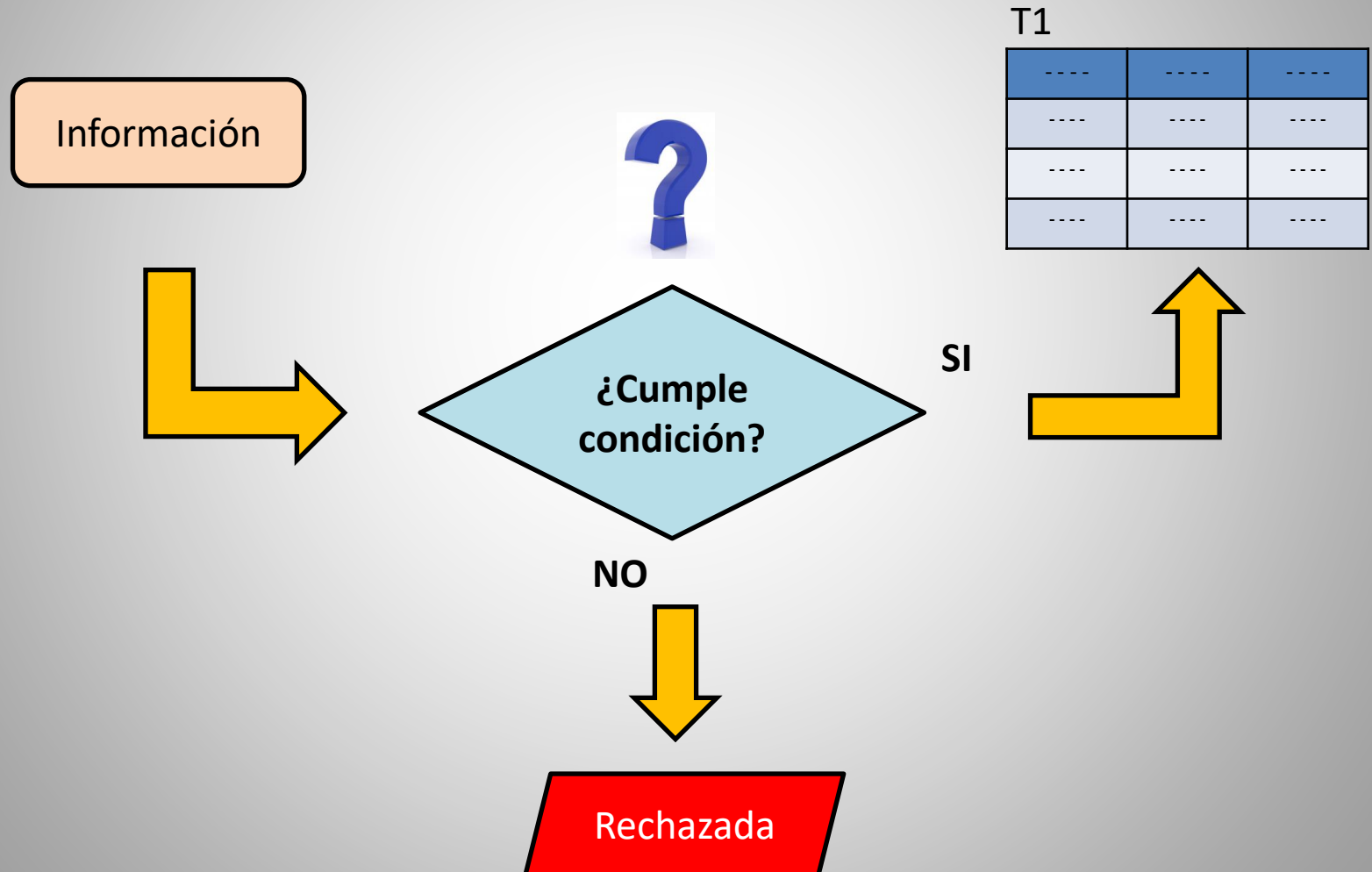
Primary Key. (definición con alter table)

```
Create table estudiantes (  
    cod_e int ,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date);
```

```
Alter table estudiantes add primary key (cod_e);
```

```
Alter table estudiantes add constraint pk_cod_e primary key (cod_e);
```

Constraints – Check



Constraints – Check

Notas

ESTUDIANTE	ASIGNATURA	NOTA 1	NOTA 2	NOTA 3
ESTUDIANTE 1	ASIGNATURA 1	3,5	4,0	2,5
ESTUDIANTE 1	ASIGNATURA 3	2,0	4,5	4,0
ESTUDIANTE 1	ASIGNATURA 4	4,0	3,0	4,0
ESTUDIANTE 1	ASIGNATURA 6	2,5	3,5	4,5
ESTUDIANTE 2	ASIGNATURA 1	3,0	4,0	3,5
ESTUDIANTE 2	ASIGNATURA 2		2,5	3,0
ESTUDIANTE 2	ASIGNATURA 5	4,5	2,0	2,0
ESTUDIANTE 3	ASIGNATURA 2	1,5	2,5	4,0
ESTUDIANTE 3	ASIGNATURA 5	3,0	3,0	3,0
ESTUDIANTE 4	ASIGNATURA 3	2,0	3,0	4,0
ESTUDIANTE 4	ASIGNATURA 4	4,5	4,0	4,5
ESTUDIANTE 4	ASIGNATURA 5	3,0	4,0	5,0
ESTUDIANTE 5	ASIGNATURA 1	5,0	4,0	3,0
ESTUDIANTE 5	ASIGNATURA 2	2,5	4,5	3,5

Valores de las notas
Entre 0.0 y 5.0

Constraints – Check

Check. (definición a nivel de columna)

```
Create table Notas (  
    estudiante varchar,  
    asignatura varchar,  
    nota1 numeric(2,1) check (nota1 between 0 and 5),  
    nota2 numeric(2,1) check (nota2 >= 0 and nota2 <= 5),  
    nota3 numeric(2,1) constraint ck_nota3  
        check (nota3 between 0 and 5));
```

Constraints – Check

Check. (definición a nivel de tabla)

```
Create table Notas (  
    estudiante varchar,  
    asignatura varchar,  
    nota1 numeric(2,1),  
    nota2 numeric(2,1),  
    nota3 numeric(2,1),  
    check (nota1 between 0 and 5),  
    check (nota2 >= 0 and nota2 <= 5),  
    constraint ck_nota3 check (nota3 between 0 and 5));
```

Constraints – Check

Check. (definición con alter table)

```
Create table Notas (  
    estudiante varchar,  
    asignatura varchar,  
    nota1 numeric(2,1),  
    nota2 numeric(2,1),  
    nota3 numeric(2,1));
```

```
Alter table Notas add check (nota1 between 0 and 5),  
    add check (nota2 >= 0 and nota2 <= 5),  
    add constraint ck_nota3 check (nota3 between 0 and 5);
```

Constraint – Reference

Información



T2

---	---	---
---	---	---
---	---	---
---	---	---



¿Está en la
tabla T2?

SI

NO

Rechazada

T1

---	---	---
---	---	---
---	---	---
---	---	---

Constraints – References

References. (definición a nivel de columna)

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    cod_carr int references carreras (cod_carr));
```

Constraints – References

References. (definición a nivel de columna)

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    cod_carr int constraint rf_carrera references carreras (cod_carr));
```

Constraints – References

References. (definición a nivel de tabla)

```
Create table estudiantes (  
    cod_e int ,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    cod_carr int,  
    constraint rf_carrera foreign key (cod_carr)  
        references carreras (cod_carr));
```


Constraints – References

References. (definición con alter table)

```
Create table estudiantes (  
    cod_e int primary key,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    cod_carr int);
```

```
Alter table estudiantes add constraint rf_carrera  
    foreign key (cod_carr) references carreras (cod_carr);
```

Constraints – Default

Default

Se puede asociar a un campo de una tabla un valor por defecto, en caso de no indicar un valor explícito, este valor por defecto se ingresará automáticamente al campo.

Constraints – Default

Default. (definición a nivel de columna)

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    cod_carr int,  
    estado varchar(8) default 'ACTIVO');
```

Constraints – Default

Default. (definición con alter table)

```
Create table estudiantes (  
    cod_e int,  
    nom_e varchar,  
    tel_e bigint,  
    dir_e varchar,  
    e_mail varchar,  
    fech_nac date,  
    cod_carr int,  
    estado varchar(8));
```

```
Alter table estudiantes alter column estado set default 'ACTIVO';
```

Dominios

Dominios

Un dominio es esencialmente un tipo de datos con unas restricciones definidas.

Es un objeto en la base de datos, luego se identifica con un nombre único, y puede ser usado en diferentes tablas.

Dominios

Dominios

Create domain nota

numeric (2,1)

constraint ck_nota check (value between 0 and 5)

null;

Create table Notas (

estudiante varchar,

asignatura varchar,

nota1 nota,

nota2 nota,

nota3 nota);

Bibliografía

Anón. s. f. «PostgreSQL: Documentation: 12: PostgreSQL 12.2 Documentation». (<https://www.postgresql.org/docs/12/index.html>).