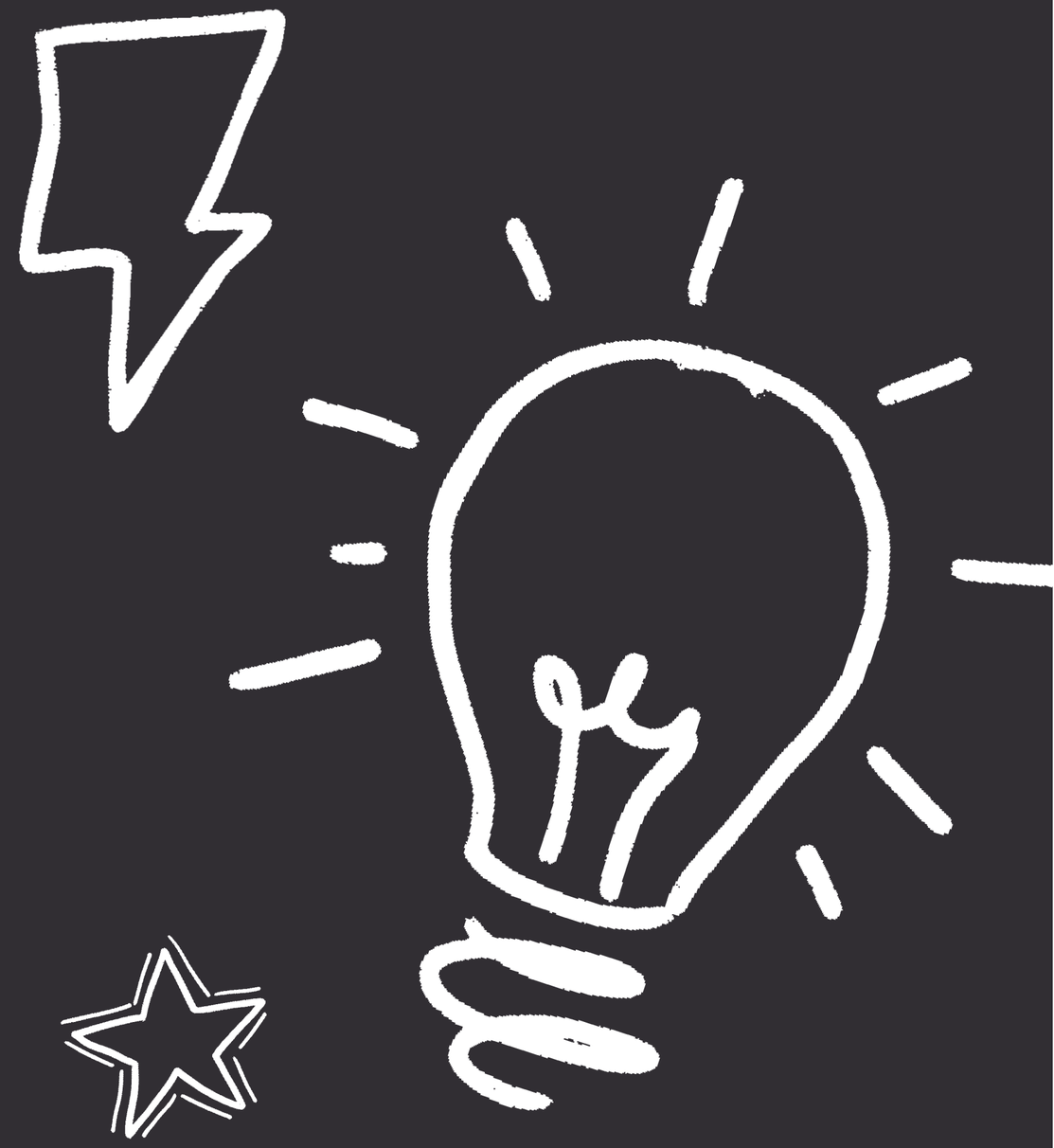
AYUDANTÍA 5 - IIC2413 - 2024-2

Cargadores y limpieza de datos





¿QUÉ VAMOS A VER?

Teoría carga y limpieza de datos

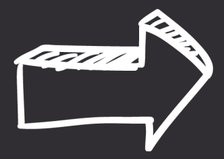
PHP para el procesamiento

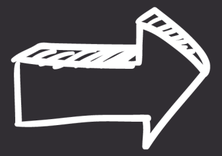
Limpieza en la práctica

Violaciones de IC más comunes

Ejemplos en código

¿Cómo se hace?

1. Carga de datos  Leer datos desde archivos en código PHP.

2. Inserción en tablas  Desde el código de PHP debemos de realizar INSERTS en las tablas de SQL.

3. Corrección de datos  A partir de los violaciones de IC optar por las 3 opciones de corrección.

Partiendo por los cargadores (carga de datos)



Cargadores

Programa encargado de leer los datos e insertarlos en las tablas SQL.

Revisa que los datos sean correctos para poder insertarlos en las tablas.

En caso de ser un dato erróneo, guardar en un archivo para luego luego corregirlo o tomar la decisión de eliminarlo (último recurso).



Ejemplo en código

Conexión con la base de datos:

```php
// Inicializamos las variables para la conexión
$host = "localhost"; // dominio de su grupo en ssh
$dbname = "database_name"; // nombre de la base de datos
$user = "username"; // el usuario
$password = "password"; // contraseña

// Establecemos la conexión con la base de datos.
$conn = pg_connect("host=$host dbname=$dbname user=$user password=$password");

// Comprobamos si la conexión se pudo efectuar
// caso contrario terminamos el programa arrojando un error.
if (!$conn) {
    die("Connection failed: " . pg_last_error());
}
```

Ejemplo en código

Creamos las tablas y comprobamos si se crean correctamente:

```php
$sql = "CREATE TABLE IF NOT EXISTS users (
    id SERIAL PRIMARY KEY,
    username VARCHAR(50) UNIQUE NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
)";

// Ejecutamos la query
$result = pg_query($conn, $sql);

// Se comprueba si la query se ejecutó correctamente
if ($result) {
    echo "Tabla 'users' creada correctamente\n";
} else {
    echo "Error creando la tabla: " . pg_last_error($conn) . "\n";
}
```

Ejemplo en código

Insertamos los valores a la tabla y comprobamos su inserción:

```php
// Insertar un valor en la tabla 'users'
$insert_sql = "INSERT INTO users (username, email) VALUES ('testuser', 'testuser@example.com')";
$insert_result = pg_query($conn, $insert_sql);

// Se comprueba si la inserción se ejecutó correctamente
if ($insert_result) {
    echo "Valor insertado correctamente en la tabla 'users'\n";
} else {
    // Incluir lógica adicional para la corrección de errores:
    // añadir a un archivo y luego procesar dichos errores sobre los datos.
    echo "Error insertando el valor: " . pg_last_error($conn) . "\n";
}

// Cerramos la conexión con el servidor
pg_close($conn);
```

❗ IMPORTANTE ❗

La corrección de datos la tienen que hacer a parte del cargador.

En otras palabras, deben modularizar estos dos procesos.



¿Limpieza de datos?



¿Qúe hacer ante un error?

Principalmente nos vamos a enfocar en 3 opciones para corregir un dato que va en contra de las IC.

1. Corrección de formato.
2. Asignación de nulo.
3. Eliminación.

Siempre tenemos que buscar la pérdida mínima de información.

IMPORTANTE

No pueden alterar el archivo base, en caso de corregir un dato puede ser mediante código, o ser cambiado a mano del archivo de datos con errores.

1. Corrección de formato

Ejemplos de datos que no cumplen el formato:

- Datos que representan lo mismo pero en diferente forma. Deben ser estandarizados.
- El dato no representa lo pedido.
- Datos con errores menores. Son fáciles y rápidos de solucionar.



Ejemplos de correcciones de la entrega 0:

- “Doctor” y “Doctora” en columna de Grado Académico. Vespertino
- “Jornada” en columna Dedicación, que indicaba las horas laborales.
- Mails contenían un espacio, ej: MOREIRA .ZARATE @llimail.kom

2. Asignación de nulo

Se utiliza en casos que el dato no pueda ser corregido, pero no es esencial en las tablas.

Esto se realiza en casos que el valor por default no pueda ser nulo, por lo que se asigna un valor decidido por ustedes para evitar borrar toda la fila y minimizar la perdida de información.

En la entrega 0 se sugirió marcar con ‘x’ para estos casos (puede ser otro valor).

Ejemplos (inventados) de la entrega 0:

- Apellido paterno no existente.
- Teléfono con dato de tipo string.

3. Eliminación del dato

Esta medida debe de aplicarse solamente como último recurso.

¿Por qué? Siempre debemos priorizar la pérdida mínima de datos.

Considerar esta opción solo en casos donde los datos hayan perdido casi todo su valor.

Si se llega a optar por esta opción se debe de eliminar en cascada.

Ejemplo de la entrega 0:

- RUN con el formato inválido (número de cifras, tipo de variable incorrecto, etc.)
- Tuplas con todos atributos nulos.
- ¿Por qué un DV inválido sí lo podemos corregir?
- Un RUN duplicado buscar información adicional para eliminar duplicidad.

Violaciones de IC más comunes

01

Datos duplicados

02

NO estandarizados

03

Tipos de dato incorrectos

04

Datos nulos (cuando no se permite)

Violaciones de IC más comunes

05

Incoherencia entre datos

06

Fuera del formato permitido

En caso de tener dudas sobre corrección de datos, preguntarlas en:

https://github.com/IIC2413/Syllabus-2024-2/discussions/156


Q&A