LET’S START WITH DBMS :)

Data masking techniques

Data masking means hiding or changing sensitive information so it’s protected from unauthorized access. This allows the data to be used for things like testing, analytics, or training without exposing the real data. The goal is to create data that looks real but isn’t actually the original information.

For example : A bank needs to train its machine learning models to detect fraudulent transactions. To protect customers' personal information, the bank uses data masking to replace real account numbers and transaction details with fake but realistic-looking data. This way, the models can be trained effectively without risking exposure of actual customer data.

LET’S START WITH DBMS :)

Data masking techniques

Common Data Masking Techniques

Substitution:

- Replacing sensitive data with random but realistic-looking values. For example, swapping real names with random names from a list.

- Example: A company's employee database contains Social Security Numbers (SSNs). To mask these, the company replaces real SSNs with randomly generated fake SSNs. So, 123-45-6789 might be substituted with 987-65-4321.

LET’S START WITH DBMS :)

Data masking techniques

Common Data Masking Techniques

Shuffling:

- Rearranging the values within a column so that the data remains realistic but is not linked to the original records.

- Example: A hospital has a list of patients with their birth dates. To protect privacy, the hospital shuffles the birth dates among the patients. For instance, Patient A's birth date of 01/15/1980 might be swapped with Patient B's birth date of 07/22/1985.

LET’S START WITH DBMS :)

Data masking techniques

Common Data Masking Techniques

Encryption:

- Encrypting data so that it is unreadable without the decryption key. This can be used as a form of masking if the data doesn't need to be human-readable.

- Example: A retail company encrypts credit card numbers in its database. A number like 4111 1111 1111 1111 might be encrypted to something like Ae34Bf98Gh65Jk21, which can’t be read without the decryption key.

LET’S START WITH DBMS :)

Data masking techniques

Common Data Masking Techniques

Nulling Out:

- Replacing sensitive data with null values or placeholders like "XXXX". This removes the original data but may reduce the usefulness of the dataset.

- Example: An HR department wants to share employee records for analysis but needs to hide salary information. They replace the salary field with null values or placeholders like N/A or XXXX.

LET’S START WITH DBMS :)

Data masking techniques

Common Data Masking Techniques

Number Variance: within a certain range: as after add/sub the data should look realistic. if data = 1000, we can make it 998 or 1003 but cannot make it 34 (so small) as it may not look real

- Altering numeric data by adding or subtracting a random value within a certain range. For example, changing salary figures slightly to mask the exact amounts

- Example: A finance department needs to protect the exact sales figures but still wants to share data for trend analysis. They slightly alter the figures by adding or subtracting a small random amount. For example, a sales figure of $10,000 might be changed to $10,020 or $9,980.

LET’S START WITH DBMS :)

Data masking techniques

Common Data Masking Techniques

Tokenization:

- Tokenization is a technique used to replace sensitive data with non-sensitive equivalents, called "tokens." These tokens can be stored and processed without exposing the original sensitive data, while maintaining a mapping to the original values when needed.

- Example: Imagine a database that stores customer credit card information. To protect this sensitive data, tokenization can be applied. Each original credit card number is replaced with a unique token (e.g., TKN_XYZ123). These tokens have no meaningful relationship to the actual credit card numbers.