Hashing is a technique used to map data (like strings or numbers) to a fixed-size value, typically called a hash code. This is done through a hash function. It is commonly used in areas like data storage, cryptography, and ensuring efficient data retrieval.

Here’s a breakdown of how hashing works:

**Hash Function**: A function that takes input (often a string) and produces a fixed-size output, typically a number or a string. This output is known as the hash value or hash code.

**Hash Table**: A data structure that uses a hash function to store and retrieve data quickly. Each key-value pair is stored in a "bucket" based on the hash value.

**Collision**: A situation where two different inputs generate the same hash value. This is handled by techniques like chaining (linking multiple entries in a single bucket) or open addressing (finding an alternative bucket).

**Applications of Hashing**

**Data Integrity**: Hashing is used to ensure data integrity by comparing hash values before and after transmission.

**Cryptography**: Hash functions are integral to digital signatures and password storage

**Hashing in Databases**: It is used for indexing and retrieving data efficiently.

The package we are going to use bicripte

ALSO TO DO COOKIES WE HAVE TO DOWNLOAD FOLLOWING THINGS

EXPRESS-SESSION

PASSPORT IT WILL ALLOW US TO ADD A LOT STRATARGY FOR AUTHENTICATION

PASSPORT-LOCAL

In **JavaScript**, environment variables are commonly used for storing sensitive information like API keys, database URLs, or configuration settings. Here’s how you can use them in different environments:

## ****1. Using Environment Variables in Node.js****

In **Node.js**, environment variables are accessed using process.env.

### ****Setting an Environment Variable****