* **Difference between PDO and Mysqli**

PDO (PHP Data Objects) and `mysqli\_connect()` are both PHP extensions used for database connectivity, particularly with MySQL databases. However, they have some differences in their usage and features, which might lead developers to choose one over the other. Here's a comparison:

1. \*\*Database Support\*\*:

- `mysqli\_connect()` is specifically designed for MySQL databases.

- PDO, on the other hand, is more versatile and supports multiple database systems, including MySQL, PostgreSQL, SQLite, and more. This makes PDO a better choice if you anticipate needing to switch between different database systems in the future.

2. \*\*Object-Oriented vs. Procedural\*\*:

- `mysqli\_connect()` is procedural in nature. It involves functions like `mysqli\_connect()`, `mysqli\_query()`, etc.

- PDO provides an object-oriented interface for working with databases. It allows for more organized and modular code through the use of objects and methods.

3. \*\*Prepared Statements\*\*:

- Both PDO and `mysqli\_connect()` support prepared statements, which are used to execute the same SQL statement repeatedly with high efficiency.

- However, PDO's prepared statements are more versatile and easier to use. PDO supports named parameters, while `mysqli\_connect()` uses "?" placeholders.

4. \*\*Error Handling\*\*:

- Error handling in PDO is more consistent and unified. PDO throws exceptions for database errors, which makes error handling more straightforward.

- `mysqli\_connect()` requires explicit error handling using functions like `mysqli\_error()` or `mysqli\_errno()`.

5. \*\*Security\*\*:

- Both PDO and `mysqli\_connect()` provide ways to prevent SQL injection attacks. However, PDO is often preferred due to its support for prepared statements and parameterized queries, which inherently protect against SQL injection when used correctly.

6. \*\*Community and Future Support\*\*:

- PDO is a part of the PHP core, which means it's widely used and well-supported. It's less likely to be deprecated or abandoned in future PHP versions compared to specific extensions like `mysqli\_connect()`.

In general, PDO tends to be favored for its versatility, object-oriented approach, and better error handling. However, `mysqli\_connect()` might still be preferred in scenarios where compatibility with older codebases or specific MySQL features is crucial. Ultimately, the choice between PDO and `mysqli\_connect()` depends on the specific requirements of your project and your personal preference as a developer.