**PDO**

PDO (PHP Data Objects) é uma extensão do PHP que fornece uma interface unificada para acessar bancos de dados relacionais. Com o PDO, é possível acessar vários bancos de dados, como MySQL, PostgreSQL e SQLite, utilizando a mesma sintaxe e conjunto de funções.

O PDO permite que o desenvolvedor escreva um código mais limpo e seguro para a manipulação de dados no banco de dados. Isso ocorre porque o PDO oferece suporte a recursos avançados, como a preparação de declarações SQL, que ajudam a proteger contra ataques de injeção de SQL. Além disso, o PDO oferece uma camada de abstração de banco de dados, o que significa que você pode alterar facilmente o banco de dados subjacente sem precisar modificar seu código.

O PDO é uma alternativa mais segura e flexível ao uso do mysql\_ e mysqli\_ funções do PHP, que são consideradas obsoletas.

**STATEMENT**

Dentro de uma query SQL no PHP, um "statement" refere-se a uma instrução SQL completa que é executada no banco de dados. Por exemplo, uma declaração SQL pode ser uma instrução SELECT que seleciona dados de uma tabela no banco de dados, uma instrução INSERT que adiciona novos registros à tabela, uma instrução UPDATE que atualiza registros existentes na tabela ou uma instrução DELETE que remove registros da tabela.

No PHP, é possível usar o objeto PDO para preparar uma instrução SQL como um "statement", que é então executado no banco de dados. O uso de statements preparados é altamente recomendado, pois permite que os dados de entrada sejam tratados de forma segura e eficiente, reduzindo o risco de ataques de injeção de SQL.

Em resumo, o "statement" dentro de uma query SQL no PHP é a instrução SQL completa que é executada no banco de dados, e pode ser preparada com segurança usando o objeto PDO.

**FETCH**

O método fetch() em PHP é usado para recuperar a próxima linha de resultados de uma consulta SQL em um objeto PDOStatement. Quando uma consulta SQL é executada usando um statement PDO, o resultado é retornado como um conjunto de resultados que podem ser iterados linha a linha.

O método fetch() é usado para recuperar a próxima linha do conjunto de resultados como um array associativo, numérico ou ambos. Por padrão, o fetch() retorna um array associativo, onde o nome da coluna é usado como o índice do array. No entanto, é possível especificar o tipo de retorno usando a constante PDO::FETCH\_\*.

**TYPES OF FETCH**

Existem vários tipos de fetch que podem ser usados com o PDO em PHP:

PDO::FETCH\_ASSOC: retorna um array indexado pelo nome da coluna no banco de dados.

PDO::FETCH\_BOTH (valor padrão): retorna um array indexado tanto pelo nome quanto pelo índice numérico da coluna no banco de dados.

PDO::FETCH\_OBJ: retorna um objeto anônimo com propriedades que correspondem às colunas do banco de dados.

PDO::FETCH\_NUM: retorna um array indexado pelo número da coluna no banco de dados.

PDO::FETCH\_CLASS: retorna um objeto de uma classe especificada pelo desenvolvedor, com propriedades que correspondem às colunas do banco de dados.

PDO::FETCH\_BOUND: vincula as colunas do resultado a variáveis no escopo do script.

PDO::FETCH\_LAZY: cria objetos PDOStatement e permite que as colunas sejam acessadas sob demanda.

PDO::FETCH\_COLUMN: retorna apenas uma única coluna de cada linha do resultado.

PDO::FETCH\_GROUP: agrupa as linhas do resultado por valor em uma ou mais colunas.

PDO::FETCH\_UNIQUE: retorna apenas a primeira ocorrência de cada valor distinto em uma ou mais colunas.

PDO::FETCH\_KEY\_PAIR: retorna um array indexado pelo valor de uma coluna e cujo valor é outra coluna.

Esses tipos podem ser passados como parâmetro no método fetch() da classe PDOStatement.