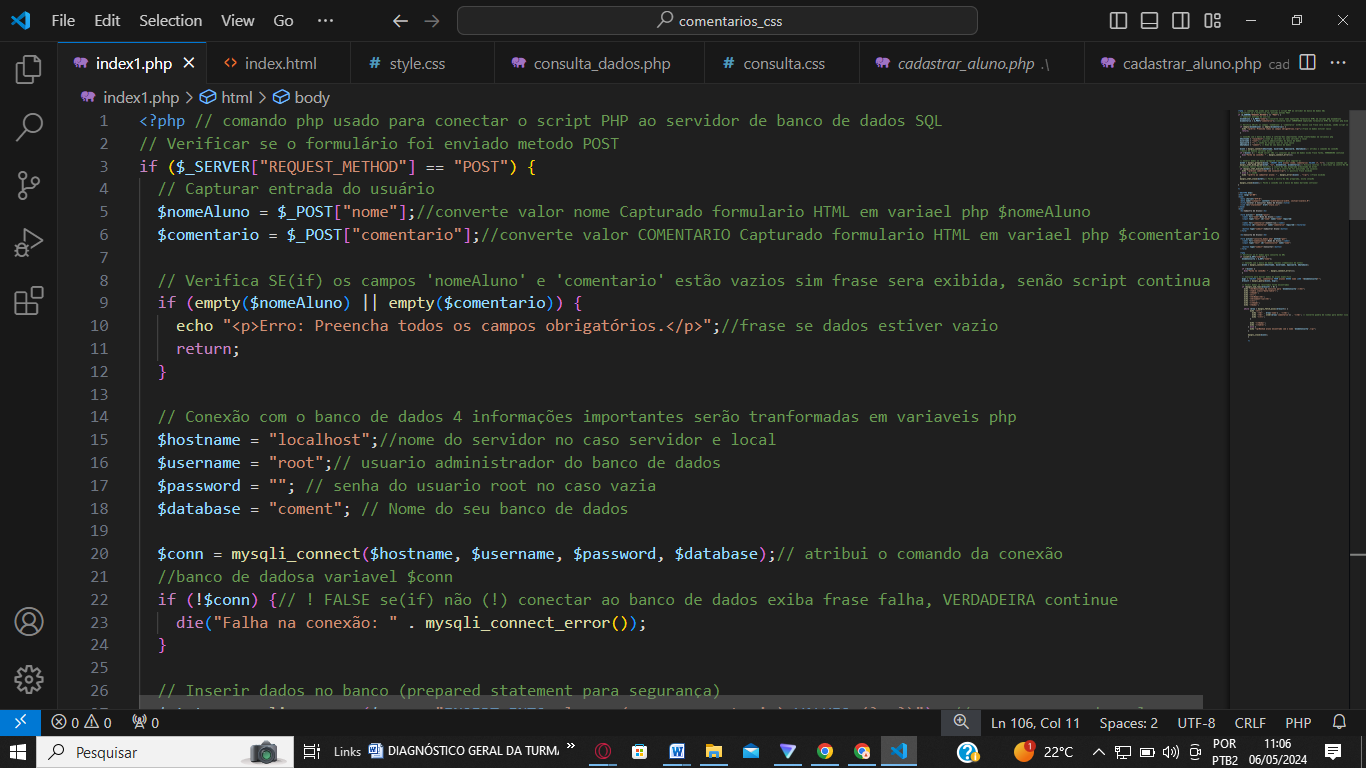
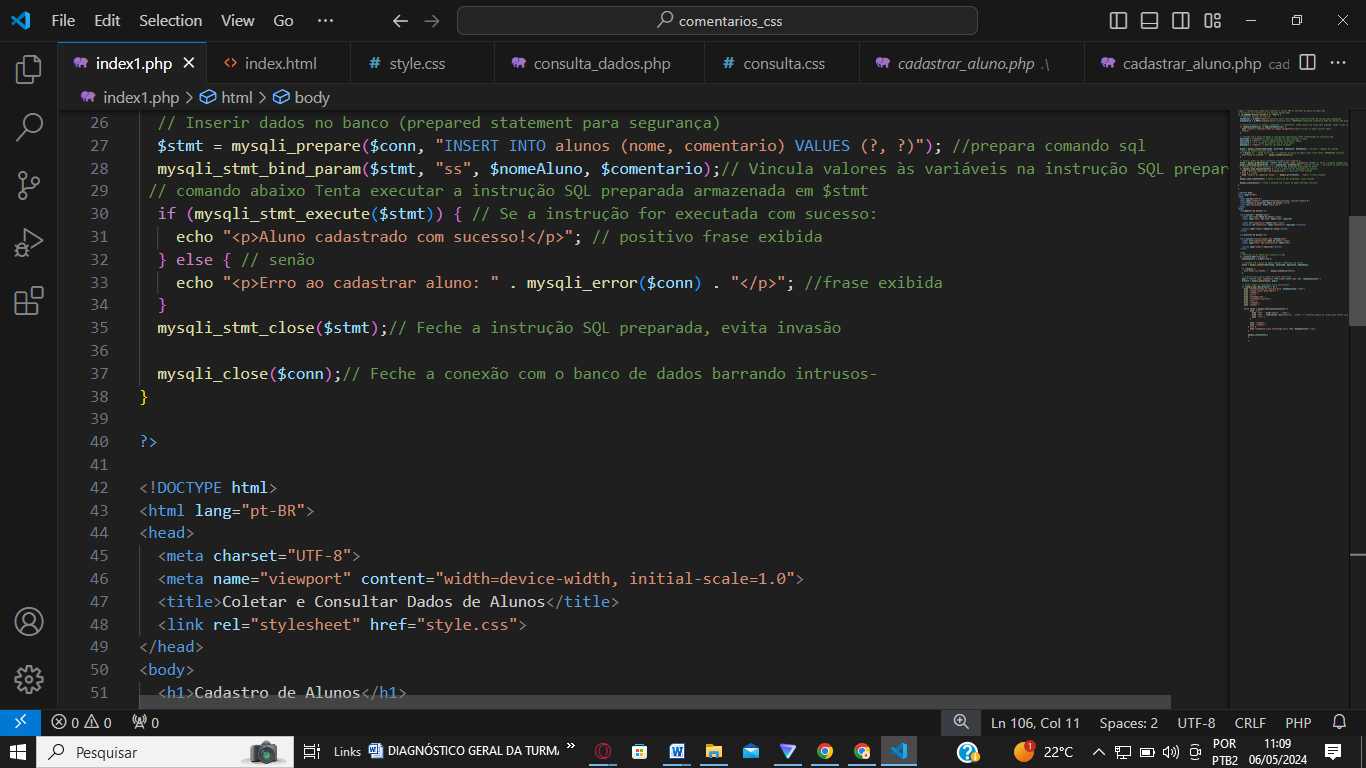
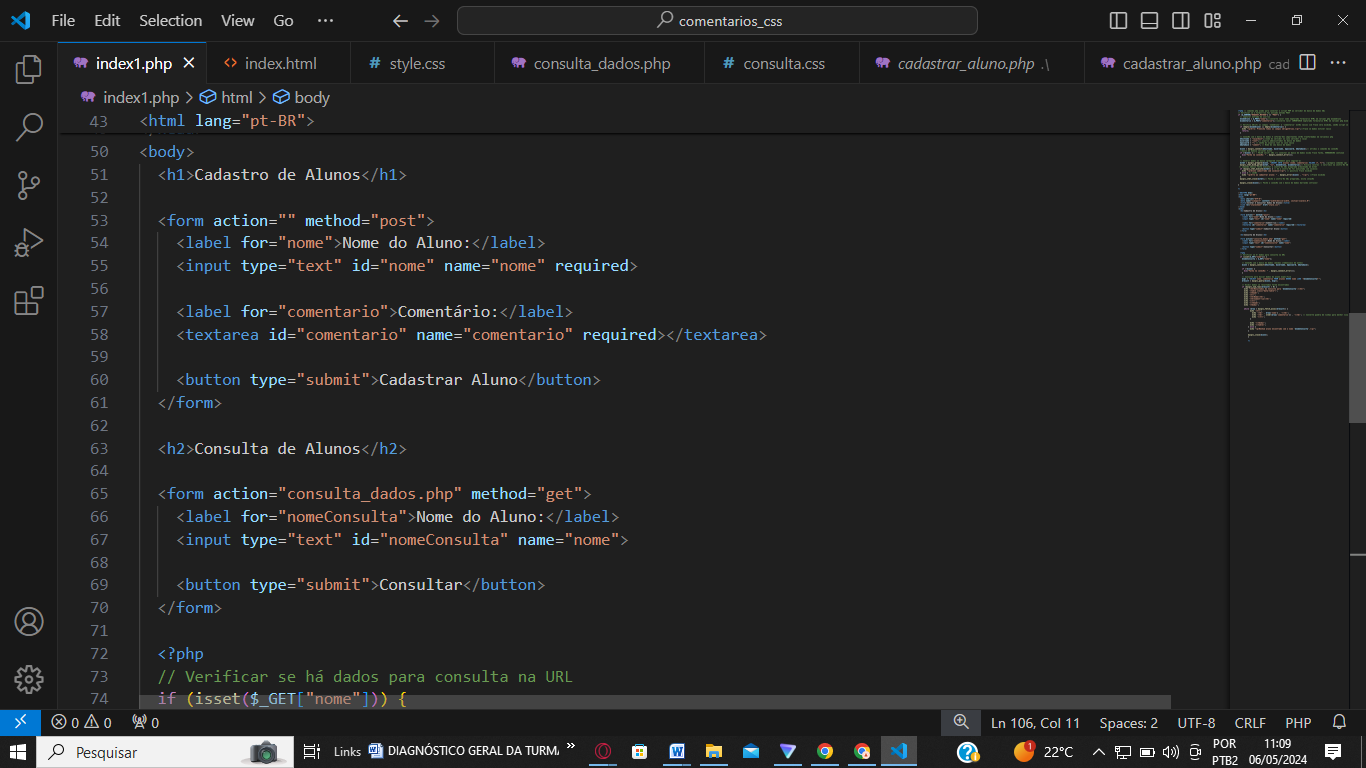
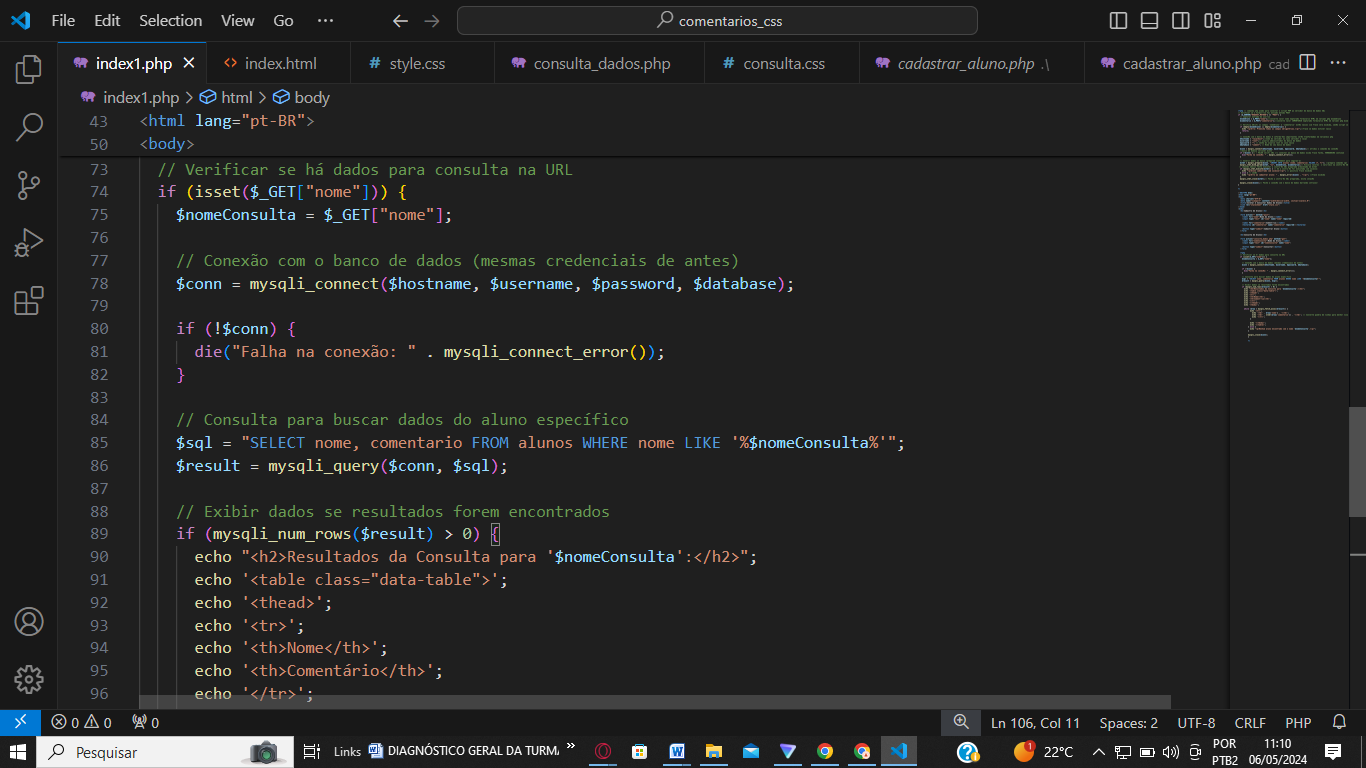
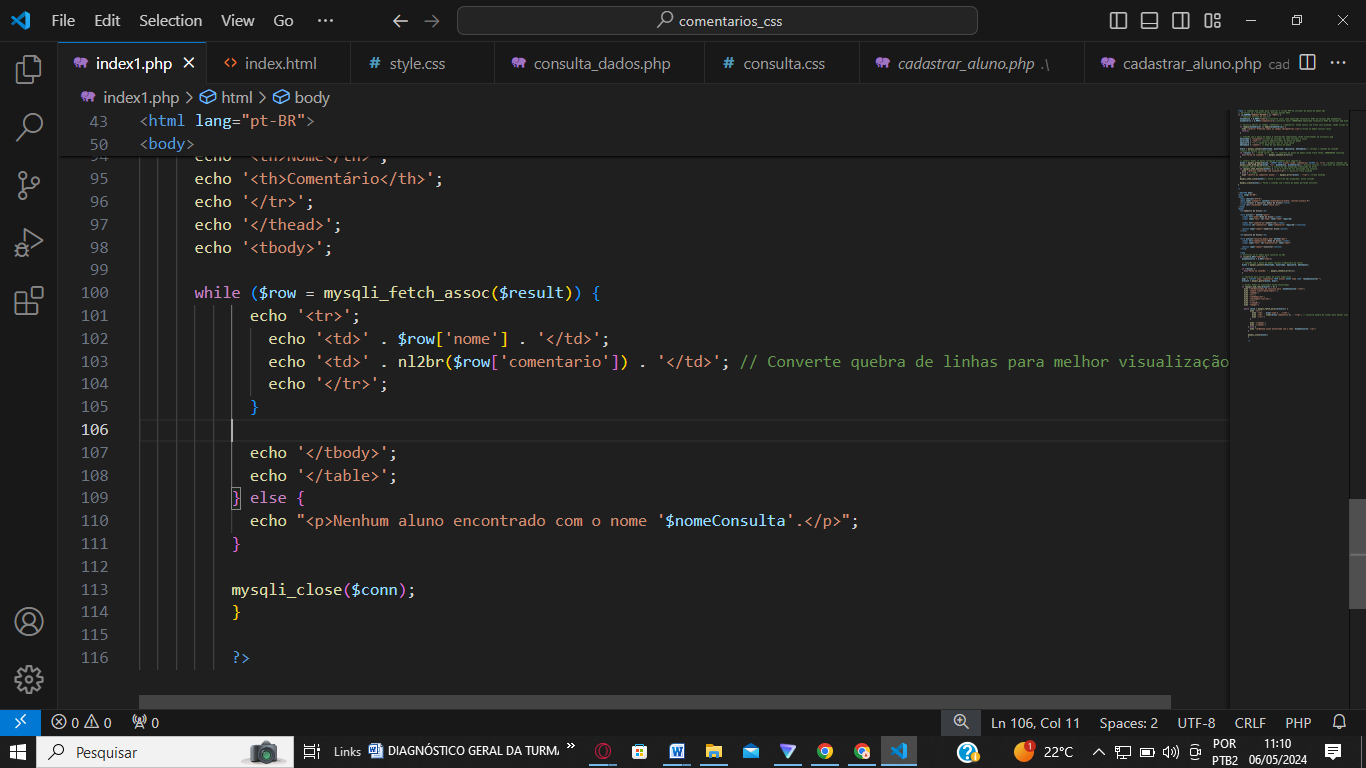
Tutoria passo a passo para criação do arquivo index.php para criação do projeto usuário comentario



**Explicação Simples e Objetiva do Código PHP e Conexão com Banco de Dados MySQL**

**1. Conexão com o Banco de Dados:**

* **$hostname = "localhost";**: Define o nome do servidor de banco de dados. No caso, "localhost" indica que o banco de dados está no mesmo servidor que o código PHP.
* **$username = "root";**: Define o nome de usuário para acessar o banco de dados. "root" é o nome de usuário padrão com privilégios administrativos.
* **$password = "";**: Define a senha do usuário root. Certifique-se de substituir por sua senha real.
* **$database = "coment";**: Define o nome do banco de dados a ser utilizado. No exemplo, o nome do banco é "coment".
* **$conn = mysqli\_connect($hostname, $username, $password, $database);**: Esta linha tenta estabelecer a conexão com o banco de dados utilizando as credenciais fornecidas. A variável $conn armazena o objeto de conexão.
* **if (!$conn)**: Verifica se a conexão foi bem-sucedida. Se $conn for FALSE, significa que a conexão falhou e a mensagem de erro die("Falha na conexão: " . mysqli\_connect\_error()); é exibida.

**2. Cadastro de Aluno:**

* **if ($\_SERVER["REQUEST\_METHOD"] == "POST")**: Verifica se o formulário foi enviado utilizando o método POST.
* **$nomeAluno = $\_POST["nome"];**: Captura o valor do campo "nome" do formulário HTML e o armazena na variável $nomeAluno.
* **$comentario = $\_POST["comentario"];**: Captura o valor do campo "comentario" do formulário HTML e o armazena na variável $comentario.
* **if (empty($nomeAluno) || empty($comentario))**: Verifica se os campos $nomeAluno e $comentario estão vazios. Se um ou ambos estiverem vazios, a mensagem de erro echo "<p>Erro: Preencha todos os campos obrigatórios.</p>"; é exibida e o script termina.
* **$stmt = mysqli\_prepare($conn, "INSERT INTO alunos (nome, comentario) VALUES (?, ?)");**: Prepara uma consulta SQL segura contra ataques de injection. A consulta insere os valores de $nomeAluno e $comentario na tabela "alunos" do banco de dados.
* **mysqli\_stmt\_bind\_param($stmt, "ss", $nomeAluno, $comentario);**: Vincula os valores das variáveis $nomeAluno e $comentario aos parâmetros ? na consulta SQL preparada.
* **if (mysqli\_stmt\_execute($stmt))**: Tenta executar a consulta SQL preparada. Se a execução for bem-sucedida, a mensagem de sucesso echo "<p>Aluno cadastrado com sucesso!</p>"; é exibida.
* **mysqli\_stmt\_close($stmt);**: Fecha a consulta SQL preparada para liberar recursos.
* **mysqli\_close($conn);**: Fecha a conexão com o banco de dados para evitar problemas de desempenho.

**3. Consulta de Alunos:**

* **if (isset($\_GET["nome"]))**: Verifica se a variável $\_GET["nome"] está presente na URL. Essa variável contém o nome do aluno para a consulta.
* **$nomeConsulta = $\_GET["nome"];**: Captura o valor da variável $\_GET["nome"] e o armazena na variável $nomeConsulta.
* **$conn = mysqli\_connect($hostname, $username, $password, $database);**: Estabelece a conexão com o banco de dados da mesma forma que no cadastro.
* **$sql = "SELECT nome, comentario FROM alunos WHERE nome LIKE '%$nomeConsulta%'";**: Cria uma consulta SQL que busca por alunos com nome que contenha o valor de $nomeConsulta. O operador LIKE permite pesquisas por nome que contenham o termo informado.
* **$result = mysqli\_query($conn, $sql);**: Executa a consulta SQL e armazena o resultado na variável $result.
* **if (mysqli\_num\_rows($result) > 0)**: Verifica se a consulta encontrou resultados. Se mysqli\_num\_rows($result) for maior que 0, significa que pelo menos um aluno foi encontrado.
  + **echo "<h2>Resultados da Consulta para '$nomeConsulta':</h2>";**:

**Explicação Simples e Objetiva do Código PHP e Conexão com Banco de Dados MySQL**

**1. Conexão com o Banco de Dados:**

* **$hostname = "localhost";**: Define o nome do servidor de banco de dados. No caso, "localhost" indica que o banco de dados está no mesmo servidor que o código PHP.
* **$username = "root";**: Define o nome de usuário para acessar o banco de dados. "root" é o nome de usuário padrão com privilégios administrativos.
* **$password = "";**: Define a senha do usuário root. Certifique-se de substituir por sua senha real.
* **$database = "coment";**: Define o nome do banco de dados a ser utilizado. No exemplo, o nome do banco é "coment".
* **$conn = mysqli\_connect($hostname, $username, $password, $database);**: Esta linha tenta estabelecer a conexão com o banco de dados utilizando as credenciais fornecidas. A variável $conn armazena o objeto de conexão.
* **if (!$conn)**: Verifica se a conexão foi bem-sucedida. Se $conn for FALSE, significa que a conexão falhou e a mensagem de erro die("Falha na conexão: " . mysqli\_connect\_error()); é exibida.

**2. Cadastro de Aluno:**

* **if ($\_SERVER["REQUEST\_METHOD"] == "POST")**: Verifica se o formulário foi enviado utilizando o método POST.
* **$nomeAluno = $\_POST["nome"];**: Captura o valor do campo "nome" do formulário HTML e o armazena na variável $nomeAluno.
* **$comentario = $\_POST["comentario"];**: Captura o valor do campo "comentario" do formulário HTML e o armazena na variável $comentario.
* **if (empty($nomeAluno) || empty($comentario))**: Verifica se os campos $nomeAluno e $comentario estão vazios. Se um ou ambos estiverem vazios, a mensagem de erro echo "<p>Erro: Preencha todos os campos obrigatórios.</p>"; é exibida e o script termina.
* **$stmt = mysqli\_prepare($conn, "INSERT INTO alunos (nome, comentario) VALUES (?, ?)");**: Prepara uma consulta SQL segura contra ataques de injection. A consulta insere os valores de $nomeAluno e $comentario na tabela "alunos" do banco de dados.
* **mysqli\_stmt\_bind\_param($stmt, "ss", $nomeAluno, $comentario);**: Vincula os valores das variáveis $nomeAluno e $comentario aos parâmetros ? na consulta SQL preparada.
* **if (mysqli\_stmt\_execute($stmt))**: Tenta executar a consulta SQL preparada. Se a execução for bem-sucedida, a mensagem de sucesso echo "<p>Aluno cadastrado com sucesso!</p>"; é exibida.
* **mysqli\_stmt\_close($stmt);**: Fecha a consulta SQL preparada para liberar recursos.
* **mysqli\_close($conn);**: Fecha a conexão com o banco de dados para evitar problemas de desempenho.

**3. Consulta de Alunos:**

* **if (isset($\_GET["nome"]))**: Verifica se a variável $\_GET["nome"] está presente na URL. Essa variável contém o nome do aluno para a consulta.
* **$nomeConsulta = $\_GET["nome"];**: Captura o valor da variável $\_GET["nome"] e o armazena na variável $nomeConsulta.
* **$conn = mysqli\_connect($hostname, $username, $password, $database);**: Estabelece a conexão com o banco de dados da mesma forma que no cadastro.
* **$sql = "SELECT nome, comentario FROM alunos WHERE nome LIKE '%$nomeConsulta%'";**: Cria uma consulta SQL que busca por alunos com nome que contenha o valor de $nomeConsulta. O operador LIKE permite pesquisas por nome que contenham o termo informado.
* **$result = mysqli\_query($conn, $sql);**: Executa a consulta SQL e armazena o resultado na variável $result.
* **if (mysqli\_num\_rows($result) > 0)**: Verifica se a consulta encontrou resultados. Se mysqli\_num\_rows($result) for maior que 0, significa que pelo menos um aluno foi encontrado.
  + **echo "<h2>Resultados da Consulta para '$nomeConsulta':</h2>";**: