

Name: Gaurang Vaghela

Rollno: TEAD-22561

Mini Project Lab

Mini Project

Problem Statement:

Create Database of Products:

✓ Showing rows 0 - 3 (4 total, Query took 0.0002 seconds.)

`SELECT * FROM `products``

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: | Filter rows: | Sort by key:

Extra options

				id	name	price
<input type="checkbox"/>				1	Laptop	50000.00
<input type="checkbox"/>				2	Smartphone	20000.00
<input type="checkbox"/>				3	Headphones	1500.00
<input type="checkbox"/>				4	Monitor	8000.00

☐ Check all | With selected: Edit Copy Delete Export

SQL Injection vulnerable web page:

CODE:

`<?php`

```
$conn = new mysqli("localhost", "root", "", "sql_injection_demo");
```

```
if (isset($_GET["id"])) {
```

```
    $product_id = $_GET["id"];
```

```
    $query = "SELECT * FROM products WHERE id = '$product_id'";
```

```
    $result = $conn->query($query);
```

```
    while ($row = $result->fetch_assoc()) {
```

```
        echo "<p>Product Name: " . $row["name"] . "</p>";
```

```
        echo "<p>Price: $" . $row["price"] . "</p>";
```

```
    }
```

```
}
```

```
?>
```


```
<!DOCTYPE html>
```

```
<html>
```

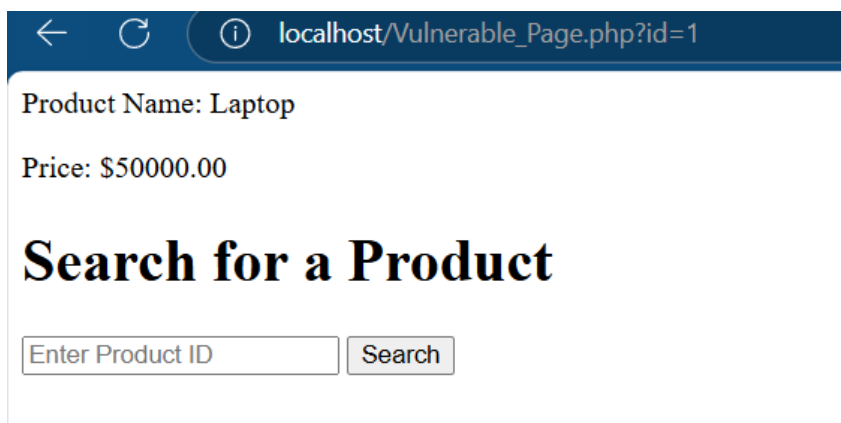
```
<head>
  <title>Product Search</title>
</head>
<body>
  <h1>Search for a Product</h1>
  <form method="GET">
    <input type="text" name="id" placeholder="Enter Product ID">
    <button type="submit">Search</button>
  </form>
</body>
</html>
```

OUTPUT:

If input is '1':



A screenshot of a web browser window. The address bar shows 'localhost/Vulnerable_Page.php'. The page title is 'Search for a Product'. Below the title is a search form with a text input field containing the placeholder text 'Enter Product ID' and a 'Search' button.



A screenshot of a web browser window showing the search results. The address bar shows 'localhost/Vulnerable_Page.php?id=1'. The page displays the product name 'Product Name: Laptop' and the price 'Price: \$50000.00'. Below this, the page title is 'Search for a Product'. At the bottom, there is a search form with a text input field containing the placeholder text 'Enter Product ID' and a 'Search' button.

If input is '1 OR 1=1':

The screenshot shows a web browser at the URL `localhost/Vulnerable_Page.php`. The page title is "Search for a Product". The search input field contains the payload `1' OR '1'='1|` and the "Search" button is clicked. The browser's address bar shows the URL updated to `localhost/Vulnerable_Page.php?id=1%27+OR+%271%27%3D%271`. The page content displays a list of products: Laptop (\$50000.00), Smartphone (\$20000.00), Headphones (\$1500.00), and Monitor (\$8000.00). Below the list, the "Search for a Product" section is repeated with a search input field labeled "Enter Product ID" and a "Search" button.

SQL Injection Prevented web page:

CODE:

```
<?php
```

```
$conn = new mysqli("localhost", "root", "", "sql_injection_demo");
```

```
if (isset($_GET["id"])) {
```

```
    $stmt = $conn->prepare("SELECT * FROM products WHERE id = ?");
```

```
    $stmt->bind_param("i", $_GET["id"]);
```

```
    $stmt->execute();
```

```
    $result = $stmt->get_result();
```

```
    while ($row = $result->fetch_assoc()) {
```

```
        echo "<p>Product Name: " . htmlspecialchars($row["name"]) . "</p>";
```

```
        echo "<p>Price: $" . htmlspecialchars($row["price"]) . "</p>";
```

```
}  
}  
?>
```

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>Product Search</title>  
</head>  
<body>  
  <h1>Search for a Product</h1>  
  <form method="GET">  
    <input type="text" name="id" placeholder="Enter Product ID">  
    <button type="submit">Search</button>  
  </form>  
</body>  
</html>
```

OUTPUT:

If input is '1':

The image shows two screenshots of a web browser. The top screenshot shows the initial search page with the title "Search for a Product", a text input field containing "1", and a "Search" button. The bottom screenshot shows the results page after clicking the search button. The browser address bar now includes "?id=1". The page displays "Product Name: Laptop" and "Price: \$50000.00" above the search form. The search form now has a placeholder "Enter Product ID" in the input field.

localhost/PreventAttack_Page.php

Search for a Product

1 Search

localhost/PreventAttack_Page.php?id=1

Product Name: Laptop

Price: \$50000.00

Search for a Product

Enter Product ID Search

If input is '1 OR 1=1':



A screenshot of a web browser window. The address bar shows 'localhost/PreventAttack_Page.php'. The page title is 'Search for a Product'. Below the title is a search input field containing the text '1' OR '1'='1|' and a 'Search' button. The page content is mostly blank, indicating a successful SQL injection that bypassed the search filter.



A screenshot of a web browser window. The address bar shows 'localhost/PreventAttack_Page.php?id=1%27+OR+%271%27%3D%271'. The page displays the product details for 'Laptop' with a price of '\$50000.00'. Below this, the page title is 'Search for a Product'. At the bottom, there is a search input field with the placeholder text 'Enter Product ID' and a 'Search' button. This indicates that the SQL injection attempt was blocked by the application's security measures.