



**Name:** Lozano, Eunice L

**Subject:** IS Elec 2 - Web Systems

**Course & Block:** BSIS – 2

**Professor:** Reyamar Llagas

**Troubleshooting Exercise**

SCENARIO	PROBLEM	Correct Fix
1	Using \$_POST instead of \$_GET	Use <b>GET</b> because the ID is coming from the URL.
2	Missing quotes around a string value in SQL	Add <b>single quotes</b> around the first name since it is text.
3	SQL injection risk from direct user input	Use a <b>prepared statement</b> to safely filter by age.
4	Inserts data even when form fields are empty	Add <b>input validation</b> to check if fields are not empty before inserting.
5	Wrong POST key (emial instead of email)	Use the <b>correct key name</b> that matches the form field.
6	Unsafe delete using raw GET input	Use <b>intval()</b> or a <b>prepared statement</b> to sanitize the ID before deleting.
7	Update query fails but still prints “Updated!”	Add <b>error checking</b> and ensure the email is properly quoted.
8	Only one record prints because fetch is called once	Use a <b>while loop</b> to display all rows from the query.
9	Using POST even though the link sends GET	Change to <b>GET</b> since links always pass data through the URL.
10	Wrong variable name used in SQL (aeg instead of age)	Use the <b>correct variable name</b> in the query.
11	Form sends GET but PHP reads POST	Make the <b>form method match</b> the PHP superglobal being used.
12	Numeric ID placed inside quotes	Remove the quotes or convert the value to an <b>integer</b> before using it.
13	Missing WHERE clause in UPDATE (updates all rows)	Add a <b>WHERE clause</b> to update only the specific record.
14	Wrong formatting of POST array values in SQL	Use the <b>correct array access</b> and wrap text values in quotes.
15	GET page number not validated, causing huge offsets	<b>Validate and limit</b> the page number to prevent large or harmful values.

Explanation:	Scenario
<ul style="list-style-type: none"><li>- Kung galing sa URL yung data, GET talaga ang dapat gamitin. POST ginagamit lang kapag form ang nag-send.</li></ul>	Scenario 1
<ul style="list-style-type: none"><li>- Yung name (like “Ana”) ay text, so dapat naka-quotes sa SQL. Pag walang quotes, iniisip ng database column name siya, kaya error.</li></ul>	Scenario 2
<ul style="list-style-type: none"><li>- Diretsong nilalagay ang GET value sa query, kaya pwedeng magpasok ng harmful code (ex. 1 OR 1=1). Kaya kailangan prepared statement para safe.</li></ul>	Scenario 3
<ul style="list-style-type: none"><li>- I-check muna kung may laman yung input. Para iwas sa blank or incomplete na data.</li></ul>	Scenario 4
<ul style="list-style-type: none"><li>- Mali ang spelling ng key (emial), kaya hindi makuha yung value. Dapat eksaktong match sa name attribute ng form.</li></ul>	Scenario 5
<ul style="list-style-type: none"><li>- intval() ensures number lang, so hindi pwedeng maglagay ng 1 OR 1=1.</li></ul>	Scenario 6
<ul style="list-style-type: none"><li>- Kailangan i-check kung nag-succeed yung query bago mag-print ng success message.</li></ul>	Scenario 7
<ul style="list-style-type: none"><li>- mysqli_fetch_assoc() gets only one row. Kung gusto lahat ng records, kailangan ng while loop para ma-loop bawat row..</li></ul>	Scenario 8
<ul style="list-style-type: none"><li>- Link (&lt;a href&gt;) always uses GET, kaya kung POST ang ginamit sa PHP, hindi niya makukuha yung id. Dapat GET ang gamitin.</li></ul>	Scenario 9
<ul style="list-style-type: none"><li>- Mali yung variable (\$aeg). Dapat \$age. Kapag mali ang spelling, undefined siya at hindi gagana yung query.</li></ul>	Scenario 10
<ul style="list-style-type: none"><li>- Form sends GET pero PHP naghahanap ng POST. Magkaka-undefined index. Dapat mag-match yung method at superglobal na ginagamit.</li></ul>	Scenario 11
<ul style="list-style-type: none"><li>- Numbers don’t need ' ' kasi di sila text. Mas efficient if walang quotes.</li></ul>	Scenario 12
<ul style="list-style-type: none"><li>- Walang WHERE, kaya lahat ng rows maa-update. Dapat laging may WHERE para specific lang ang ma-update.</li></ul>	Scenario 13
<ul style="list-style-type: none"><li>- Strings kailangan naka-quotes, at dapat tama ang syntax ng array keys. Without correct quotes, magiging invalid ang SQL.</li></ul>	Scenario 14
<ul style="list-style-type: none"><li>- Nililimit ang allowed page number para hindi bumagsak ang system.</li></ul>	Scenario 15