Unit 3 MCQ

1. What is the purpose of a user-defined function in PHP?

- A) To handle server configurations
- B) To perform pre-defined tasks only
- C) To allow users to create custom functionalities
- D) To include files in PHP

Answer: C) To allow users to create custom functionalities

---

2. Which of the following is a built-in PHP function used to display information about the PHP environment?

- A) `print()`
- B) `include()`
- C) `phpinfo()`
- D) `header()`

Answer: C) `phpinfo()`

---

3. In PHP, which superglobal variable is used to store information about form data submitted via GET method?

- A) `$_POST`
- B) `$_SESSION`
- C) `$_COOKIE`
- D) `$_GET`

Answer: D) `$_GET`

---

4. Which PHP function is used to get the current date and time?

- A) `getDate()`
- B) `date()`
- C) `time()`
- D) `date_time()`

Answer: B) `date()`

---

5. Which function is used to round a floating-point number to the nearest integer in PHP?

- A) `round()`
- B) `floor()`
- C) `ceil()`
- D) `nearest()`

Answer: A) `round()`

---

6. What does the `$_POST` superglobal variable do?

- A) Stores cookies
- B) Contains information about the request headers
- C) Collects data sent via HTTP POST method
- D) Stores session data

Answer: C) Collects data sent via HTTP POST method

---

7. Which of the following is used to start a PHP session?

- A) `session_start()`
- B) `start_session()`
- C) `session()`
- D) `$_SESSION`

Answer: A) `session_start()`

---

8. Which PHP function is used to find the length of a string?

- A) `strlen()`
- B) `str_len()`
- C) `count()`
- D) `size()`

Answer: A) `strlen()`

---

9. What is the purpose of the `header()` function in PHP?

- A) To modify the content displayed in the page header
- B) To set or modify HTTP headers before output
- C) To include external files
- D) To print the output to the browser

Answer: B) To set or modify HTTP headers before output

---

10. Which PHP function can you use to display output on the browser?

- A) `echo`
- B) `print()`
- C) `phpinfo()`
- D) All of the above

Answer: D) All of the above

Unit 4 MCQ

1. Which HTML form element allows users to input multiple lines of text?

- A) Text Box
- B) Text Area
- C) Password
- D) Radio Button

Answer: B) Text Area

---

2. In a form, which attribute specifies how the form data should be sent when it is submitted?

- A) `method`
- B) `action`
- C) `target`
- D) `submit`

Answer: A) `method`

---

3. Which PHP superglobal variable is used to handle file uploads?

- A) `$_POST`
- B) `$_GET`
- C) `$_FILES`
- D) `$_SESSION`

Answer: C) `$_FILES`

---

4. Which form element is used to allow a user to select multiple options in a form?

- A) Combo Box
- B) Radio Button
- C) Text Box
- D) Checkbox

Answer: D) Checkbox

---

5. Which method attribute value should be used to send sensitive data securely in a form?

- A) `GET`
- B) `POST`
- C) `SAVE`
- D) `PUT`

Answer: B) `POST`

---

6. In PHP, which function can you use to display error messages for debugging purposes?

- A) `error_reporting()`
- B) `show_errors()`
- C) `report_errors()`

- D) `display_errors()`

Answer: A) `error_reporting()`

---

7. What does the regular expression symbol `^` represent?

- A) Any single character
- B) End of a line
- C) Start of a line
- D) Any digit

Answer: C) Start of a line

---

8. Which error level in PHP displays non-critical warnings that do not stop script execution?

- A) E_ERROR
- B) E_WARNING
- C) E_PARSE
- D) E_STRICT

Answer: B) E_WARNING

---

9. Which regular expression symbol is used to match any single character except newline?

- A) `*`
- B) `+`
- C) `.`
- D) `^`

Answer: C) `.`

---

10. What does the `move_uploaded_file()` function in PHP do?

- A) Uploads a file to a database
- B) Moves an uploaded file to a new location

- C) Deletes a file from the server
- D) Renames a file on the server

Answer: B) Moves an uploaded file to a new location

Unit 5 mcq

1. Which PHP function is commonly used to establish a connection to a MySQL database?

- A) `mysql_connect()`
- B) `mysql_query()`
- C) `mysqli_connect()`
- D) `db_connect()`

Answer: C) `mysqli_connect()`

---

2. Which SQL command is used to remove a record from a table?

- A) `DELETE`
- B) `DROP`
- C) `REMOVE`
- D) `ERASE`

Answer: A) `DELETE`

---

3. In PHP, which function is used to select a database to work with after a connection is established?

- A) `mysqli_select_db()`
- B) `mysql_select_database()`
- C) `select_database()`
- D) `choose_db()`

Answer: A) `mysqli_select_db()`

---

4. Which SQL statement is used to modify existing data in a database table?

- A) `INSERT`
- B) `UPDATE`

- C) `ALTER`
- D) `MODIFY`

Answer: B) `UPDATE`

---

5. What is ODBC used for in PHP?

- A) Connecting to an Oracle database
- B) Opening a network connection
- C) Providing a standard API for connecting to different databases
- D) Displaying database results in HTML

Answer: C) Providing a standard API for connecting to different databases

---

6. Which of the following is the correct syntax for inserting data into a table using SQL?

- A) `INSERT DATA INTO table_name VALUES (...)`
- B) `INSERT INTO table_name VALUES (...)`
- C) `ADD INTO table_name VALUES (...)`
- D) `PUT INTO table_name VALUES (...)`

Answer: B) `INSERT INTO table_name VALUES (...)`

---

7. In PHP, which function is used to execute an SQL query?

- A) `execute()`
- B) `mysqli_query()`
- C) `mysql_query_execute()`
- D) `db_query()`

Answer: B) `mysqli_query()`

---

8. Which SQL clause is used to filter results based on a specific condition?

- A) `ORDER BY`
- B) `GROUP BY`
- C) `WHERE`

- D) `HAVING`

Answer: C) `WHERE`

---

9. In PHP, which function is used to close a database connection?

- A) `mysqli_disconnect()`
- B) `db_close()`
- C) `close()`
- D) `mysqli_close()`

Answer: D) `mysqli_close()`

---

10. Which SQL statement is used to retrieve data from a database?

- A) `FETCH`
- B) `GET`
- C) `SELECT`
- D) `SHOW`

Answer: C) `SELECT`

Unit 3 Questions

3 Marks questions

1. Define a user-defined function in PHP that accepts two arguments, adds them together, and returns the result. Provide an example of calling this function with sample values.

Answer:

```
php
Copy code
function addNumbers($a, $b) {
    return $a + $b;
}

// Example call
$result = addNumbers(5, 10);
echo "The result is: " . $result; // Output: The result is: 15
```

In this example, the function `addNumbers` takes two arguments, adds them, and returns the sum. It is called with values 5 and 10, giving an output of 15.

---

2. Explain the difference between the `$_GET` and `$_POST` superglobal variables in PHP. In which scenarios would you use each?

Answer:

- `$_GET` and `$_POST` are both superglobal arrays in PHP used to collect data from HTML forms.
- `$_GET` collects form data sent through the URL as part of the query string. It is useful for sending non-sensitive data and allowing bookmarking of the URL with parameters.
- `$_POST` collects form data sent in the HTTP request body, making it more secure for sensitive data. It is often used for actions that modify data, such as submitting a form to create or update a record in a database.

Example:

- Use `$_GET` for search forms, where the parameters can be displayed in the URL.
- Use `$_POST` for login forms, where data like passwords should be kept secure and not visible in the URL.

---

3. Describe how to set a cookie in PHP with a name, value, and expiration time. Write a short PHP code snippet demonstrating how to set a cookie named "user" with the value "John" that expires in 7 days.

Answer:

- Cookies are set using the `setcookie()` function in PHP, which requires the cookie name, value, and expiration time.
- The expiration time is set using `time()` (current time) plus the number of seconds for the desired expiration period.

Code Example:

```
php
Copy code
setcookie("user", "John", time() + (7 * 24 * 60 * 60)); // 7 days from now
```

- This code sets a cookie named "user" with the value "John" that expires in 7 days.

---

4. Write a PHP script that displays the current date and time in the format "Y-m-d H:i

" and calculates the square root of 49, displaying the result on the same page.

Answer:

```
php
Copy code
// Display current date and time
echo "Current date and time: " . date("Y-m-d H:i:s") . "<br>";

// Calculate and display the square root of 49
echo "Square root of 49 is: " . sqrt(49); // Output: 7
```

- This script uses `date()` to format and display the current date and time and `sqrt()` to calculate and display the square root of 49, which is 7.

4 marks questions

1. Write a PHP function that takes a string as an argument, converts it to uppercase, and returns the result. Then, write a short code snippet that calls this function with the string "hello world" and displays the result.

Answer:

```
php
Copy code
// Define the function
function convertToUpper($inputString) {
    return strtoupper($inputString);
}

// Call the function with "hello world"
$result = convertToUpper("hello world");
echo "The uppercase result is: " . $result; // Output: The uppercase result
is: HELLO WORLD
```

- This code defines a function `convertToUpper` that uses the built-in `strtoupper()` function to convert a given string to uppercase. The function is called with "hello world", and it outputs "HELLO WORLD."

---

2. Explain how sessions work in PHP. Write a code snippet to start a session, set a session variable `username` with the value "admin", and then retrieve and display this session variable on the page.

Answer:

- Sessions in PHP allow data to be stored across multiple pages. Each user has a unique session ID, usually stored in a cookie, which helps the server track user data. Sessions are useful for storing user data, like login status, that needs to persist while navigating through different pages on a website.

Code Example:

```
php
Copy code
// Start the session
session_start();

// Set a session variable
$_SESSION['username'] = "admin";

// Retrieve and display the session variable
echo "The username is: " . $_SESSION['username']; // Output: The username is:
admin
```

- This code snippet starts a session using `session_start()`, sets a session variable `username` with the value "admin", and then retrieves and displays this value using `$_SESSION['username']`.

Unit 4 questions

3 marks questions

1. Explain the difference between the GET and POST methods in an HTML form. Provide an example of when to use each method.

Answer:

- The `GET` method sends form data as URL parameters, which makes it visible in the browser's address bar. It is best used for actions that do not modify server data, such as search forms.
- The `POST` method sends form data in the request body, making it more secure for sensitive information. It is commonly used for actions that modify data, such as login or registration forms.

Example:

- Use `GET` for search forms where parameters are shown in the URL.
- Use `POST` for forms that handle sensitive data, like passwords.

---

2. Write a PHP code snippet that allows a user to upload a file to the server. Explain the purpose of the `$_FILES` superglobal in this process.

Answer:

```
php
Copy code
// Check if a file is uploaded
if (isset($_FILES['uploadedFile'])) {
    $targetDirectory = "uploads/";
    $targetFile = $targetDirectory .
basename($_FILES['uploadedFile']['name']);

    // Move the uploaded file to the target directory
    if (move_uploaded_file($_FILES['uploadedFile']['tmp_name'], $targetFile))
{
        echo "File uploaded successfully!";
    } else {
        echo "Error uploading file.";
    }
}
```

- `$_FILES` is a PHP superglobal used to access file upload information. It contains data such as the file name, temporary file path, and file size, allowing us to handle file uploads.

---

3. What are regular expressions in PHP, and name three common symbols used in regular expressions? Give a brief description of each symbol.

Answer:

- Regular expressions are patterns used to match character combinations in strings. They are commonly used for validation and search/replace operations.
- **Symbols:**
  - `^` : Indicates the start of a string.
  - `$` : Indicates the end of a string.
  - `.` : Matches any single character except newline.

---

4. What are the different error levels in PHP? Provide examples of two error levels and explain when each might occur.

Answer:

- PHP has various error levels that indicate the type and severity of errors.

- **E_WARNING**: Non-fatal errors that do not halt script execution, such as including a missing file.
- **E_NOTICE**: Minor issues, often related to uninitialized variables or incorrect variable types. These do not stop script execution but may indicate potential bugs.

4 Marks questions

1. Explain the process of uploading files to a web server using PHP. Include the role of form attributes and the `$_FILES` superglobal in this process.

Answer: To upload files to a web server using PHP, an HTML form with the attribute `enctype="multipart/form-data"` is used. This attribute allows the form to handle file data instead of just text. The form typically includes a file input element where users can select files for upload.

When the form is submitted, PHP stores file information in the `$_FILES` superglobal, which contains details such as:

- **name**: The original name of the uploaded file.
- **type**: The MIME type of the file.
- **size**: The file size in bytes.
- **tmp_name**: The temporary location on the server where the file is stored.
- **error**: Any error code associated with the file upload process.

---

2. Describe the different types of errors in PHP and explain how PHP's error levels help in debugging and managing errors effectively.

Answer: PHP categorizes errors by severity, helping developers identify issues effectively. Common types include:

- **Parse Errors (E_PARSE)**: Syntax errors in the code, often preventing the script from running.
- **Fatal Errors (E_ERROR)**: Critical errors that stop script execution immediately, such as calling an undefined function.
- **Warnings (E_WARNING)**: Non-fatal errors that do not halt script execution. These often occur when including a missing file or calling an incorrect function parameter.
- **Notices (E_NOTICE)**: Minor issues that don't interrupt script execution, typically generated when accessing undefined variables or using variables in unintended ways.

5 unit questions

3 marks questions

1. Explain how to establish a database connection in PHP using MySQL. What key details are required for this connection?

Answer: To establish a database connection in PHP using MySQL, the `mysqli` or `PDO` (PHP Data Objects) extension is typically used. Key details required for the connection include:

- **Hostname**: The server address where the MySQL database is hosted, often "localhost" if on the same server.
- **Username**: The username for the MySQL database.
- **Password**: The password associated with the MySQL username.
- **Database Name**: The specific database to connect to within the server.

These details are passed to the connection function, and if successful, a connection is established. If the connection fails, an error message can be displayed to help diagnose the issue.

---

2. What is the purpose of SQL `INSERT`, `UPDATE`, and `DELETE` commands in database operations? Give a brief description of each.

Answer:

- **INSERT**: The `INSERT` command is used to add new records (rows) to a table within a database. It specifies the table and the values for each column.
- **UPDATE**: The `UPDATE` command is used to modify existing records in a table. It allows changes to be made to one or more columns based on specified conditions (e.g., updating the name of a user with a specific ID).
- **DELETE**: The `DELETE` command is used to remove records from a table based on conditions. Unlike `DROP`, which removes entire tables, `DELETE` targets specific rows within a table.

These commands allow for the management of data within a database, enabling dynamic applications.

---

3. Describe how prepared statements help prevent SQL injection in PHP. Why is this important?

Answer: Prepared statements in PHP help prevent SQL injection by separating SQL code from user input. Instead of directly inserting user input into SQL commands, prepared statements use

placeholders, which are then bound to user-provided values. This ensures that any input is treated strictly as data, not as executable SQL code.

This is important because it protects against SQL injection attacks, where attackers try to execute malicious SQL commands through user input fields. SQL injection can lead to unauthorized access, data breaches, or data loss, making prepared statements an essential security measure in PHP database operations.

---

4. What is ODBC, and how is it used in PHP for database connectivity?

Answer: ODBC (Open Database Connectivity) is a standard API that allows PHP to connect to different types of databases, such as SQL Server, Oracle, and more, using a single, consistent interface. In PHP, the ODBC functions can be used to connect to an ODBC-compliant database.

This is done by specifying a Data Source Name (DSN), which contains the details of the database driver and connection settings. Using ODBC is beneficial when applications need to interact with multiple types of databases, as it abstracts the specifics of each database into a common interface.

4 marks questions

1. What are the key steps involved in connecting a PHP application to a MySQL database, and what are the common methods used for this connection?

Answer: The key steps involved in connecting a PHP application to a MySQL database include:

1. **Establishing a Connection**: To connect to the database, you need to specify the host (usually `localhost`), the database username, password, and the database name.
2. **Using Connection Methods**: The two most common methods to establish a connection are:
    - **`mysqli`**: A PHP extension that allows access to MySQL databases. It supports both procedural and object-oriented programming.
    - **`PDO` (PHP Data Objects)**: A more flexible database access layer that allows you to connect to different types of databases, including MySQL, PostgreSQL, and SQLite.
3. **Error Handling**: After attempting to connect, it's important to check if the connection was successful. If the connection fails, proper error handling is necessary to ensure the application does not proceed with further database operations.

These steps allow PHP to interact securely with MySQL databases, handling potential errors gracefully.

---

2. Explain the importance of SQL commands like `INSERT`, `UPDATE`, and `DELETE` in database management, and provide a brief description of each operation.

Answer: SQL commands like `INSERT`, `UPDATE`, and `DELETE` are fundamental for managing data within a database:

1. **`INSERT`**: The `INSERT` command is used to add new records into a database table. This operation allows data to be inserted into the specified columns of a table. It is essential for populating a database with new information.
2. **`UPDATE`**: The `UPDATE` command modifies existing records in a database. It allows you to change the values of one or more columns based on specified conditions, such as updating a user's email address. This operation is critical for maintaining accurate and up-to-date data.
3. **`DELETE`**: The `DELETE` command is used to remove records from a database table based on certain conditions. Unlike `DROP`, which removes the entire table, `DELETE` only affects specific rows. This operation is important for cleaning up outdated or incorrect data.