

## Outlines

### Data Access with MySQL, Email, and Web Services:

Using MySQL with PHP:

- Connect to MySQL and select the database
- Execute simple queries
- Retrieve query results
- Count returned records
- Update records
- Add, view, and delete records with PHP

### Email:

- Understand email
- Receive email with PHP
- Send email

### Web Services:

- Introduction to web services
- Web services model
- Platform elements

## Data Access with MySQL, Email, and Web Services

### Using MySQL with PHP

MySQL is a popular open-source database management system, commonly used with PHP. In this chapter, we will explore how to use MySQL with PHP.

### Connecting to MySQL and Selecting the Database

To connect to MySQL, we use the `mysqli_connect()` function in PHP. This function takes four arguments: the server name, username, password, and database name. Here's an example:

```
$con = mysqli_connect("localhost", "my_user", "my_password", "my_db");
```

Once we have connected to the database, we need to select the database we want to work with:

## B.C.A. Semester — IV BCA-402 : Building Application Using PHP

### Unit 4 : Data Access with MySql, Email, Web Services

---

```
mysqli_select_db($con, "my_db");
```

### Executing Simple Queries

To execute a simple query with MySQL, we use the `mysqli_query()` function in PHP. This function takes two arguments: the database connection and the SQL query. Here's an example:

```
$result = mysqli_query($con, "SELECT * FROM my_table");
```

### Retrieving Query Results

To retrieve the results of a query, we use the `mysqli_fetch_array()` function in PHP. This function returns an array of the rows in the result set. Here's an example:

```
while ($row = mysqli_fetch_array($result)) {  
  
    echo $row['column_name'];  
  
}
```

### Counting Returned Records

To count the number of records returned by a query, we use the `mysqli_num_rows()` function in PHP. Here's an example:

```
$num_rows = mysqli_num_rows($result);  
  
echo "There are " . $num_rows . " rows in the result set.";
```

### Updating Records

To update records in a MySQL database, we use the `mysqli_query()` function with an UPDATE statement. Here's an example:

```
mysqli_query($con, "UPDATE my_table SET column_name = 'new_value' WHERE id = 1");
```

### Adding, Viewing, and Deleting Records with PHP

To add a record to a MySQL database, we use the `mysqli_query()` function with an INSERT statement. Here's an example:

```
mysqli_query($con, "INSERT INTO my_table (column1, column2) VALUES ('value1', 'value2')");
```

## B.C.A. Semester — IV BCA-402 : Building Application Using PHP

### Unit 4 : Data Access with MySql, Email, Web Services

---

To view a record from a MySQL database, we use a SELECT statement:

```
$result = mysqli_query($con, "SELECT * FROM my_table WHERE id = 1");  
  
$row = mysqli_fetch_array($result);  
  
echo $row['column_name'];
```

To delete a record from a MySQL database, we use the `mysqli_query()` function with a DELETE statement. Here's an example:

```
mysqli_query($con, "DELETE FROM my_table WHERE id = 1");
```

## Email

### Understanding Email

Email is a method of exchanging messages between people using electronic devices. In PHP, we can use the `mail()` function to send emails.

### Receiving Email with PHP

To receive email with PHP, we can use the `imap_open()` function. This function takes four arguments: the server name, username, password, and mailbox name. Here's an example:

```
$mailbox = imap_open("{mail.example.com:143}INBOX", "username", "password");
```

### Sending Email

To send email with PHP, we use the `mail()` function. This function takes five arguments: the recipient's email address, the subject of the email, the message, any additional headers, and any additional parameters. Here's an example:

```
$to = "recipient@example.com";  
  
$subject = "Test Email";  
  
$message = "This is a test email.";  
  
$headers = "From: sender@example.com";  
  
mail($to, $subject, $message, $headers);
```

## **Web Services**