يعني إيه **MVC؟**

**Model-View-Controller** هو من **Design Patterns** المشهوره ومهمته انه يفصل الابلكيشن علي كذا جزء عشان التعامل مع ال**Project** يكون اسهل وبيتقسم كده:

* **Model:**هو الجزء المسؤول عن التعامل مع الداتا، سواء كانت داتا من داتابيز أو من أي مكان تاني زي JSON. كمان ال**Business Logic** كله بيكون هنا، يعني أي **Logic** متعلق بالداتا بيتم تنفيذه في ال**Model**.
* **View:**ده الجزء اللي بيعرض الداتا لليوزر. يعني لو عندك صفحه بتعرض بيانات، كل ده بيتم فال**View** وبيبقي مش مسؤول عن اللوجيك او اي حاجه تاني، هو بس بيعرض الداتا ليوزر.
* **Controller:**ال **Controller** هو المسؤول عن ربط ال **Model** بال **View**. يعني لو ال **View** محتاجة بيانات من ال **Model**، ال **Controller** هو اللي بيجيبها. وكمان لو المستخدم عمل تغيير (زي إضافة أو تعديل للداتا)، ال **Controller** هو اللي بيتعامل مع التغييرات دي وينقلها لل **Model** عشان يتم تنفيذها، وبعدين يعرض ال **View** الداتا الجديده.

إزاي تعمل **MVC؟** ومثال بسيط

**Model:** الكلاس المسؤول عن الداتا.

class Product {

public static function getAll() {

// fetch products from the database

}

public static function create($name) {

// insert a new product into the database

}

}

**View:** الفايل المسؤول عن عرض الداتا.

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Products List</title>

</head>

<body>

<h1>Products</h1>

<!-- Loop through the products array and display each product's name -->

<?php foreach($products as $product): ?>

<p><?php echo $product['name']; ?></p>

<?php endforeach; ?>

</body>

</html>

**Controller:** الكلاس المسؤول عن معالجة الطلبات وتنظيم التفاعل بين الـ Model والـ View.

class ProductController {

public function index() {

$products = Product::getAll(); // Fetch products from the model

include 'views/products.php'; // Load the view to display products

}

public function add() {

$name = $\_POST['name']; // Retrieve product name from form data

Product::create($name); // Call model method to create product

header('Location: products'); // Redirect to the products page

}

}