

## allseeds.php

```

<!-- Top of the table is set here -->
<table class="table table-striped">
    <thead class="thead-dark">
        <tr>
            <th scope="col">Id</th> <!-- One column is set here, each one is a column -->
            <th scope="col">Type</th>
            <th scope="col">Color</th>
            <th scope="col">Stage</th>
            <th scope="col">Location Now</th>
        </tr>
    </thead>
<tbody>

<!-- SQL operations, done in PHP -->
<?php
$sql = "SELECT * FROM `seed_id_table` ORDER BY seed_id DESC";
$result = mysqli_query($conn, $sql);
// Start a while loop, we will do everything inside this loop.
while($row=mysqli_fetch_assoc($result)){
    $seed_id = $row['seed_id'] ;
    $seed_type = $row['seed_type'];
    $seed_color = $row['seed_color'];
    $seed_stage = $row['seed_stage'];
    $seed_location = $row['seed_location'];
    ?>

<!-- We use HTML for each different row, however we need to use PHP inside it to show database data -->
<!-- We set values of rows and columns from the variables we set earlier -->
<tr>
    <th scope="row"><?php echo $seed_id;?></th>

    <td><?php echo seed_name_converter($seed_type); ?></td>

    <!-- To add colored circles according to harvesting color we used if loops.
    This basically checks the variables with each color's string name,
    When it finds the color it concatenate the circle next to it
-->

```

PHP code is a scripting language that enables developers to create dynamic content that interacts with databases. In the context of WordPress, PHP code can be used to manipulate how WordPress generates pages on the front-end of a WordPress site.

The code in the above example first establishes a connection to the WordPress database. Once connected, the code then retrieves all data from the 'seed\_id\_table' table. This data is then used to populate the HTML table that is output on the page.

The code also uses some conditional logic (e.g. if/else statements) to display colored circles next to each seed entry, depending on the color of that particular seed. Finally, the code includes the site's footer at the bottom of the page.

```

<th scope="row"><?php echo $seed_id;?></th>

<td><?php echo seed_name_converter($seed_type); ?></td>

<!-- To add colored circles according to harvesting color we used if loops.
This basically checks the variables with each color's string name,
When it finds the color it concatenate the circle next to it
-->
<td><?php if($seed_color == "Red"){
    $newcolor=$seed_color." 🍅 ";
    echo $newcolor;
}
else if($seed_color == "Green"){
    $newcolor=$seed_color." 🍌 ";
    echo $newcolor;
}
else if($seed_color == "Blue"){
    $newcolor=$seed_color." 🍷 ";
    echo $newcolor;
}
else if($seed_color == "Orange"){
    $newcolor=$seed_color." 🍊 ";
    echo $newcolor;
}
else if($seed_color == "Yellow"){
    $newcolor=$seed_color." 🍋 ";
    echo $newcolor;
}
else if($seed_color == "Brown"){
    $newcolor=$seed_color." 🍫 ";
    echo $newcolor;
}
else if($seed_color == "Black"){
    $newcolor=$seed_color." 🍚 ";
    echo $newcolor;
} ?>
</td>

<td><?php echo $seed_stage; ?></td>

```

The php code above is used to change the color of the text depending on the value of the seed\_color variable. If the value of seed\_color is "Red", then the text will be displayed as red. If the value of seed\_color is "Green", then the text will be displayed as green, and so on. This can be useful for making data easier to read, or for highlighting certain values.