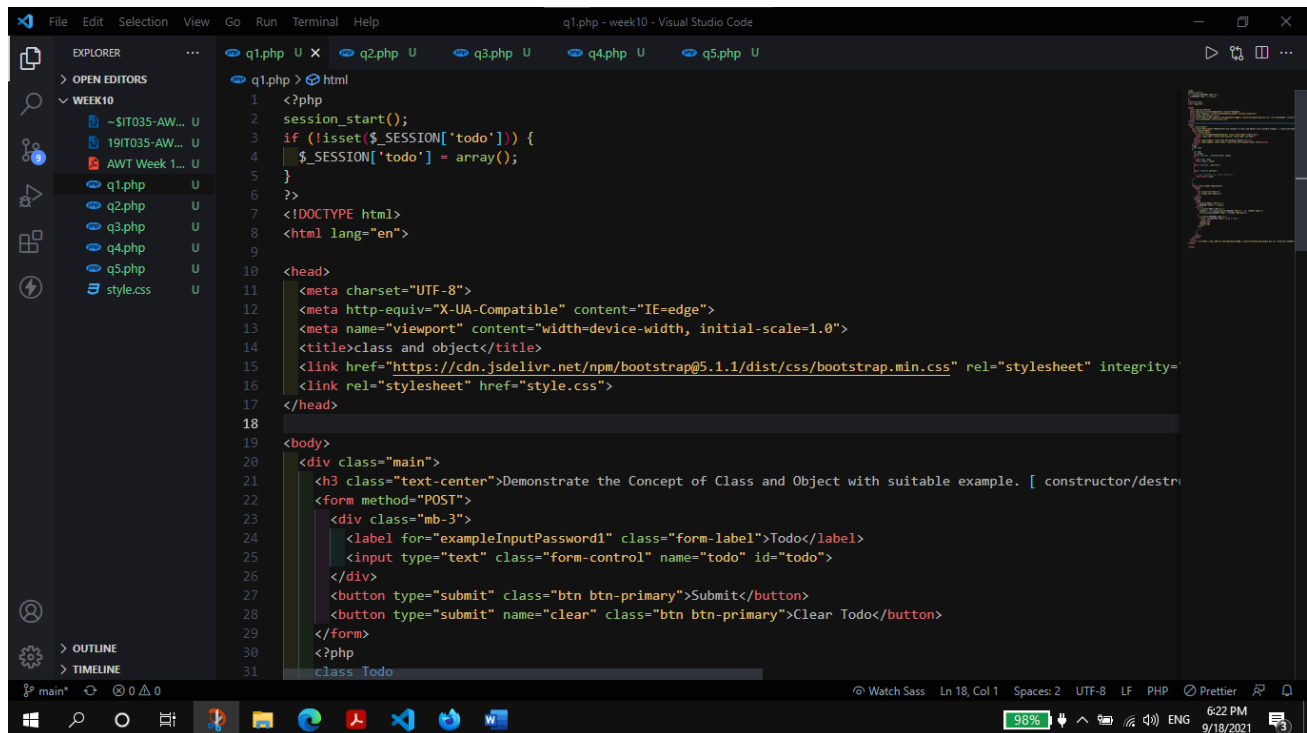


AWT Practical Week - 10 Task

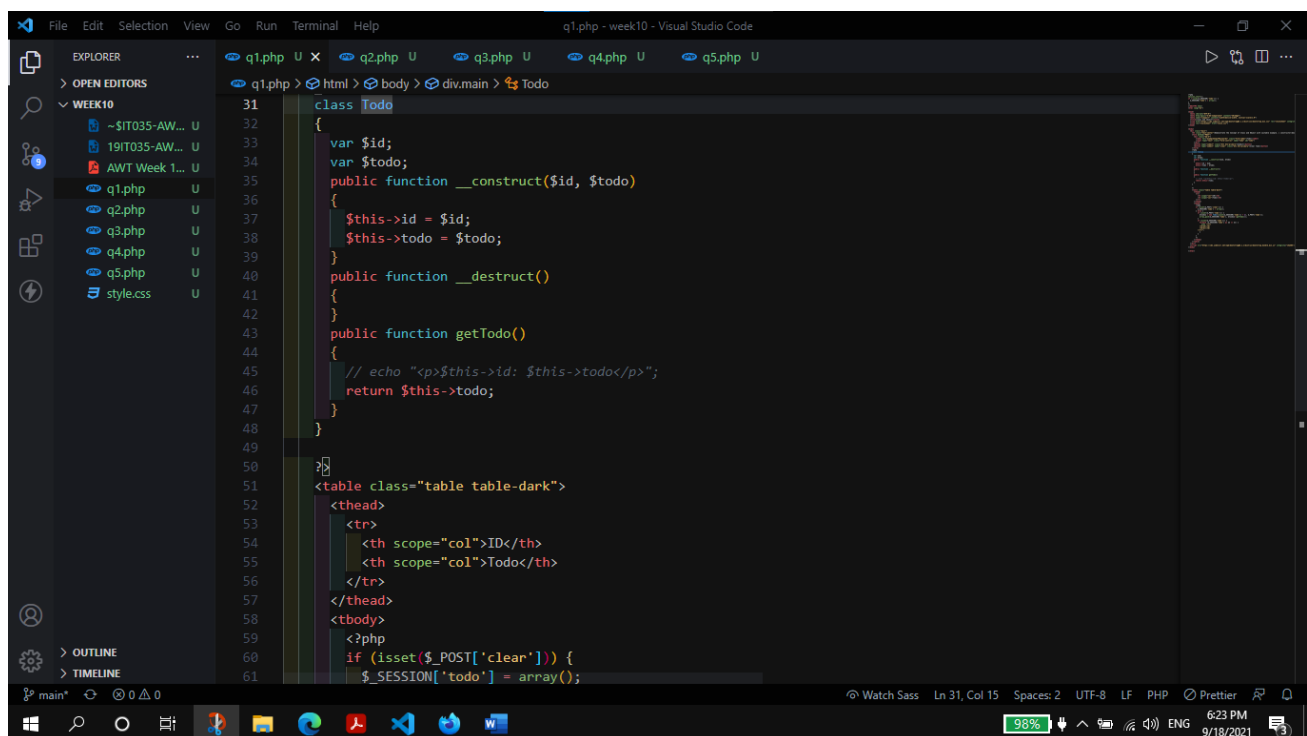
OOP Concepts using PHP

1) Demonstrate the Concept of Class and Object with suitable example. [constructor/destructor]

Code:



```
1 <?php
2 session_start();
3 if (!isset($_SESSION['todo'])) {
4     $_SESSION['todo'] = array();
5 }
6 >?
7 <!DOCTYPE html>
8 <html lang="en">
9
10 <head>
11     <meta charset="UTF-8">
12     <meta http-equiv="X-UA-Compatible" content="IE=edge">
13     <meta name="viewport" content="width=device-width, initial-scale=1.0">
14     <title>class and object</title>
15     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet" integrity=
16     <link rel="stylesheet" href="style.css">
17 </head>
18
19 <body>
20     <div class="main">
21     <h3 class="text-center">Demonstrate the Concept of Class and Object with suitable example. [ constructor/destru
22     <form method="POST">
23         <div class="mb-3">
24             <label for="exampleInputPassword1" class="form-label">Todo</label>
25             <input type="text" class="form-control" name="todo" id="todo">
26         </div>
27         <button type="submit" class="btn btn-primary">Submit</button>
28         <button type="submit" name="clear" class="btn btn-primary">Clear Todo</button>
29     </form>
30 <?php
31 class Todo
```



```
31 class Todo
32 {
33     var $id;
34     var $todo;
35     public function __construct($id, $todo)
36     {
37         $this->id = $id;
38         $this->todo = $todo;
39     }
40     public function __destruct()
41     {
42     }
43     public function getTodo()
44     {
45         // echo "<p>$this->id: $this->todo</p>";
46         return $this->todo;
47     }
48 }
49
50
51 <table class="table table-dark">
52     <thead>
53     <tr>
54         <th scope="col">ID</th>
55         <th scope="col">Todo</th>
56     </tr>
57     </thead>
58     <tbody>
59     <?php
60     if (isset($_POST['clear'])) {
61         $_SESSION['todo'] = array();
```

```

56 </tr>
57 </thead>
58 <tbody>
59 <?php
60 if (isset($_POST['clear'])) {
61     $_SESSION['todo'] = array();
62 } else {
63     if (isset($_POST['todo'])) {
64         $todoo = new Todo((count($_SESSION['todo']) + 1), $_POST['todo']);
65         array_push($_SESSION['todo'], $todoo->getTodo());
66     }
67     if (isset($_SESSION['todo'])) {
68         foreach ($_SESSION['todo'] as $k => $v) {
69             echo "<tr>
70                 <td>$k</td>
71                 <td>$v</td>
72             </tr>";
73         }
74     }
75 }
76 </tbody>
77 </table>
78 </div>
79 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-/bQ
80 </body>
81 </html>

```

Output:

Polymorphism

https://localhost/week10/q1.php

Getting Started Most Visited SGP Learning AWT Application Express Lo... Template Homepage - Scott Bar... Flat UI Palette v1 | ... Color Chart - HTML ... Other Bookmarks

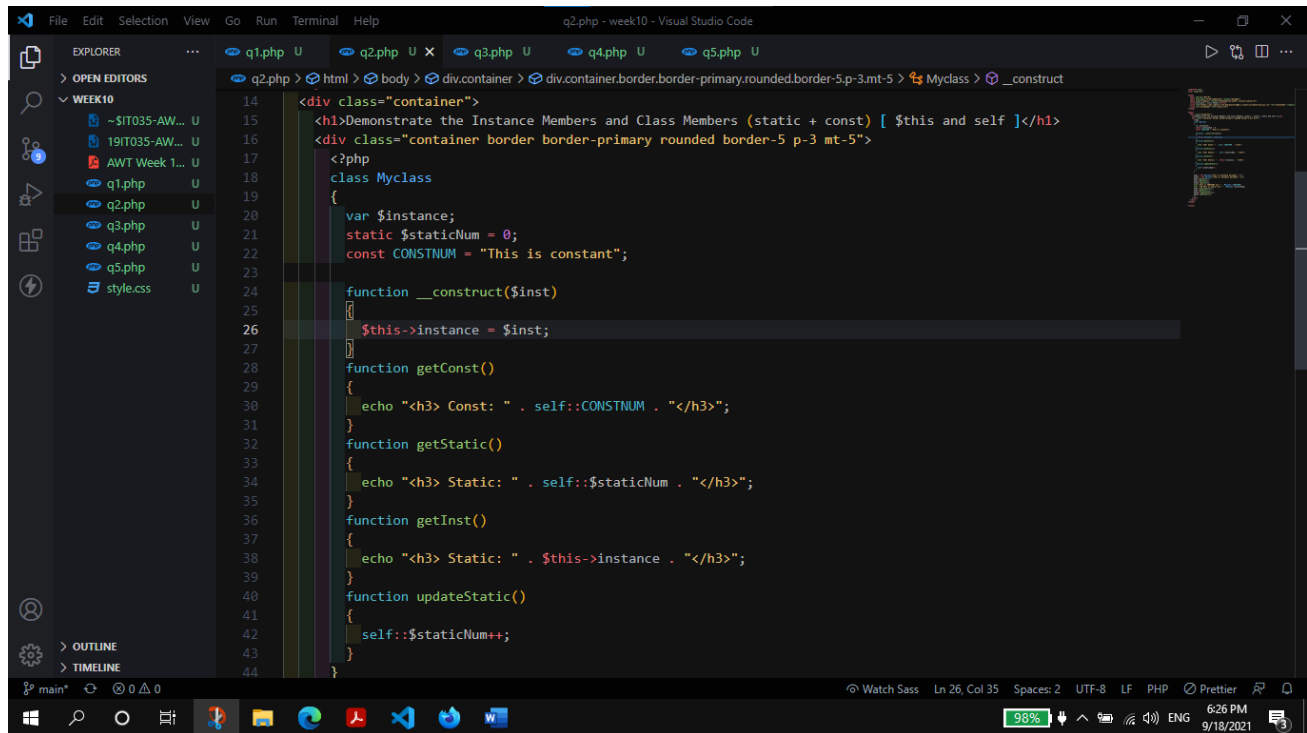
Todo

ID	Todo
0	AWT
1	PY
2	SGP

98% 6:24 PM 9/18/2021

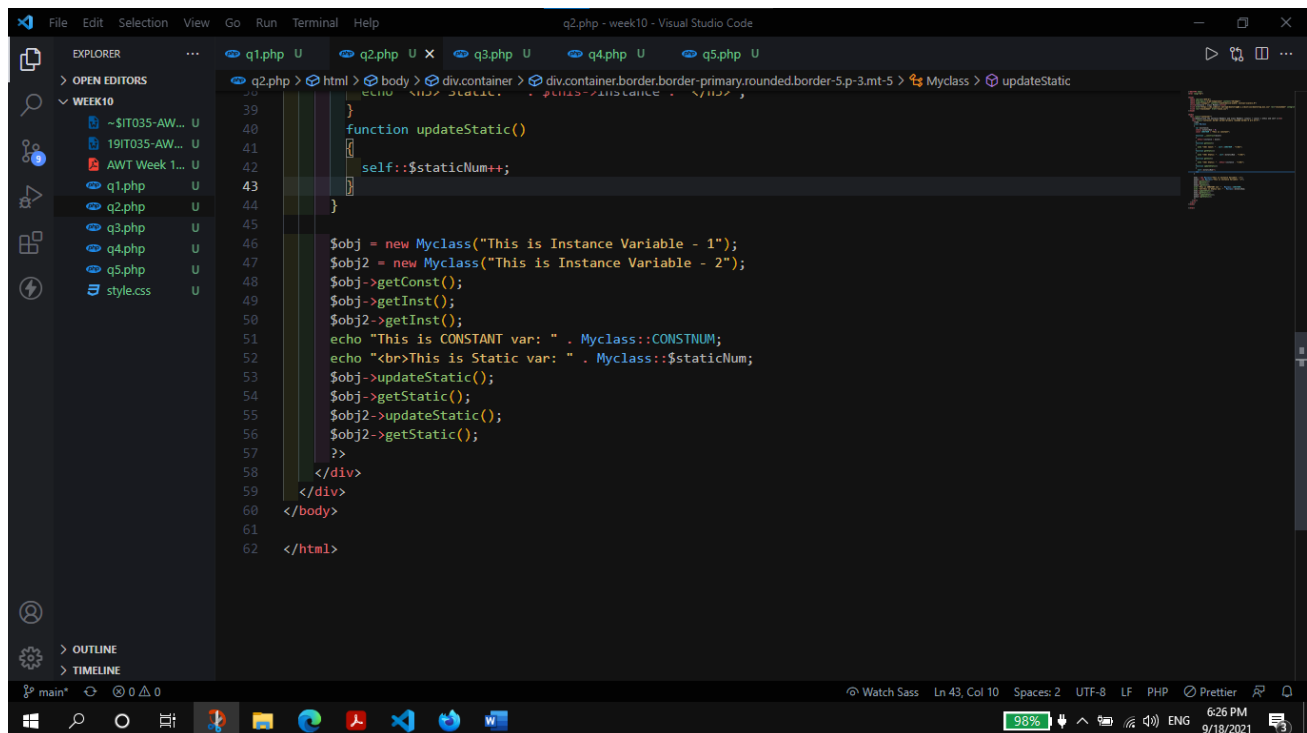
2) Demonstrate the Instance Members and Class Members (static + const) [\$this and self]

Code:



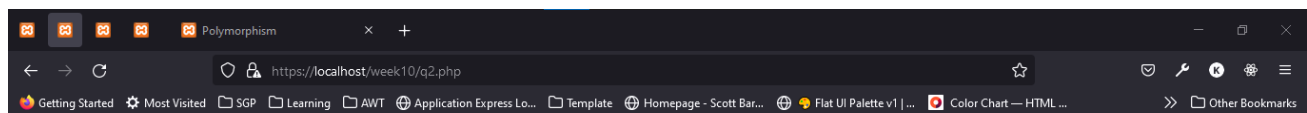
```

14 <div class="container">
15 <h1>Demonstrate the Instance Members and Class Members (static + const) [ $this and self ]</h1>
16 <div class="container border border-primary rounded border-5 p-3 mt-5">
17 <?php
18 class Myclass
19 {
20     var $instance;
21     static $staticNum = 0;
22     const CONSTNUM = "This is constant";
23
24     function __construct($inst)
25     {
26         $this->instance = $inst;
27     }
28     function getConst()
29     {
30         echo "<h3> Const: " . self::CONSTNUM . "</h3>";
31     }
32     function getStatic()
33     {
34         echo "<h3> Static: " . self::$staticNum . "</h3>";
35     }
36     function getInst()
37     {
38         echo "<h3> Static: " . $this->instance . "</h3>";
39     }
40     function updateStatic()
41     {
42         self::$staticNum++;
43     }
44 }
  
```



```

39     }
40     function updateStatic()
41     {
42         self::$staticNum++;
43     }
44 }
45
46 $obj = new Myclass("This is Instance Variable - 1");
47 $obj2 = new Myclass("This is Instance Variable - 2");
48 $obj->getConst();
49 $obj->getInst();
50 $obj2->getInst();
51 echo "This is CONSTANT var: " . Myclass::CONSTNUM;
52 echo "<br>This is Static var: " . Myclass::$staticNum;
53 $obj->updateStatic();
54 $obj->getStatic();
55 $obj2->updateStatic();
56 $obj2->getStatic();
57 ?>
58 </div>
59 </div>
60 </body>
61 </html>
  
```

Output:

Demonstrate the Instance Members and Class Members (static + const) [\$this and self]

Const: This is constant

Static: This is Instance Variable - 1

Static: This is Instance Variable - 2

This is CONSTANT var: This is constant

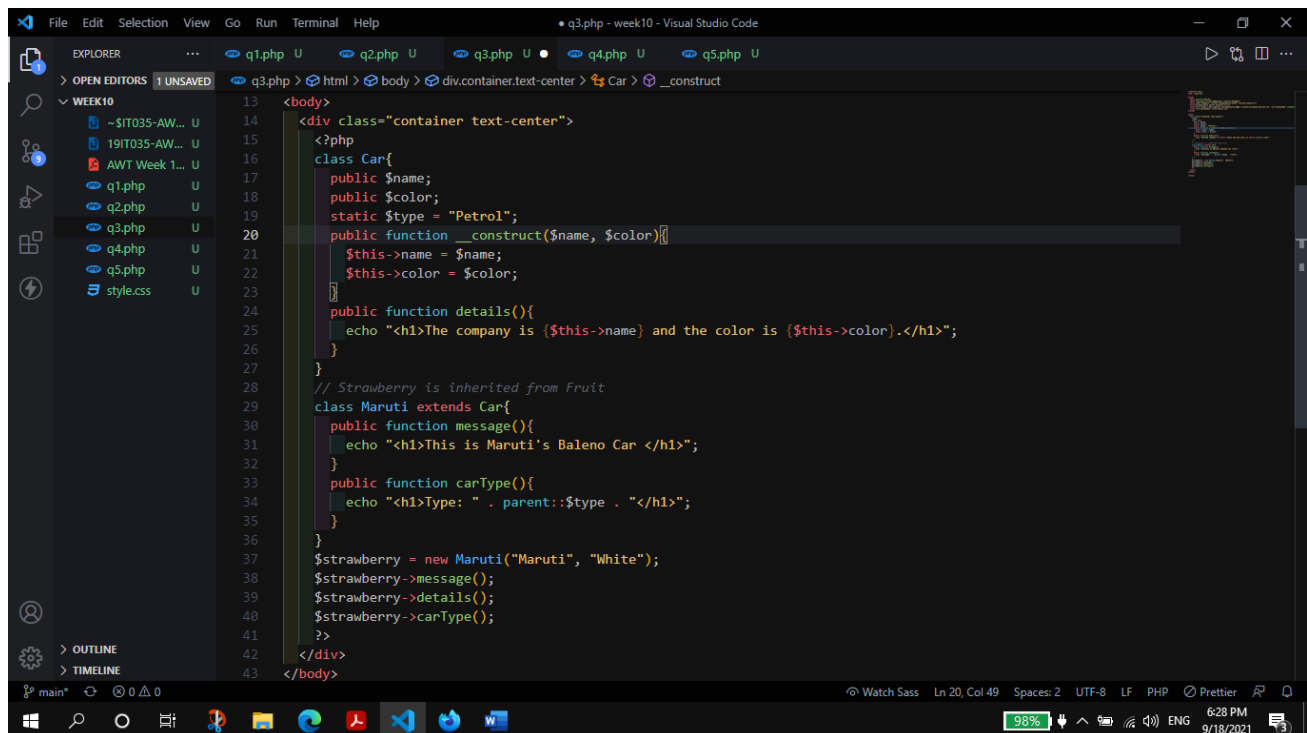
This is Static var: 0

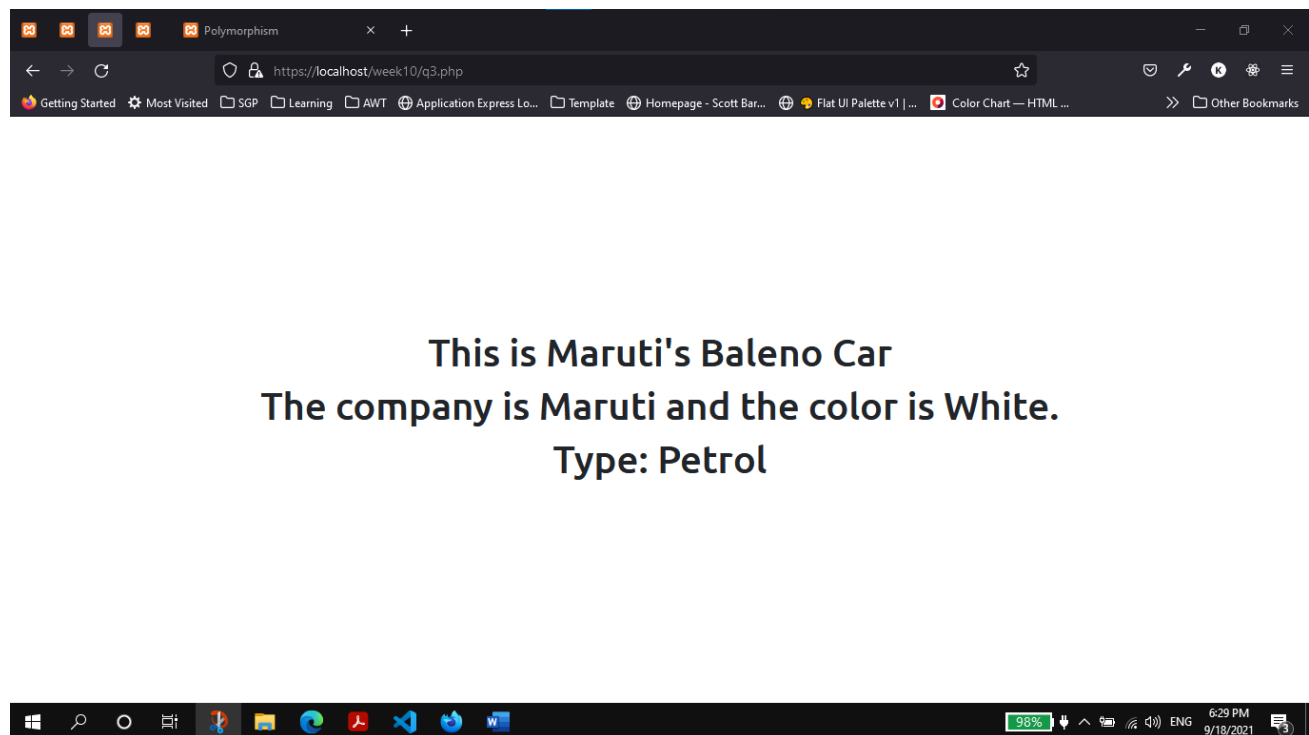
Static: 1

Static: 2

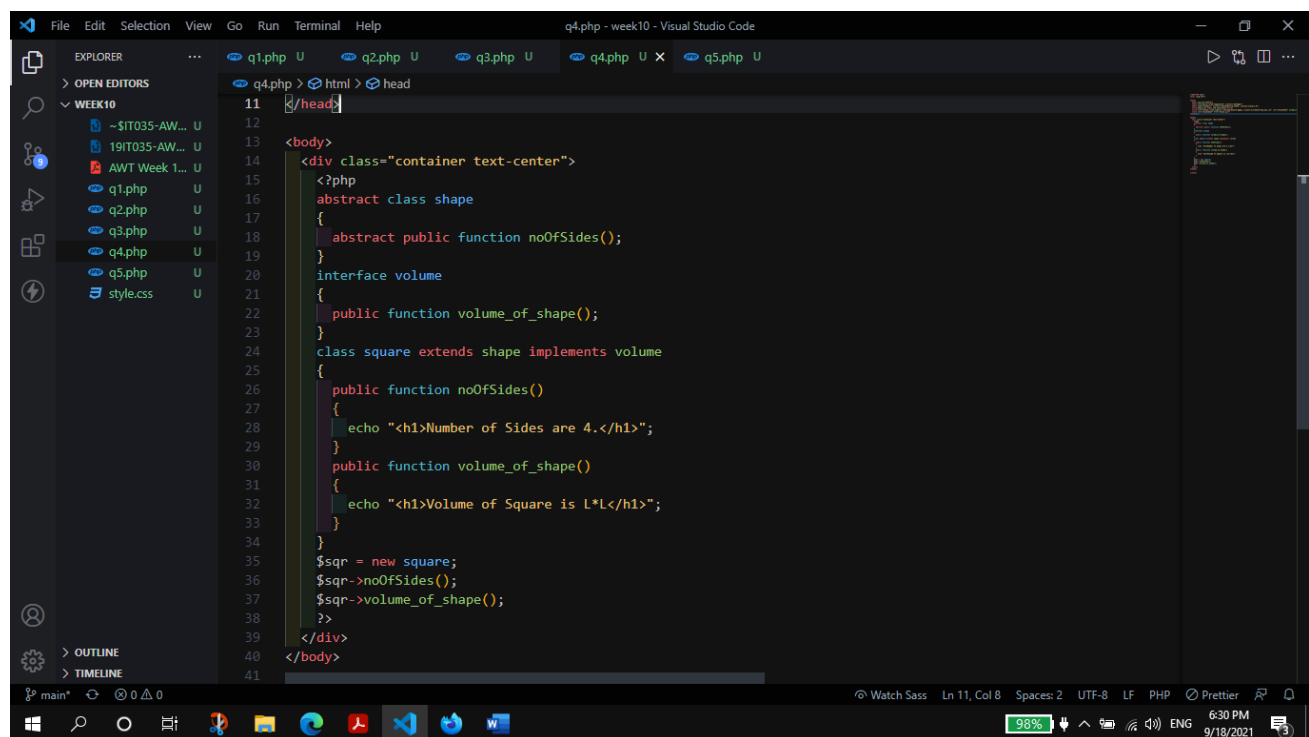


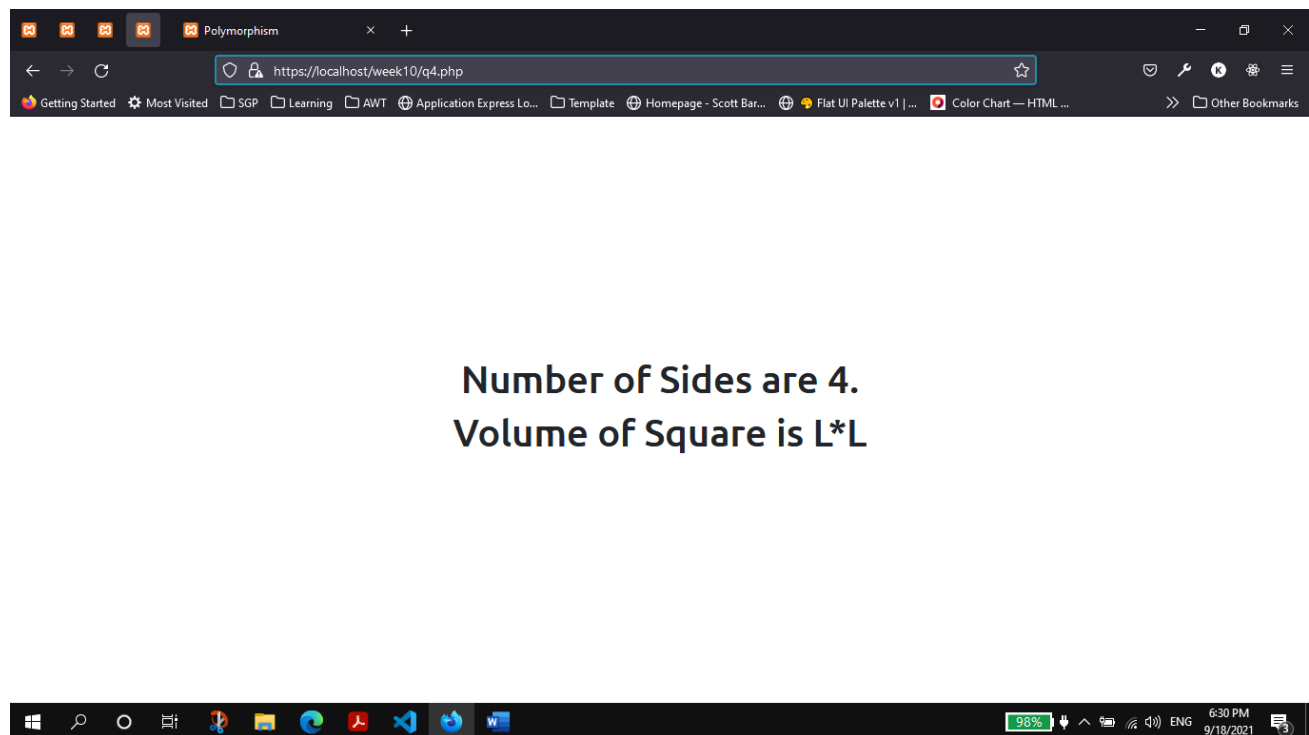
3) Demonstrate the inheritance/s [parent keyword].

Code:

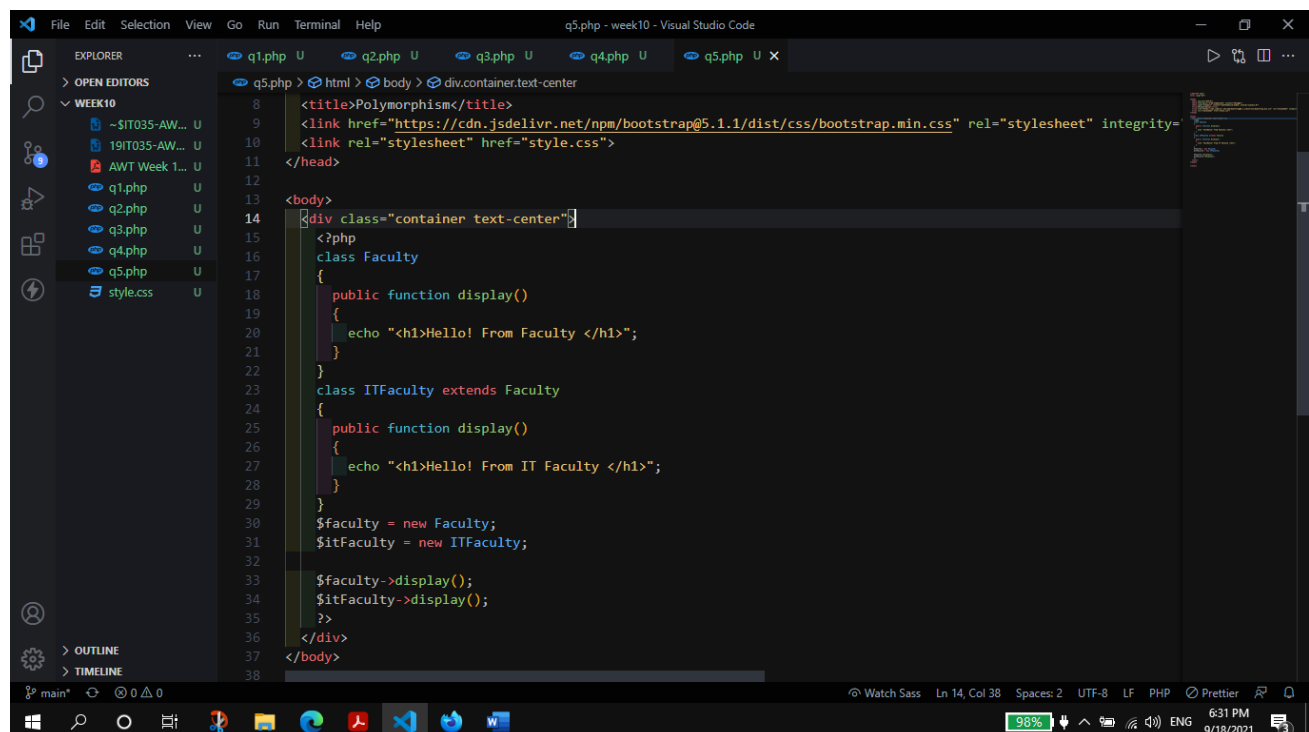
Output:

4) Demonstrate the Abstract Class and Interfaces with suitable examples.

Code:

Output:

5) Demonstrate Polymorphism with suitable examples.

Code:

Output: