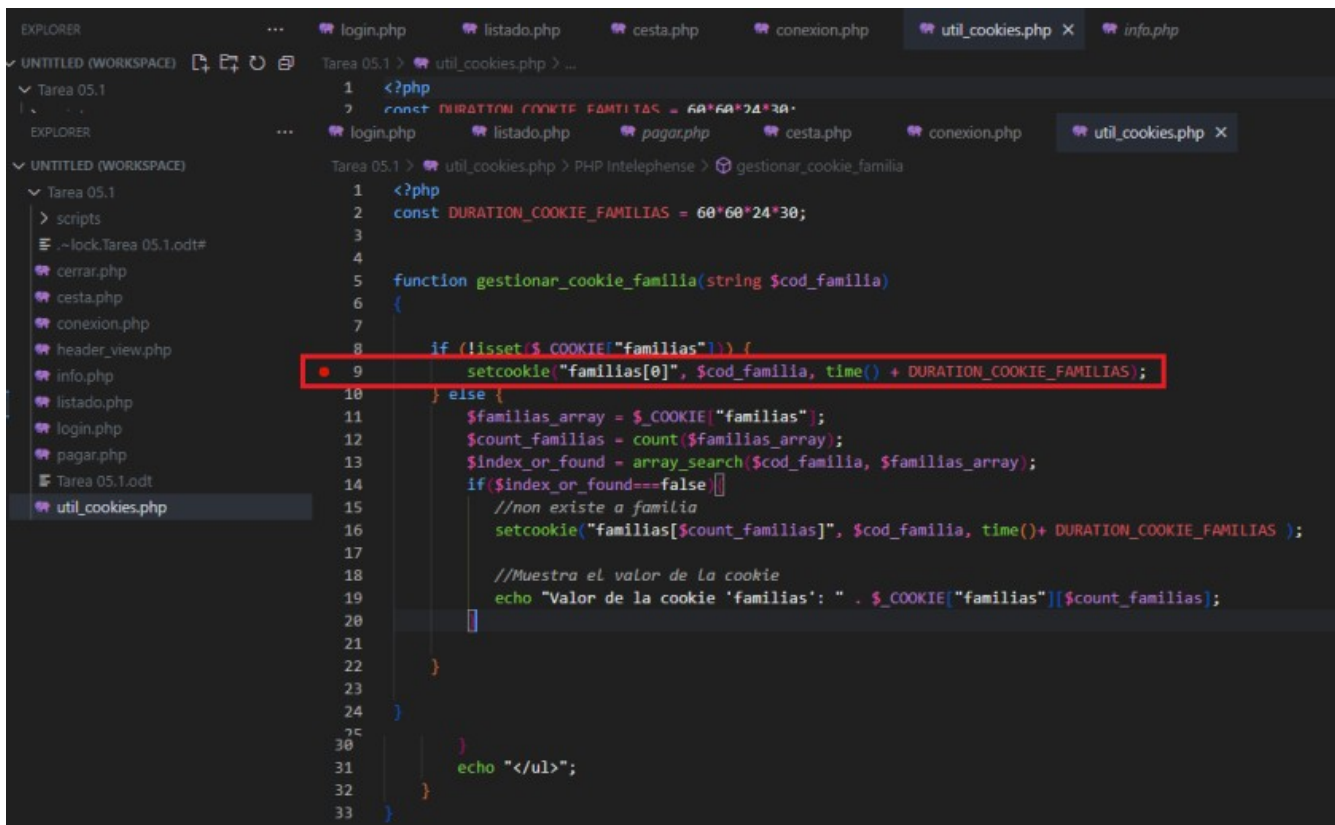


5- Asegúrate de que tienes bien configurado XDebug y que eres capaz de realizar depuración en tu código.

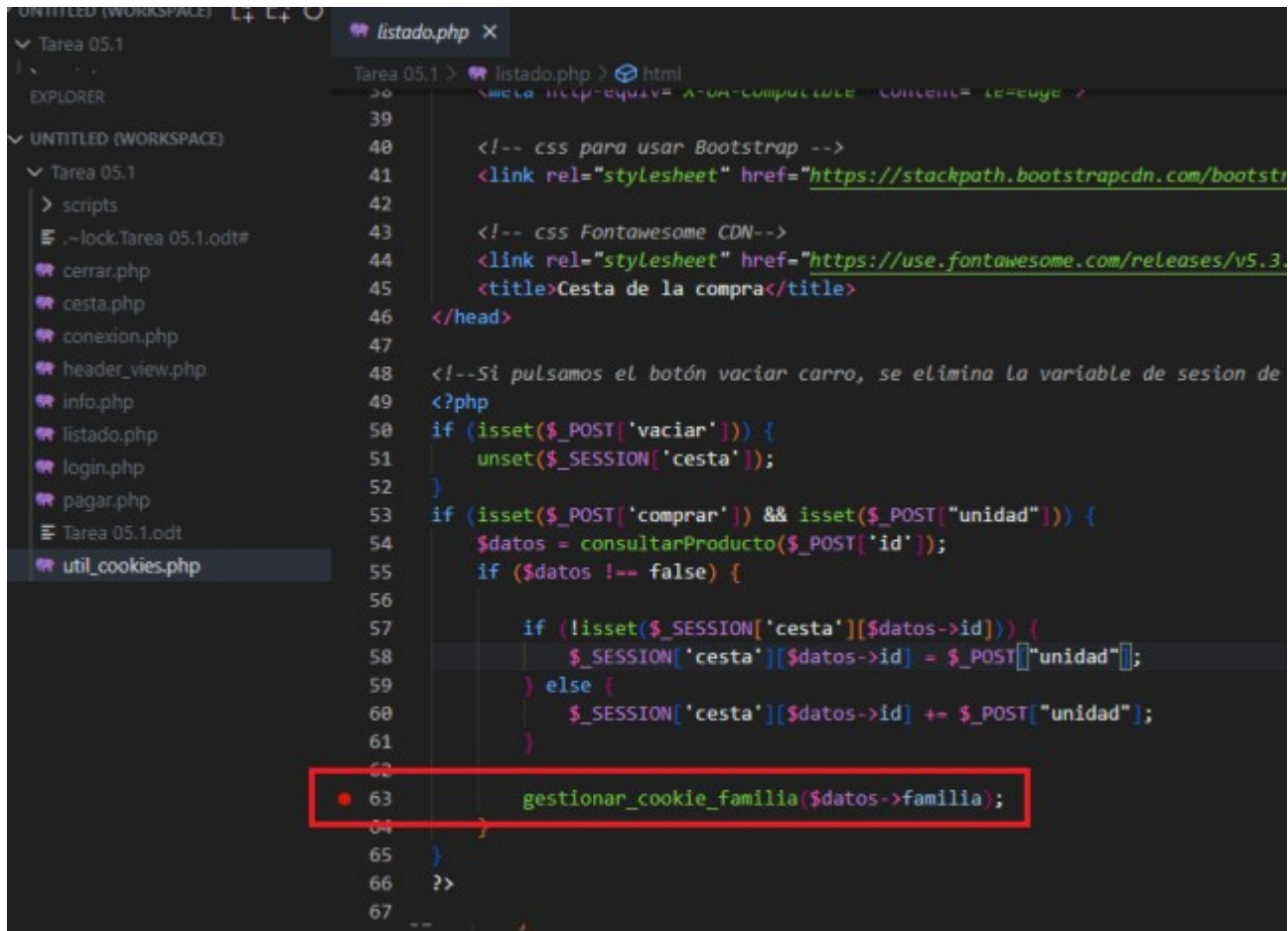
Sitúa puntos de ruptura en las instrucciones en las que añades una nueva cookie para registrar los códigos de familia

El primer punto de ruptura está en util\_cookies.php, concretamente en la línea en la que se establece la nueva cookie para registrar los códigos de familia (línea 9)



```
1  <?php
2  const DURATION_COOKIE_FAMILIAS = 60*60*24*30;
3
4
5  function gestionar_cookie_familia(string $cod_familia)
6  {
7
8      if (!isset($_COOKIE["familias"])) {
9          setcookie("familias[0]", $cod_familia, time() + DURATION_COOKIE_FAMILIAS);
10     } else {
11         $familias_array = $_COOKIE["familias"];
12         $count_familias = count($familias_array);
13         $index_or_found = array_search($cod_familia, $familias_array);
14         if ($index_or_found === false) {
15             //non existe a familia
16             setcookie("familias[$count_familias]", $cod_familia, time() + DURATION_COOKIE_FAMILIAS);
17
18             //Muestra el valor de la cookie
19             echo "Valor de la cookie 'familias': " . $_COOKIE["familias"][$count_familias];
20         }
21     }
22 }
23
24
25
26
27
28
29
30
31     echo "</ul>";
32 }
33 }
```

El segundo punto de ruptura es en listado.php, justo después de darle al botón de añadir, se gestionan las cookies de familia con la función `gestionar_cookie_familia`. Esto ocurre después de añadir productos al carrito.



```
38 <meta http-equiv="X-UA-Compatible" content="ie=edge" />
39
40 <!-- css para usar Bootstrap -->
41 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">
42
43 <!-- css Fontawesome CDN -->
44 <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.3.1/css/all.css">
45 <title>Cesta de la compra</title>
46 </head>
47
48 <!-- Si pulsamos el botón vaciar carro, se elimina la variable de sesion de
49 <?php
50 if (isset($_POST['vaciar'])) {
51     unset($_SESSION['cesta']);
52 }
53 if (isset($_POST['comprar']) && isset($_POST['unidad'])) {
54     $datos = consultarProducto($_POST['id']);
55     if ($datos !== false) {
56
57         if (!isset($_SESSION['cesta'][$datos->id])) {
58             $_SESSION['cesta'][$datos->id] = $_POST['unidad'];
59         } else {
60             $_SESSION['cesta'][$datos->id] += $_POST['unidad'];
61         }
62
63         gestionar_cookie_familia($datos->familia);
64     }
65 }
66 ?>
67 --
```

y comprueba en tiempo de depuración si `$_COOKIE` contiene el valor que acabas de añadir a la cookie, justo después de crearla. Incluye una captura del depurador en el Visual Studio Code donde se observe:

- la línea de código que se acaba de ejecutar (señálala expresamente o indica en un texto de qué línea se trata)
- el valor de la variable que contenga el código de familia que se acaba de añadir
- el contenido del array superglobal `$_COOKIE`. Explica en la propia imagen por qué no está presente el código de familia que se acaba de añadir a la cookie

```

RUN AND DEBUG Listen for Xdebug (Ta... listado.php x
VARIABLES
$host: "localhost"
$pass: "secreto"
$path: "/listado.php"
$relativePath: "."
> $stmt: PDOStatement
$user: "gestor"
✓ Superglobals
> $_GET: array(0)
> $_POST: array(3)
✓ ✓ $_COOKIE: array(2)
  PHPSESSID: "o6g2r3e3jgvq8me9o26p426e10"
  > familias: array(7)
> $_FILES: array(0)
> $_ENV: array(0)
> $_REQUEST: array(3)
> $_SERVER: array(36)
  $path: "/listado.php"
  $relativePath: "."
  $fullPath: "C:/Users/Amaho/Desktop/DAW Dual..."
  $consulta: "select id, nombre, pvp from pro..."
> $stmt: PDOStatement
WATCH

36 <meta charset="UTF-8">
37 <meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
38 <meta http-equiv="X-UA-Compatible" content="ie=edge">
39
40 <!-- css para usar Bootstrap -->
41 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-Vi..."
42
43 <!-- css Fontawesome CDN -->
44 <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.3.1/css/all.css" integrity="sha384-mzrmESqonljUremFsq..."
45 <title>Cesta de la compra</title>
46 </head>
47
48 <!--Si pulsamos el botón vaciar carro, se elimina la variable de sesion de la cesta-->
49 <?php
50 if (isset($_POST['vaciar'])) { $_POST = array(3)
51   unset($_SESSION['cesta']); $_SESSION = array(2)
52 }
53 if (isset($_POST['comprar']) && isset($_POST['unidad'])) { $_POST = array(3)
54   $datos = consultarProducto($_POST['id']); $datos = stdClass, $_POST = array(3)
55   if ($datos != false) { $datos = stdClass
56     if (!isset($_SESSION['cesta'][$datos->id])) { $_SESSION = array(2), $datos = stdClass
57       $_SESSION['cesta'][$datos->id] = $_POST['unidad']; $_SESSION = array(2), $datos = stdClass, $_POST = array(3)
58     } else {
59       $_SESSION['cesta'][$datos->id] += $_POST['unidad']; $_SESSION = array(2), $datos = stdClass, $_POST = array(3)
60     }
61   }
62
63   gestionar_cookie_familia $datos->familia; $datos = stdClass
64 }
65
```

Lo único que aparece en `$_COOKIE` es la cookie de sesión, no la de familias.

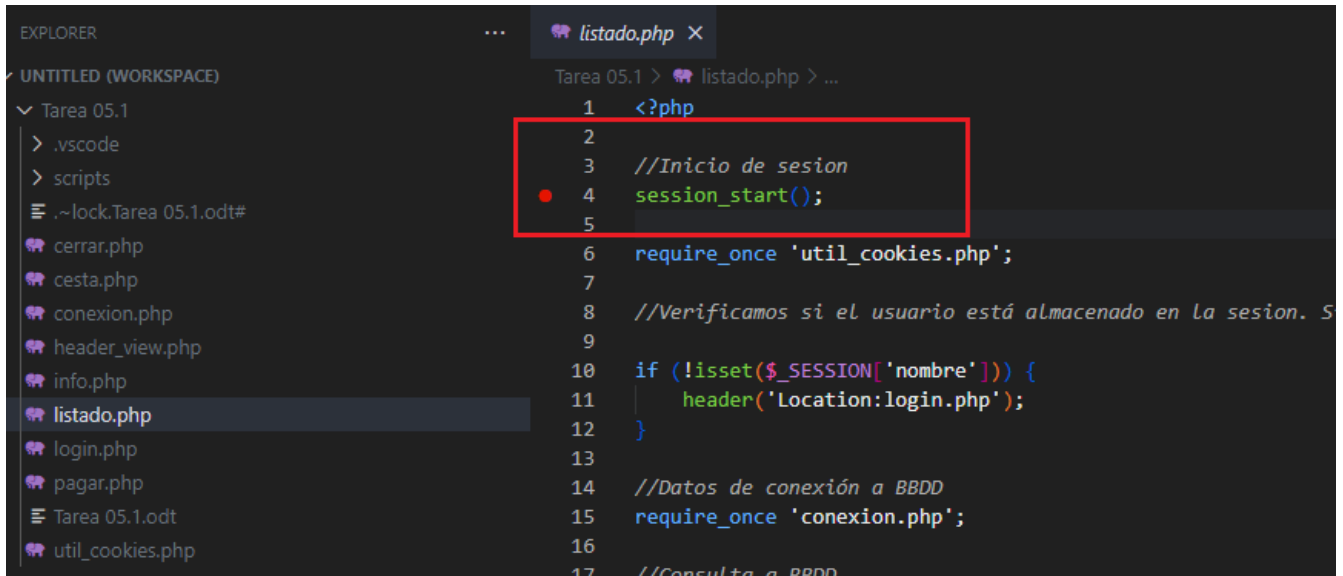
El motivo de esto, al parecer, es que PHP no actualiza `$_COOKIE` en tiempo real durante la ejecución de un script.

Le añado un echo después de setcookie para comprobar el valor de la cookie de familias:

```
util_cookies.php X
Tarea 05.1 > util_cookies.php > PHP Intelephense > gestionar_cookie_familia
1  <?php
2  const DURATION_COOKIE_FAMILIAS = 60*60*24*30;
3
4
5  function gestionar_cookie_familia(string $cod_familia)
6  {
7
8      if (!isset($_COOKIE["familias"])) {
9          setcookie("familias[0]", $cod_familia, time() + DURATION_COOKIE_FAMILIAS);
10     } else {
11         $familias_array = $_COOKIE["familias"];
12         $count_familias = count($familias_array);
13         $index_or_found = array_search($cod_familia, $familias_array);
14         if($index_or_found===false){
15             //non existe a familia
16             setcookie("familias[$count_familias]", $cod_familia, time()+ DURATION_COOKIE_FAMILIAS );
17
18             //Muestra el valor de la cookie
19             echo "Valor de la cookie 'familias': " . $_COOKIE["familias"][$count_familias];
20         }
21     }
22 }
23
24 }
```

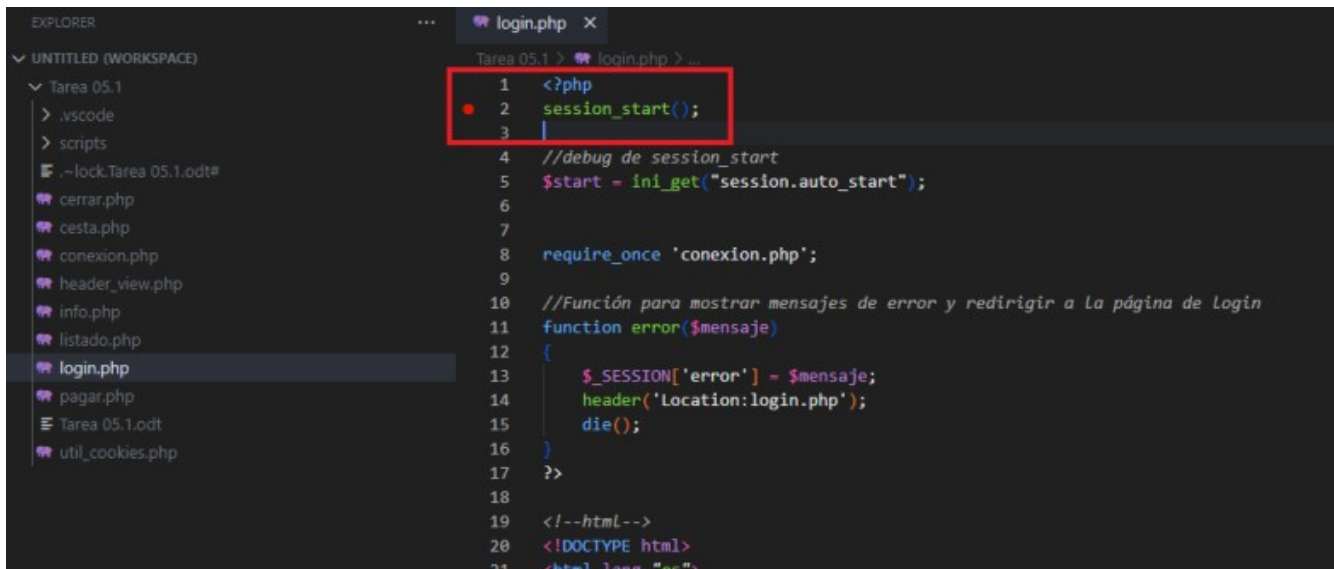
6- Sitúa puntos de ruptura en la instrucción session\_start(). Muestra en sendas capturas:

### Listado.php



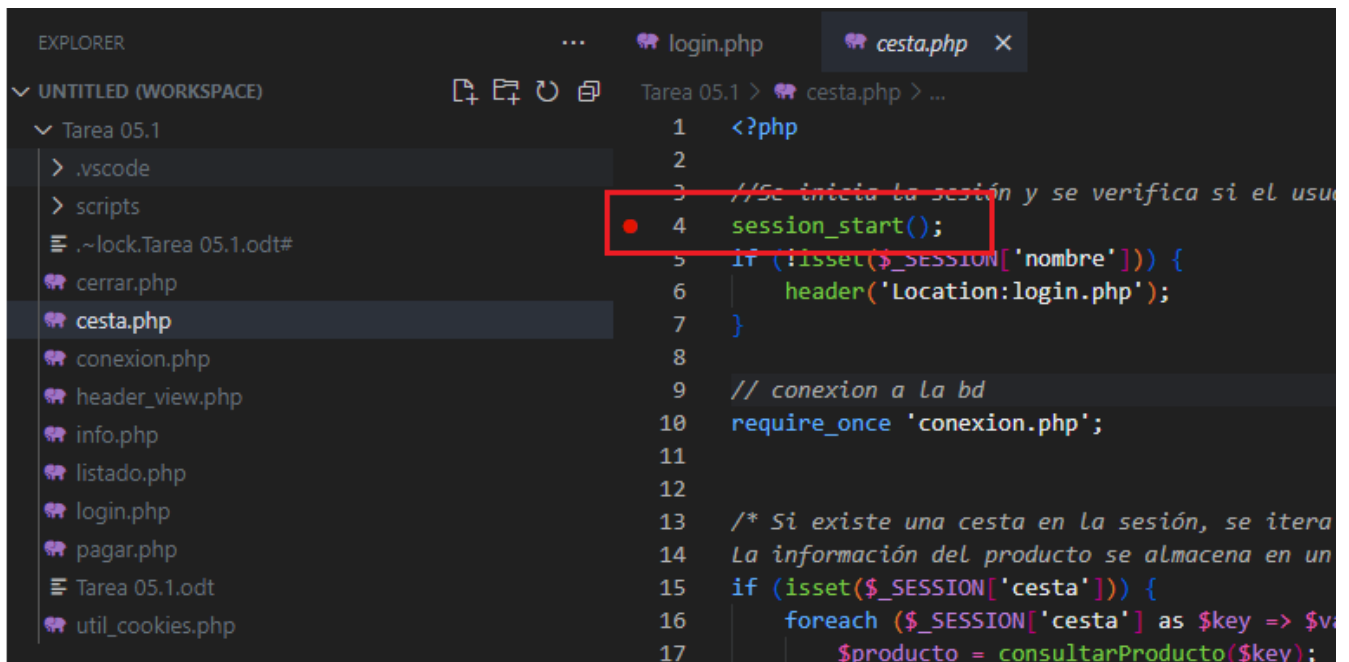
```
1 <?php
2
3 //Inicio de sesion
4 session_start();
5
6 require_once 'util_cookies.php';
7
8 //Verificamos si el usuario está almacenado en la sesion. S
9
10 if (!isset($_SESSION['nombre'])) {
11     header('Location:login.php');
12 }
13
14 //Datos de conexión a BBDD
15 require_once 'conexion.php';
16
17 //Consulta a BBDD
```

### login.php



```
1 <?php
2 session_start();
3
4 //debug de session_start
5 $start = ini_get("session.auto_start");
6
7
8 require_once 'conexion.php';
9
10 //Función para mostrar mensajes de error y redirigir a la página de Login
11 function error($mensaje)
12 {
13     $_SESSION['error'] = $mensaje;
14     header('Location:login.php');
15     die();
16 }
17 ?>
18
19 <!--html-->
20 <!DOCTYPE html>
21 <html lang="es">
```

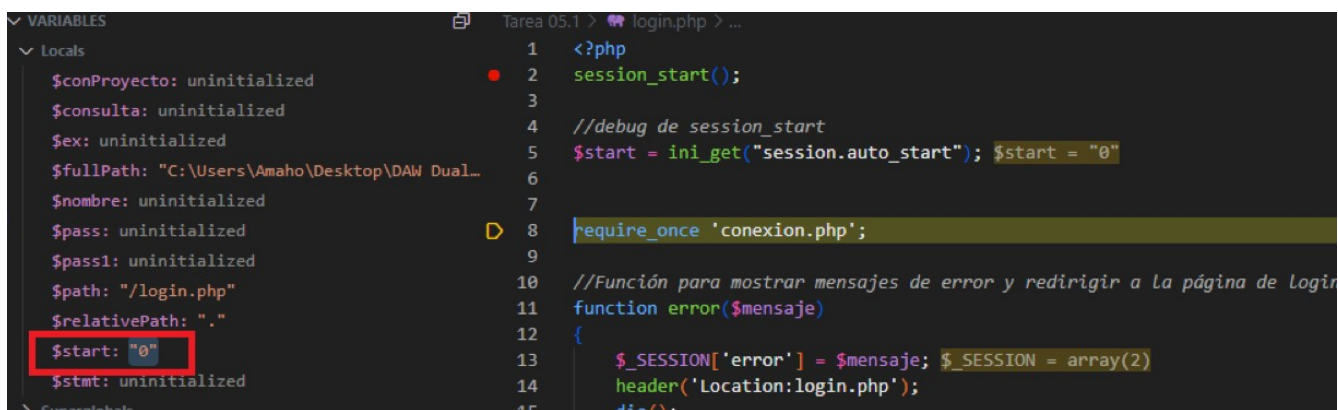
## Cesta.php



```
1 <?php
2
3 //Se inicia la sesión y se verifica si el usu
4 session_start();
5 if (!isset($_SESSION['nombre'])) {
6     header('Location:login.php');
7 }
8
9 // conexion a la bd
10 require_once 'conexion.php';
11
12
13 /* Si existe una cesta en la sesión, se itera
14 La información del producto se almacena en un
15 if (isset($_SESSION['cesta'])) {
16     foreach ($_SESSION['cesta'] as $key => $v
17         $producto = consultarProducto($key);
```

a) valor de la directiva session.auto\_start

## Login.php



```
1 <?php
2 session_start();
3
4 //debug de session_start
5 $start = ini_get("session.auto_start"); $start = "0"
6
7
8 require_once 'conexion.php';
9
10 //Función para mostrar mensajes de error y redirigir a la página de Login
11 function error($mensaje)
12 {
13     $_SESSION['error'] = $mensaje; $_SESSION = array(2)
14     header('Location:login.php');
15     die();
```

VARIABLES

Locals

- \$conProyecto: uninitialized
- \$consulta: uninitialized
- \$ex: uninitialized
- \$fullPath: "C:\Users\Amaho\Desktop\DAW Dual..."
- \$nombre: uninitialized
- \$pass: uninitialized
- \$pass1: uninitialized
- \$path: "/login.php"
- \$relativePath: "."
- \$start: "0"**
- \$stmt: uninitialized

- b) antes de ejecutarse session\_start(), ¿existe \$\_SESSION? ¿Por qué?  
c) justo después de ejecutarse session\_start() ¿existe \$\_SESSION? ¿Está vacía? ¿Por qué?

Antes de ejecutarse session\_start(), \$\_SESSION existe, pero está vacío, porque PHP inicializa automáticamente la variable \$\_SESSION como array vacío antes de llamar a session\_start(). Después de ejecutarse, la variable no está vacía, ya tiene un valor establecido.

The screenshot shows a PHP IDE with a file named 'login.php'. The code is as follows:

```
1 <?php
2 session_start();
3
4 //debug de session_start
5 $start = ini_get("session.auto_start"); $start = uninitialized
6
7
8 require_once 'conexion.php';
9
10 //Función para mostrar mensajes de error y redirigir a la página de login
11 function error($mensaje)
12 {
13     $_SESSION['error'] = $mensaje;
14     header('Location:login.php');
15     die();
16 }
```

The IDE's 'VARIABLES' panel on the left shows the state of variables. Under 'Locals', the following variables are listed, all with the value 'uninitialized':

- \$conProyecto: uninitialized
- \$consulta: uninitialized
- \$ex: uninitialized
- \$fullPath: "C:\Users\Amaho\Desktop\DAW Dual..."
- \$nombre: uninitialized
- \$pass: uninitialized
- \$pass1: uninitialized
- \$path: "/login.php"
- \$relativePath: "."
- \$start: uninitialized** (highlighted with a red box)
- \$stmt: uninitialized

Under 'Superglobals', the variable \$\_SESSION is listed with the value 'uninitialized'.