**Practical-7**

**OOP with PHP -1**

1. **Create a php page which takes the student’s name and marks of three subjects from the user. Manage all information via session and display all information on the next page.**

**Code:**

**index.php**

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <title>Bank Account Operations</title>  </head>  <body>  <h1>Bank Account Operations</h1>    <form method="post" action="process.php">  <label for="account\_number">Account Number:</label>  <input type="text" name="account\_number" required><br><br>    <label for="current\_balance">Current Balance:</label>  <input type="number" name="current\_balance" required><br><br>    <label for="amount">Amount:</label>  <input type="number" name="amount" required><br><br>    <input type="submit" name="deposit" value="Deposit">  <input type="submit" name="withdraw" value="Withdraw">  </form>  </body>  </html> |

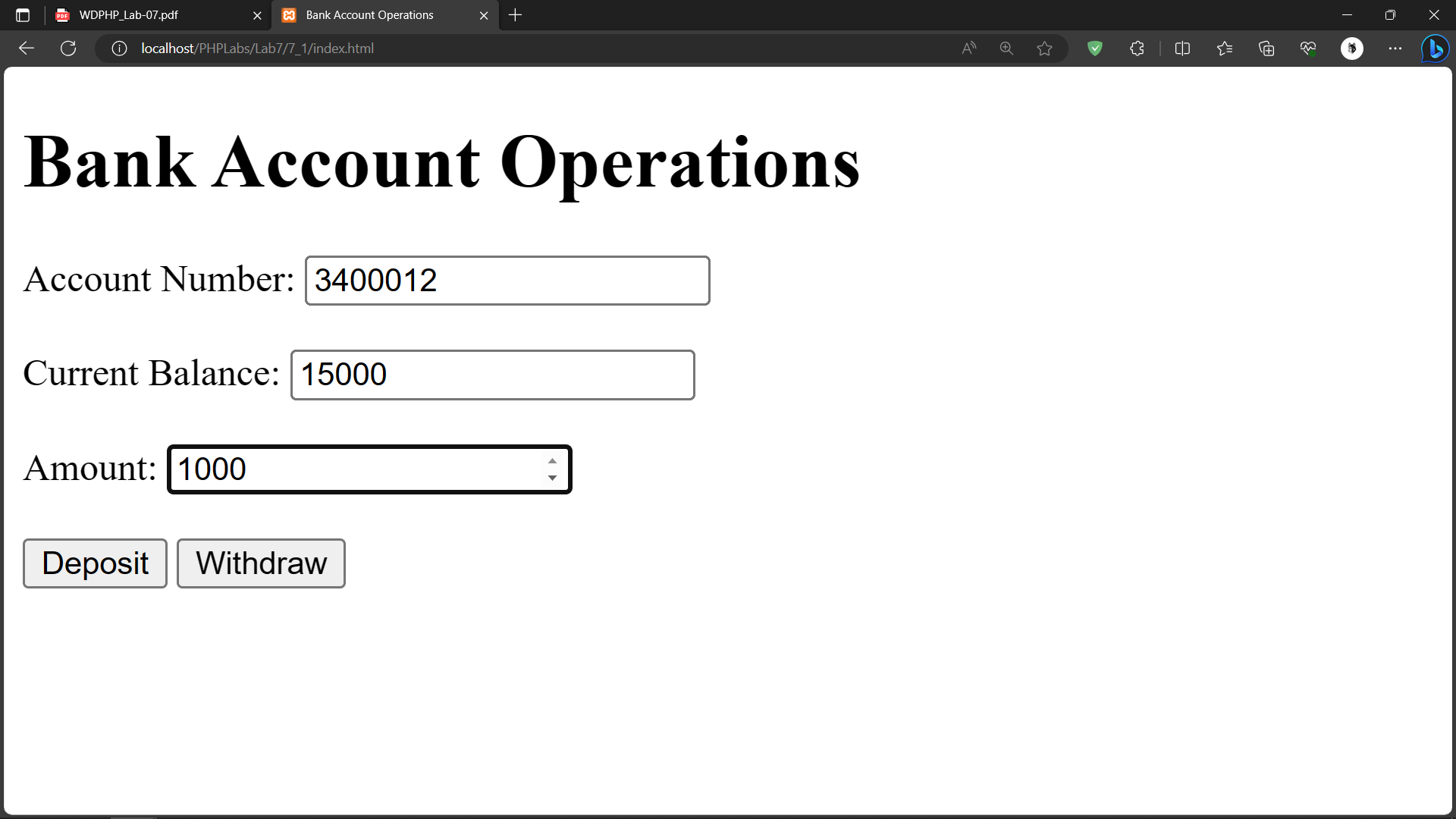
**process.php**

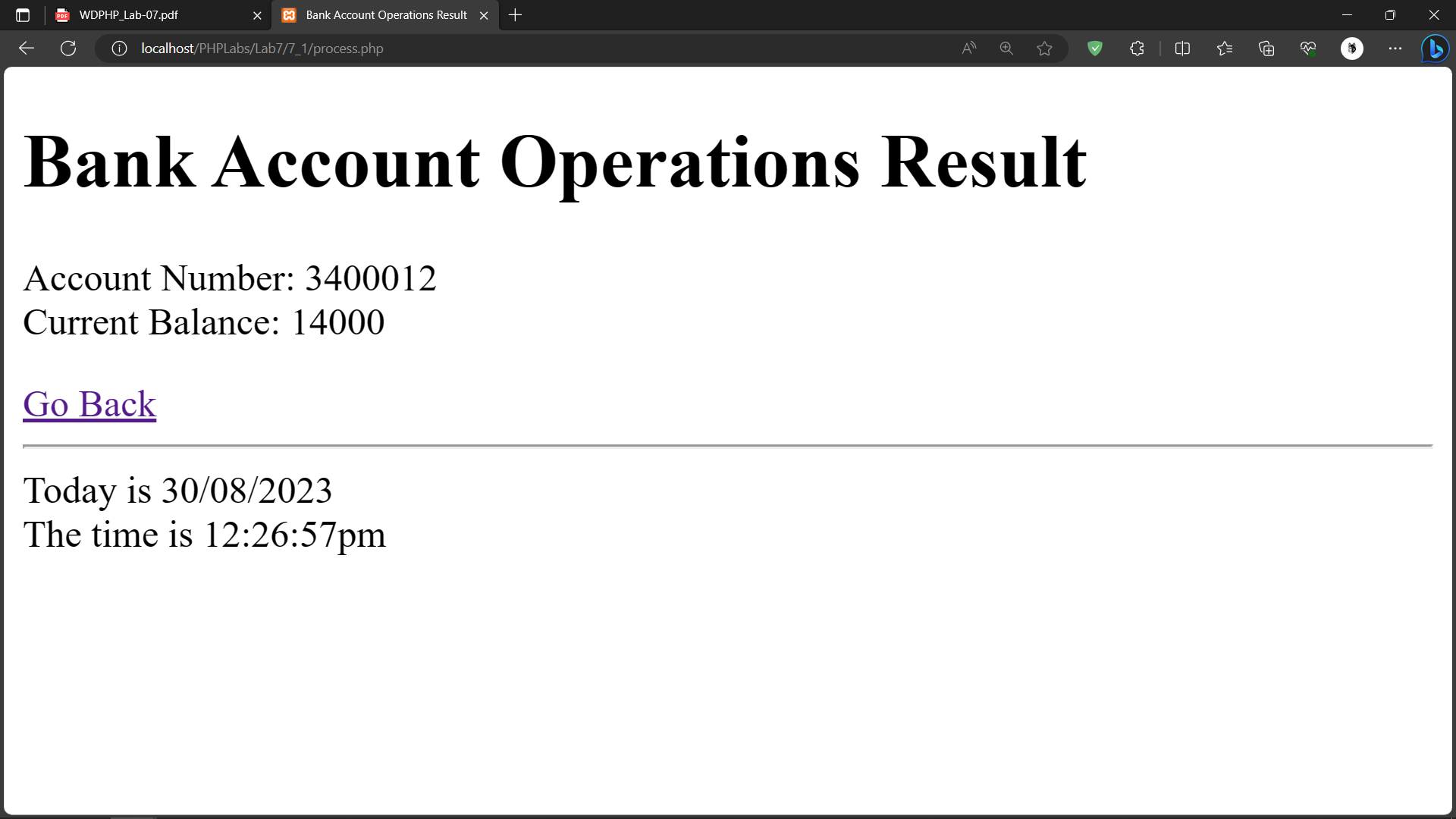
|  |
| --- |
| <?php  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  $accountNumber = $\_POST["account\_number"];  $currentBalance = $\_POST["current\_balance"];  $amount = $\_POST["amount"];    include 'Bank.php';  $bank = new Bank($accountNumber, $currentBalance);  if (isset($\_POST["deposit"])) {  $bank->deposit($amount);  } elseif (isset($\_POST["withdraw"])) {  $bank->withdraw($amount);  }  $accountInfo = $bank->getAccountInfo();  }  ?>  <!DOCTYPE html>  <html>  <head>  <title>Bank Account Operations Result</title>  </head>  <body>  <h1>Bank Account Operations Result</h1>  <?php  if (isset($accountInfo)) {  echo "<p>{$accountInfo}</p>";  }  ?>  <a href="index.html">Go Back</a>  <?php  echo "<hr>Today is " . date("d/m/Y") . "<br>";  echo "The time is " . date("h:i:sa");  ?>  </body>  </html> |

**bank.php**

|  |
| --- |
| <?php  class Bank {  private $accountNumber;  private $currentBalance;  public function \_\_construct($accountNumber, $currentBalance) {  $this->accountNumber = $accountNumber;  $this->currentBalance = $currentBalance;  }  public function deposit($amount) {  if ($amount > 0) {  $this->currentBalance += $amount;  }  }  public function withdraw($amount) {  if ($amount > 0 && $this->currentBalance >= $amount) {  $this->currentBalance -= $amount;  }  }  public function getAccountInfo() {  return "Account Number: {$this->accountNumber}<br>Current Balance: {$this->currentBalance}";  }  }  ?> |

**Output:**





**2. Create a class Food. Having a property name, category and price. Create a**

**constructor which assigns the values to the properties. Class should have a method**

**order() which takes the quantity and calculates the total price. Display() will**

**display all food details as well as total price of a food item. getTotalPrice() return**

**the total price of each food item (price\*qty) (\*qty and total price are not class**

**variables). Give all necessary input by the HTML form. And display output in**

**HTML form. Take the necessary input of three food items from HTML form. [ 3**

**objects of a class should be created] . Create a php function which calculates the**

**final sum of all the object’s total price and display.**

**Code:**

**Index.html**

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <title>Food Order Form</title>  </head>  <body>  <h1>Food Order Form</h1>  <form method="post" action="process.php">  <h2>Food Item 1</h2>  <label for="name1">Name:</label>  <input type="text" name="name1" required><br>  <label for="category1">Category:</label>  <input type="text" name="category1" required><br>  <label for="price1">Price:</label>  <input type="number" name="price1" step="0.01" required><br>  <label for="quantity1">Quantity:</label>  <input type="number" name="quantity1" required><br><br>  <h2>Food Item 2</h2>  <label for="name2">Name:</label>  <input type="text" name="name2" required><br>  <label for="category2">Category:</label>  <input type="text" name="category2" required><br>  <label for="price2">Price:</label>  <input type="number" name="price2" step="0.01" required><br>  <label for="quantity2">Quantity:</label>  <input type="number" name="quantity2" required><br><br>  <h2>Food Item 3</h2>  <label for="name3">Name:</label>  <input type="text" name="name3" required><br>  <label for="category3">Category:</label>  <input type="text" name="category3" required><br>  <label for="price3">Price:</label>  <input type="number" name="price3" step="0.01" required><br>  <label for="quantity3">Quantity:</label>  <input type="number" name="quantity3" required><br><br>  <input type="submit" value="Calculate Total">  </form>  </body>  </html> |

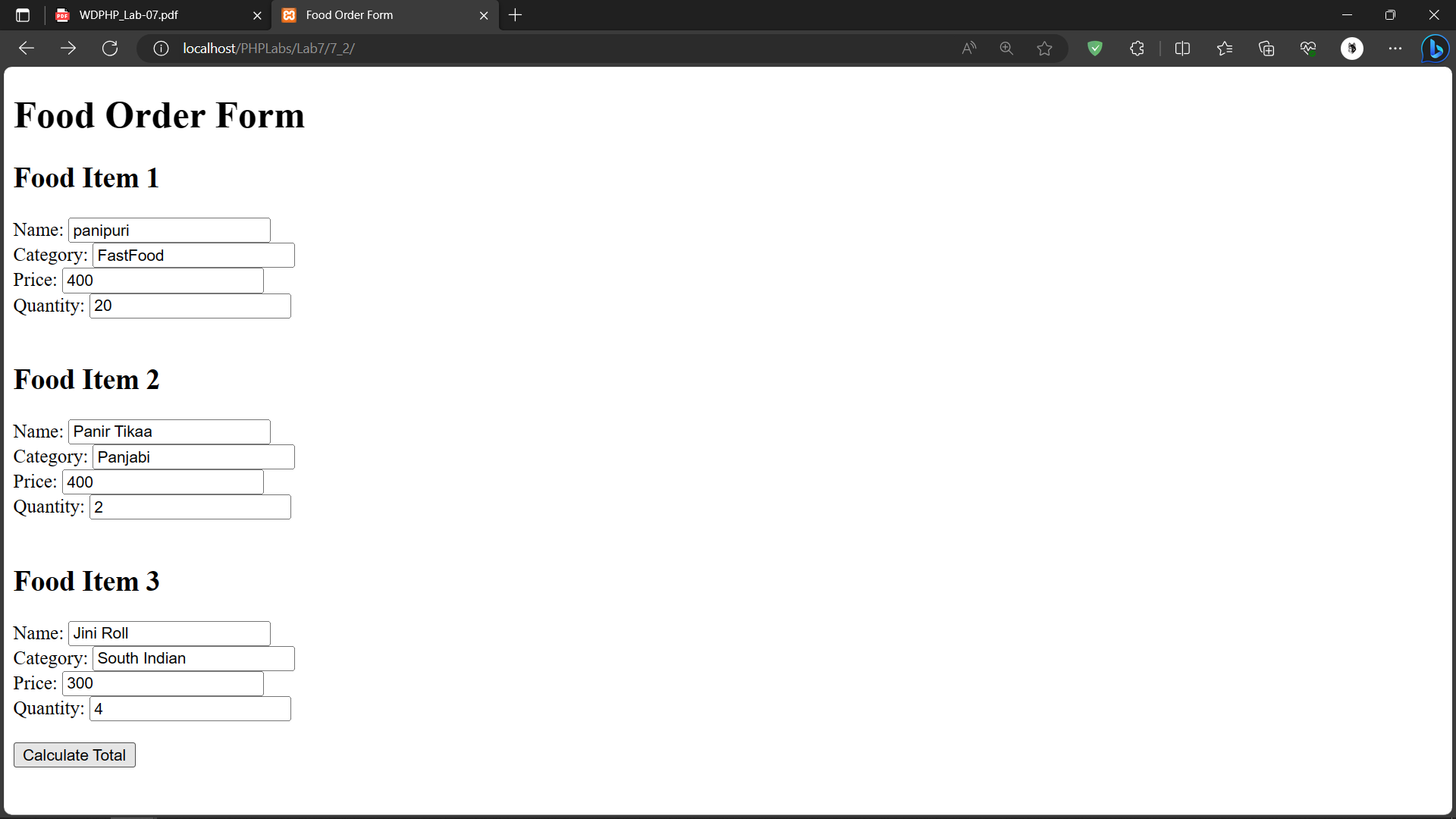
**Food.php**

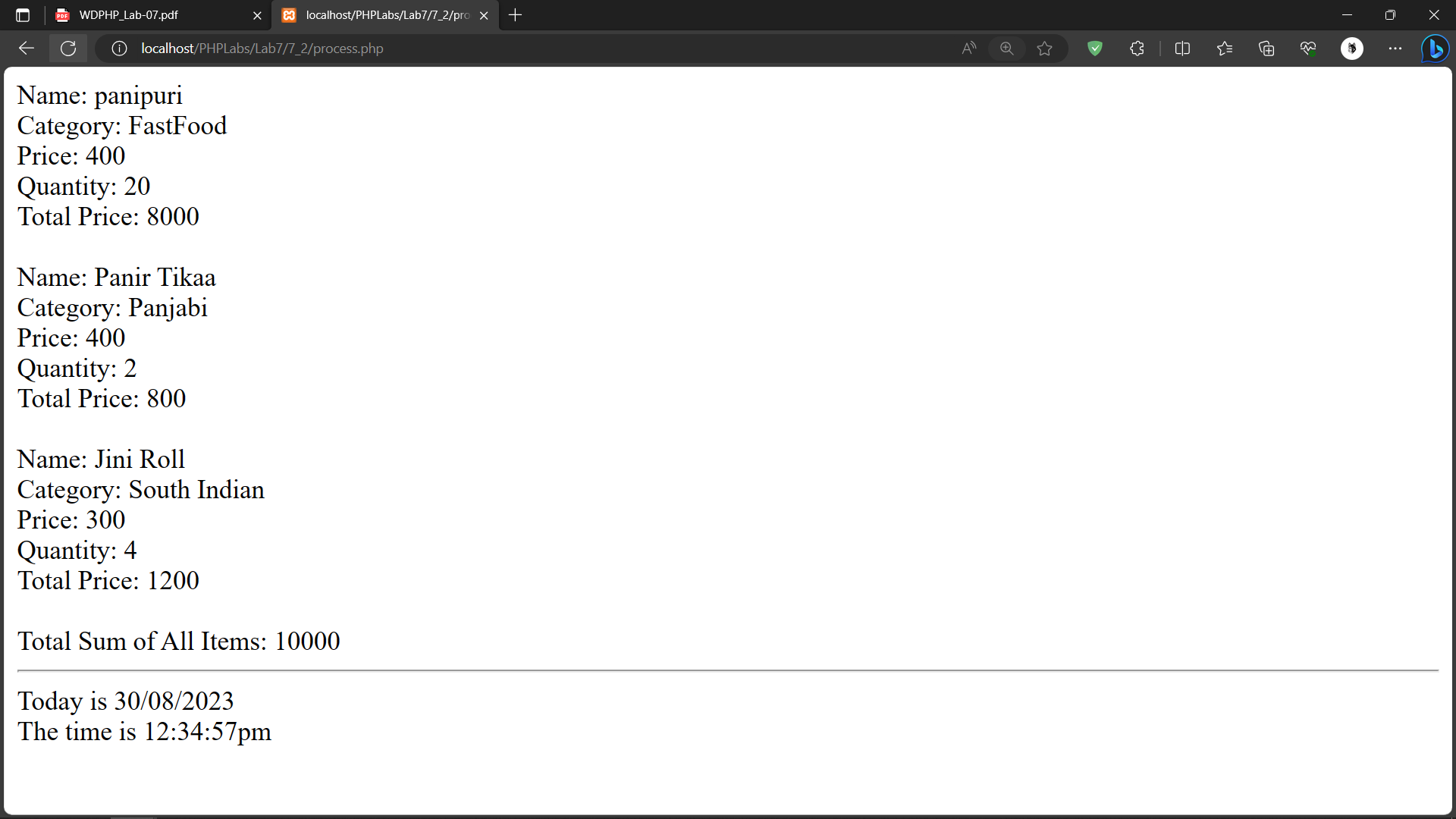
|  |
| --- |
| <?php  class Food {  private $name;  private $category;  private $price;  public function \_\_construct($name, $category, $price) {  $this->name = $name;  $this->category = $category;  $this->price = $price;  }  public function order($quantity) {  return $this->price \* $quantity;  }  public function display() {  echo "Name: {$this->name}<br>";  echo "Category: {$this->category}<br>";  echo "Price: {$this->price}<br>";  }  }  ?> |

**process.php**

|  |
| --- |
| <?php  include 'Food.php';  $foodItems = array();  $totalSum = 0;  for ($i = 1; $i <= 3; $i++) {  $name = $\_POST["name$i"];  $category = $\_POST["category$i"];  $price = $\_POST["price$i"];  $quantity = $\_POST["quantity$i"];  $food = new Food($name, $category, $price);  $totalPrice = $food->order($quantity);  $food->display();  echo "Quantity: $quantity<br>";  echo "Total Price: $totalPrice<br><br>";  $totalSum += $totalPrice;  }  echo "Total Sum of All Items: $totalSum";  echo "<hr>Today is " . date("d/m/Y") . "<br>";  echo "The time is " . date("h:i:sa");  ?> |

**Output:**





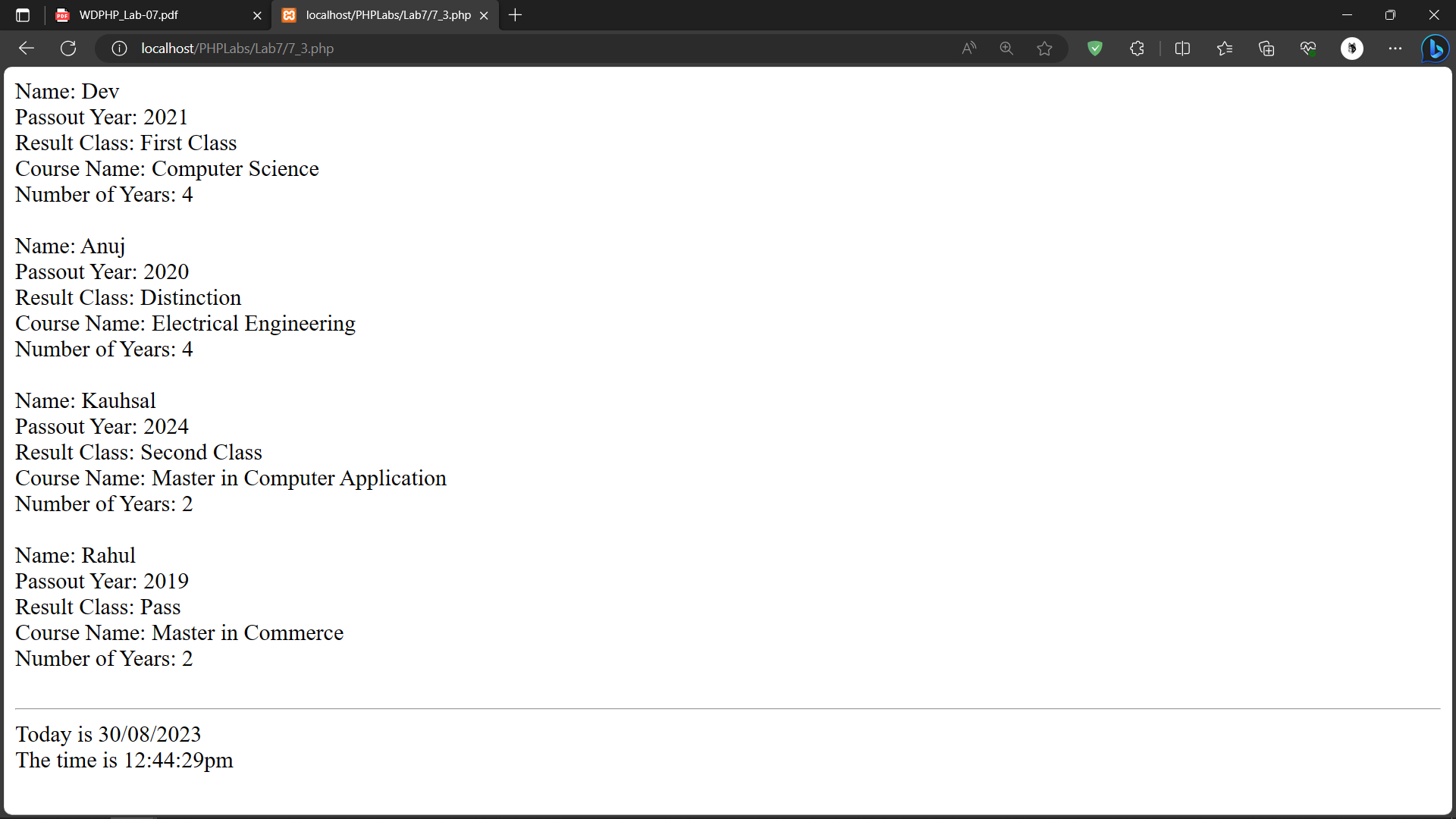
1. **Create a class course having properties coursename,noofyear. Create setter and getter public methods for each property. Create a student class which inherits the course class. Student class should have name, passout year, resultclass properties . Student class contain public method setvalue() which set all the properties value. And display() which display all the values. Create four different object of student class and display details of each object.**

**Code:**

**7\_3.php**

|  |
| --- |
| <?php  class Course {  private $courseName;  private $noOfYears;  public function getCourseName() {  return $this->courseName;  }  public function setCourseName($courseName) {  $this->courseName = $courseName;  }  public function getNoOfYears() {  return $this->noOfYears;  }  public function setNoOfYears($noOfYears) {  $this->noOfYears = $noOfYears;  }  }  class Student extends Course {  private $name;  private $passoutYear;  private $resultClass;  public function setValue($name, $passoutYear, $resultClass) {  $this->name = $name;  $this->passoutYear = $passoutYear;  $this->resultClass = $resultClass;  }  public function display() {  echo "Name: " . $this->name . "<br>";  echo "Passout Year: " . $this->passoutYear . "<br>";  echo "Result Class: " . $this->resultClass . "<br>";  echo "Course Name: " . $this->getCourseName() . "<br>";  echo "Number of Years: " . $this->getNoOfYears() . "<br><br>";  }  }  $student1 = new Student();  $student1->setCourseName("Computer Science");  $student1->setNoOfYears(4);  $student1->setValue("Dev", 2021, "First Class");  $student2 = new Student();  $student2->setCourseName("Electrical Engineering");  $student2->setNoOfYears(4);  $student2->setValue("Anuj", 2020, "Distinction");  $student3 = new Student();  $student3->setCourseName("Master in Computer Application");  $student3->setNoOfYears(2);  $student3->setValue("Kauhsal", 2024, "Second Class");  $student4 = new Student();  $student4->setCourseName("Master in Commerce");  $student4->setNoOfYears(2);  $student4->setValue("Eve", 2019, "Pass");  $student1->display();  $student2->display();  $student3->display();  $student4->display();  echo "<hr>Today is " . date("d/m/Y") . "<br>";  echo "The time is " . date("h:i:sa");  ?> |

**Output:**



**4** .**Create a class vehicle having properties prodyear,company name and a protected**

**method for setting and getting values. Create subclass TwoWheeler inherits from**

**vehicle. Properties are nameofvehicle, color. It contains methods setvalues() and**

**getvalues() which set the values of all properties and display it respectively. Create a**

**subclass FourWheeler inherits from vehicle class. Properties are vehiclename, color,**

**price, tolltaxrate . It contains constructor which set all the properties. Display()**

**method for displaying the details. Create an object of TwoWheeler and FourWheeler**

**and display all the details for both the objects.**

**Code:**

**7\_4.php**

|  |
| --- |
| <?php  class Vehicle {  protected $prodYear;  protected $companyName;  public function setValues($prodYear, $companyName) {  $this->prodYear = $prodYear;  $this->companyName = $companyName;  }  public function getValues() {  return "Production Year: {$this->prodYear}<br>Company Name: {$this->companyName}";  }  }  class TwoWheeler extends Vehicle {  private $nameOfVehicle;  private $color;  public function setValues($prodYear, $companyName) {  parent::setValues($prodYear, $companyName);  }  public function setAdditionalValues($nameOfVehicle, $color) {  $this->nameOfVehicle = $nameOfVehicle;  $this->color = $color;  }  public function getValues() {  $parentValues = parent::getValues();  return "$parentValues<br>Name of Vehicle: {$this->nameOfVehicle}<br>Color: {$this->color}";  }  }  class FourWheeler extends Vehicle {  private $vehicleName;  private $color;  private $price;  private $tollTaxRate;  public function \_\_construct($prodYear, $companyName, $vehicleName, $color, $price, $tollTaxRate) {  parent::setValues($prodYear, $companyName);  $this->vehicleName = $vehicleName;  $this->color = $color;  $this->price = $price;  $this->tollTaxRate = $tollTaxRate;  }  public function display() {  $parentValues = parent::getValues();  echo "$parentValues<br>Vehicle Name: {$this->vehicleName}<br>Color: {$this->color}<br>Price: {$this->price}<br>Toll Tax Rate: {$this->tollTaxRate}";  }  }  // Create an object of TwoWheeler  $twoWheeler = new TwoWheeler();  $twoWheeler->setValues(2020, "Honda");  $twoWheeler->setAdditionalValues("Scooter", "Red");  // Create an object of FourWheeler  $fourWheeler = new FourWheeler(2019, "Toyota", "Sedan", "Blue", 25000, 5.0);  // Display details for both objects  echo "<h2>Two-Wheeler Details</h2>";  echo $twoWheeler->getValues();  echo "<h2>Four-Wheeler Details</h2>";  $fourWheeler->display();  echo "<hr>Today is " . date("d/m/Y") . "<br>";  echo "The time is " . date("h:i:sa");  ?> |

**Output:**

