

## Лабораторна робота №4

### Тема: Поведінкові шаблони

**Мета роботи:** навчитися реалізовувати структурні шаблони проєктування  
Ланцюжок відповідальностей, Посередник, Спостерігач, Стратегія, Мemento

### Хід роботи:

#### Завдання 0: Підготовка до виконання завдання

1. Створити директорию **lab-4** в Вашому репозиторії з лабораторками.
2. Перейти до Завдання 1 😊

#### Завдання 1: Ланцюжок відповідальностей.

1. Створіть систему підтримки користувачів.
2. Вона має функціонувати так, як покрокове меню (наприклад, коли телефонуєте до оператора мобільного зв'язку). Але замість голосових повідомлень Ви маєте виводити відповідні повідомлення в консоль і чекати на вибір користувача.
3. Система повинна мати мінімум 4 рівні. Всі питання мають на меті обрати правильний рівень підтримки (тобто Handler) для користувача. Тобто вже на першому питанні може бути підібрано правильний рівень підтримки, тоді меню закінчується. А може бути так, що на жодному питанні не буде знайдено правильний рівень (Handler), тоді меню має повторитися.
4. Покажіть правильність роботи свого коду запустивши його в головному методі програми.

### Лістинг програми:

#### Handler.php:

```
<?php

abstract class Handler
{
    protected $nextHandler;

    public function setNext(Handler $handler)
    {
        $this->nextHandler = $handler;
        return $handler;
    }
}
```

					ДУ «Житомирська політехніка».22.121.16.000 – Лр4			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Кохан Т.О			Звіт з лабораторної роботи №4	Літ.	Арк.	Аркушів
Перевір.		Фант М. О.					1	17
Керівник						ФІКТ Гр. ІПЗ-22-3[1]		
Н. контр.								
Зав. каф.		Морозов А.В.						

```
public function handle($request)
{
    $response = $this->process($request);
    if ($response !== null) {
        return $response;
    } elseif ($this->nextHandler) {
        return $this->nextHandler->handle($request);
    }

    return "Ваш запит найближчим часом буде оброблений.";
}

abstract protected function process($request);
}
```

ConcreteHandlers.php:

```
<?php

require_once 'Handler.php';

class GeneralInquiryHandler extends Handler
{
    protected function process($request)
    {
        if ($request === '1') {
            return "Загальні питання: Ваша проблема була вирішена.";
        }
        return null;
    }
}

class BillingHandler extends Handler
{
    protected function process($request)
    {
        if ($request === '2') {
            return "Питання щодо рахунків: Ваша проблема була вирішена.";
        }
        return null;
    }
}

class TechnicalSupportHandler extends Handler
{
    protected function process($request)
    {
        if ($request === '3') {
            return "Технічна підтримка: Ваша проблема була вирішена.";
        }
        return null;
    }
}

class OtherInquiriesHandler extends Handler
{
    protected function process($request)
    {
        if ($request === '4') {
            return "Інші питання: Ваша проблема була вирішена.";
        }
        return null;
    }
}
```

```
}
```

**SupportSystem.php:**

```
<?php

require_once 'ConcreteHandlers.php';

class SupportSystem
{
    private $handler;
    private $questions;

    public function __construct()
    {
        $this->handler = new GeneralInquiryHandler();
        $this->handler->setNext(new BillingHandler())
            ->setNext(new TechnicalSupportHandler())
            ->setNext(new OtherInquiriesHandler());

        $this->questions = [
            'start' => [
                'question' => 'Ласкаво просимо до служби підтримки. Оберіть тип
вашого питання:',
                'options' => [
                    '1' => 'Загальні питання',
                    '2' => 'Питання щодо рахунків',
                    '3' => 'Технічна підтримка',
                    '4' => 'Інші питання'
                ]
            ],
            '1' => [
                'question' => 'Загальні питання. Оберіть одну з наступних
опцій:',
                'options' => [
                    '1.1' => 'Деталі тарифів',
                    '1.2' => 'Інші загальні питання'
                ]
            ],
            '2' => [
                'question' => 'Питання щодо рахунків. Оберіть одну з наступних
опцій:',
                'options' => [
                    '2.1' => 'Перевірка балансу',
                    '2.2' => 'Оплата рахунку'
                ]
            ],
            '3' => [
                'question' => 'Технічна підтримка. Оберіть одну з наступних
опцій:',
                'options' => [
                    '3.1' => 'Проблеми з підключенням',
                    '3.2' => 'Проблеми з пристроєм'
                ]
            ],
            '4' => [
                'question' => 'Інші питання. Оберіть одну з наступних опцій:',
                'options' => [
                    '4.1' => 'Скарги',
                    '4.2' => 'Пропозиції'
                ]
            ]
        ];
    }
}
```

```
public function getNextQuestion($level)
{
    return $this->questions[$level] ?? null;
}

public function getResponse($level)
{
    return $this->handler->handle($level);
}
}
```

index.php:

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0,
maximum-scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Main</title>
</head>
<body>
<h1>Система підтримки користувачів</h1>
<h2>Вітаємо, чим можемо допомогти?</h2>
<?php
require_once 'SupportSystem.php';

session_start();

if (!isset($_SESSION['support_system'])) {
    $_SESSION['support_system'] = new SupportSystem();
    $_SESSION['current_level'] = 'start';
}

$supportSystem = $_SESSION['support_system'];
$currentLevel = $_SESSION['current_level'];
$response = '';
$showPhoneForm = false;

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    if (isset($_POST['choice'])) {
        $choice = $_POST['choice'];
        $nextQuestion = $supportSystem->getNextQuestion($choice);

        if ($nextQuestion !== null) {
            $_SESSION['current_level'] = $choice;
            $currentLevel = $choice;
        } else {
            $response = $supportSystem->getResponse($choice);
            $_SESSION['current_level'] = 'start';
            $showPhoneForm = true;
        }
    } elseif (isset($_POST['phone'])) {
        $phone = $_POST['phone'];
        echo "<p>Дякуємо! Ми зателефонуємо вам на номер $phone.</p>";
        $_SESSION['current_level'] = 'start';
    }
}

$question = $supportSystem->getNextQuestion($currentLevel);
?>

<?php if ($response && $showPhoneForm) : ?>
    <p><?php echo $response; ?></p>
</p>
```

```
<form method="post">
    <p>Будь ласка, залиште свій номер телефону для зворотного дзвінка:</p>
    <input type="tel" name="phone" required>
    <button type="submit">Відправити</button>
</form>
<?php elseif ($question) : ?>
    <form method="post">
        <p><?php echo $question['question']; ?></p>
        <?php foreach ($question['options'] as $key => $value) : ?>
            <div>
                <input type="radio" id="choice<?php echo $key; ?>" name="choice"
value="<?php echo $key; ?>" required>
                <label for="choice<?php echo $key; ?>"><?php echo $value;
?></label>
            </div>
        <?php endforeach; ?>
        <button type="submit">Відправити</button>
    </form>
<?php endif; ?>
</body>
</html>
```

Результати виконання програми:

Система підтримки користувачів

Вітаємо, чим можемо допомогти?

Ласкаво просимо до служби підтримки. Оберіть тип вашого питання:

- ☐ Загальні питання
  - ☐ Питання щодо рахунків
  - ☐ Технічна підтримка
  - ☐ Інші питання
- 

Система підтримки користувачів

Вітаємо, чим можемо допомогти?

Ваш запит найближчим часом буде оброблений.

Будь ласка, залиште свій номер телефону для зворотного дзвінка:

Система підтримки користувачів

Вітаємо, чим можемо допомогти?

Загальні питання. Оберіть одну з наступних опцій:

- ☐ Деталі тарифів
  - ☐ Інші загальні питання
- 

Дякуємо! Ми зателефонуємо вам на номер 0681234567.

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

## Завдання 2: Посередник.

1. Відрефакторте [код](#) продемонстрований на лекції за допомогою використання шаблону Посередник.
2. В результаті рефакторингу Aircraft не повинен “знати” про Runway і навпаки. Обидві сутності повинні “знати” лише про CommandCentre.
3. Покажіть правильність роботи свого коду запустивши його в головному методі програми.

### Лістинг програми:

#### index.php:

```
<?php

interface Mediator {
    public function notify($sender, $event);
}

class CommandCentre implements Mediator {
    private $runways = [];
    private $aircrafts = [];

    public function addRunway(Runway $runway) {
        $this->runways[] = $runway;
    }

    public function addAircraft(Aircraft $aircraft) {
        $this->aircrafts[] = $aircraft;
    }

    public function notify($sender, $event) {
        if ($event == 'landing') {
            foreach ($this->runways as $runway) {
                if (!$runway->isBusy()) {
                    $runway->land($sender);
                    return;
                }
            }
            echo "Could not land, all runways are busy.\n";
        } elseif ($event == 'takeoff') {
            foreach ($this->runways as $runway) {
                if ($runway->isBusy() && $runway->getAircraft() === $sender) {
                    $runway->takeOff();
                    return;
                }
            }
            echo "Could not take off, the runway is not assigned to this
aircraft.\n";
        }
    }
}

class Aircraft {
    private $name;
    private $mediator;

    public function construct($name, Mediator $mediator) {
```

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        $this->name = $name;
        $this->mediator = $mediator;
    }

    public function getName() {
        return $this->name;
    }

    public function land() {
        echo "<p>Aircraft {$this->getName()} is landing.</p>\n";
        $this->mediator->notify($this, 'landing');
    }

    public function takeOff() {
        echo "<p>Aircraft {$this->getName()} is taking off.</p>\n";
        $this->mediator->notify($this, 'takeoff');
    }
}

class Runway {
    private $id;
    private $mediator;
    private $busy = false;
    private $aircraft = null;

    public function __construct(Mediator $mediator) {
        $this->id = uniqid();
        $this->mediator = $mediator;
    }

    public function land(Aircraft $aircraft) {
        echo "<p>Runway {$this->id}: Aircraft {$aircraft->getName()} has
landed.</p>\n";
        $this->busy = true;
        $this->aircraft = $aircraft;
    }

    public function takeOff() {
        echo "<p>Runway {$this->id}: Aircraft {$this->aircraft->getName()} has
taken off.</p>\n";
        $this->busy = false;
        $this->aircraft = null;
    }

    public function isBusy() {
        return $this->busy;
    }

    public function getAircraft() {
        return $this->aircraft;
    }
}

$mediator = new CommandCentre();

$runway1 = new Runway($mediator);
$runway2 = new Runway($mediator);

$mediator->addRunway($runway1);
$mediator->addRunway($runway2);

$aircraft1 = new Aircraft("Boeing 747", $mediator);
$aircraft2 = new Aircraft("Airbus A320", $mediator);

$aircraft1->land();

```

```
$aircraft2->land();  
$aircraft1->takeOff();
```

## Результат виконання програми:

Aircraft Boeing 747 is landing.

Runway 665706b204076: Aircraft Boeing 747 has landed.

Aircraft Airbus A320 is landing.

Runway 665706b20407f: Aircraft Airbus A320 has landed.

Aircraft Boeing 747 is taking off.

Runway 665706b204076: Aircraft Boeing 747 has taken off.

### Завдання 3: Спостерігач.

1. Додайте до Вашого LightHTML з завдання 5 ЛР №3 можливість додавання EventListener до Ваших HTML елементів.
2. Додайте можливість підписки на різні івенти ("click", "mouseover" тощо).
3. Покажіть правильність роботи свого коду запустивши його в головному методі програми.

## Лістинг програми:

### LightNode.php:

```
<?php  
abstract class LightNode  
{  
    abstract public function getOuterHTML();  
    abstract public function getInnerHTML();  
}  
  
class LightTextNode extends LightNode  
{  
    private $text;  
  
    public function __construct($text)  
    {  
        $this->text = $text;  
    }  
  
    public function getOuterHTML()  
    {  
        return $this->text;  
    }  
  
    public function getInnerHTML()  
    {
```

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				8
Змн.	Арк.	№ докум.	Підпис	Дата		



```

        return $this->text;
    }
}

class LightElementNode extends LightNode
{
    private $tag;
    private $displayType;
    private $closingType;
    private $cssClasses = array();
    private $childNodes = array();
    private $eventManager;

    public function __construct($tag, $displayType, $closingType, $cssClasses =
array(), $childNodes = array())
    {
        $this->tag = $tag;
        $this->displayType = $displayType;
        $this->closingType = $closingType;
        $this->cssClasses = $cssClasses;
        $this->childNodes = $childNodes;
        $this->eventManager = new EventManager();
    }

    public function addChild($childNode)
    {
        $this->childNodes[] = $childNode;
    }

    public function addEventListener($event, $handler)
    {
        $this->eventManager->subscribe($event, $handler);
    }

    public function triggerEvent($event, $data = null)
    {
        $this->eventManager->notify($event, $data);
    }

    public function getOuterHTML()
    {
        $html = "<{$this->tag}";

        if (!empty($this->cssClasses)) {
            $html .= ' class="' . implode(' ', $this->cssClasses) . '"';
        }

        // Для кліків та наведень додамо атрибути
        if ($this->eventManager->listeners) {
            $html .= " onclick=\"handleClick()\"";
            $html .= " onmouseover=\"handleMouseOver()\"";
        }

        if ($this->closingType === 'single') {
            $html .= '>';
        } else {
            $html .= '>';
            foreach ($this->childNodes as $child) {
                $html .= $child->getOuterHTML();
            }
            $html .= "</{$this->tag}>";
        }

        return $html;
    }
}

```

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

```
public function getInnerHTML()
{
    $html = '';
    foreach ($this->childNodes as $child) {
        $html .= $child->getOuterHTML();
    }
    return $html;
}
}
```

**EventManager.php:**

```
<?php
class EventManager
{
    public $listeners = array();

    public function subscribe($eventType, $callback)
    {
        if (!isset($this->listeners[$eventType])) {
            $this->listeners[$eventType] = array();
        }
        $this->listeners[$eventType][] = $callback;
    }

    public function notify($eventType, $data = null)
    {
        if (isset($this->listeners[$eventType])) {
            foreach ($this->listeners[$eventType] as $listener) {
                call_user_func($listener, $data);
            }
        }
    }
}
```

**index.php:**

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0,
maximum-scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Main</title>
    <style>
        #message {
            margin-top: 20px;
            padding: 10px;
            border: 1px solid darkgray;
            display: none;
        }
    </style>
    <script>
        function handleClick() {
            displayMessage('Element clicked!');
        }

        function handleMouseOver() {
            displayMessage('Mouse over the element!');
        }

        function displayMessage(message) {
```

```

        const messageDiv = document.getElementById('message');
        messageDiv.innerText = message;
        messageDiv.style.display = 'block';
    }
</script>
</head>
<body>
<?php
require_once 'LightNode.php';
require_once 'EventManager.php';

$childText = new LightTextNode('Click me!');
$element = new LightElementNode('button', 'block', 'double', ['btn', 'btn-
primary'], [$childText]);

$element->addEventListener('click', function() {
    echo "<script>handleClick();</script>";
});

$element->addEventListener('mouseover', function() {
    echo "<script>handleMouseOver();</script>";
});

echo $element->getOuterHTML();
?>
<div id="message"></div>
<div id="message"></div>
</body>
</html>

```

### Результат виконання програми:

Click me!

Click me!

Mouse over the element!

Element clicked!

### Завдання 4: Стратегія.

1. Додайте новий елемент Image, який за допомогою стратегії в залежності від переданого href буде завантажувати картинку або з файлової системи або з мережі.
2. Покажіть правильність роботи свого коду запустивши його в головному методі програми.

### Лістинг програми:

#### LightNode.php:

```

<?php

abstract class LightNode
{
    abstract public function getOuterHTML();
}

```

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        abstract public function getInnerHTML();
    }

class LightTextNode extends LightNode
{
    private $text;

    public function __construct($text)
    {
        $this->text = $text;
    }

    public function getOuterHTML()
    {
        return $this->text;
    }

    public function getInnerHTML()
    {
        return $this->text;
    }
}

class LightImageNode extends LightNode
{
    private $src;
    private $strategy;
    private $altText;

    public function __construct($href, ImageLoadingStrategy $strategy, $altText
= "")
    {
        $this->strategy = $strategy;
        $this->src = $this->strategy->getImageSrc($href);
        $this->altText = $altText;
    }

    public function getOuterHTML()
    {
        return 'altText . '"/>';
    }

    public function getInnerHTML()
    {
        return '';
    }
}

```

### ImageLoadingStrategy.php:

```

<?php
interface ImageLoadingStrategy
{
    public function getImageSrc($href);
}

class FileImageLoadingStrategy implements ImageLoadingStrategy
{
    public function getImageSrc($href)
    {
        return $href;
    }
}

class NetworkImageLoadingStrategy implements ImageLoadingStrategy
{

```

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

```

public function getImageSrc($href)
{
    return $href;
}

```

## index.php:

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0,
maximum-scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Main</title>
</head>
<body>
<?php
require_once 'LightNode.php';
require_once 'ImageLoadingStrategy.php';

$fileStrategy = new FileImageLoadingStrategy();
$networkStrategy = new NetworkImageLoadingStrategy();

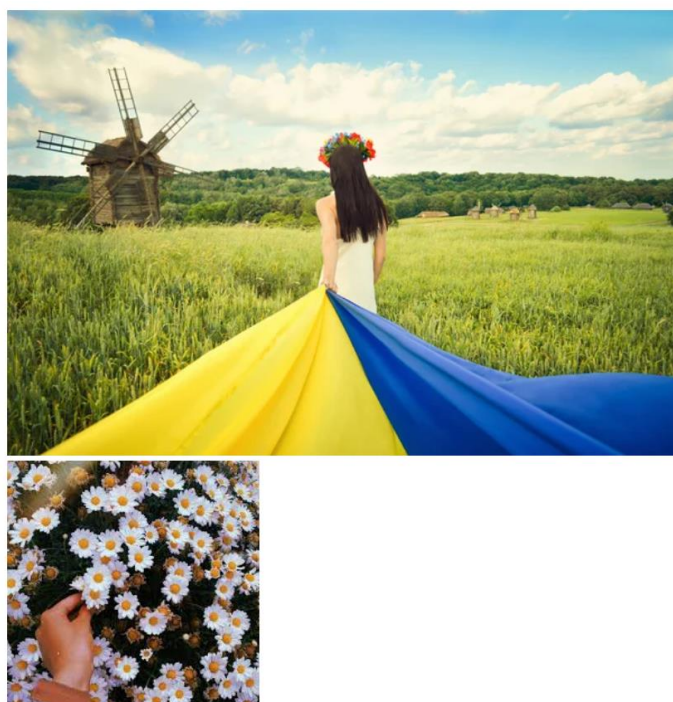
$fileImage = new LightImageNode('example.jpg', $fileStrategy, 'Local Image');

$networkImage = new LightImageNode('https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRGrNZsK5Gdy5sDxadEg-K-xL7sOmIsEHI9eA&s',
$networkStrategy, 'Network Image');

echo $fileImage->getOuterHTML();
echo '<br>';
echo $networkImage->getOuterHTML();
?>
</body>
</html>

```

## Результат виконання програми:



		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				13
Змн.	Арк.	№ докум.	Підпис	Дата		

### Завдання 5: Мементо.

1. Уявіть, що ви створюєте програму текстового редактора.
2. Для цього завдання буде достатньо РОС-версії: клас `TextEditor`, який містить в собі поточну версію текстового документу, який представлений спеціальним класом `TextDocument`. Назви класам можете давати власні.
3. Реалізуйте функцію збереження і скасування, яка дозволить користувачам відмінити зміни, зроблені в документі. Завдання полягає в тому, щоб зберегти стан документа в різні моменти часу та відновити його, коли це необхідно, не розкриваючи внутрішню реалізацію документа чи текстового редактору.

### Лістинг програми:

#### TextEditor.php:

```
<?php
namespace Task5;

class TextEditor
{
    private $document;
    private $caretaker;

    public function __construct()
    {
        $this->document = new TextDocument();
        $this->caretaker = new Caretaker();
    }

    public function setText($content)
    {
        $this->caretaker->saveMemento($this->document->createMemento());
        $this->document->setContent($content);
    }

    public function undo()
    {
        $memento = $this->caretaker->getLastMemento();
        if ($memento !== null) {
            $this->document->restoreFromMemento($memento);
        }
    }

    public function getText()
    {
        return $this->document->getContent();
    }
}
```

#### TextDocument.php:

```
<?php
namespace Task5;
```

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				14
Змн.	Арк.	№ докум.	Підпис	Дата		

```

class TextDocument
{
    private $content;

    public function __construct($content = "")
    {
        $this->content = $content;
    }

    public function setContent($content)
    {
        $this->content = $content;
    }

    public function getContent()
    {
        return $this->content;
    }

    public function createMemento()
    {
        return new Memento($this->content);
    }

    public function restoreFromMemento(Memento $memento)
    {
        $this->content = $memento->getContent();
    }
}

```

### Caretaker.php:

```

<?php
namespace Task5;

class Caretaker
{
    private $mementos = [];

    public function saveMemento(Memento $memento)
    {
        $this->mementos[] = $memento;
    }

    public function getLastMemento()
    {
        if (count($this->mementos) > 0) {
            return array_pop($this->mementos);
        }
        return null;
    }
}

```

### Memento.php:

```

<?php
namespace Task5;

```

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				15
Змн.	Арк.	№ докум.	Підпис	Дата		

```

class Memento
{
    private $content;

    public function __construct($content)
    {
        $this->content = $content;
    }

    public function getContent()
    {
        return $this->content;
    }
}

```

## index.php:

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0,
maximum-scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Editor</title>
</head>
<body>
<h1>Text Editor</h1>
<?php
use Task5\TextEditor;
require_once 'TextDocument.php';
require_once 'Memento.php';
require_once 'TextEditor.php';
require_once 'Caretaker.php';

session_start();

if (!isset($_SESSION['editor'])) {
    $_SESSION['editor'] = new TextEditor();
}

$editor = $_SESSION['editor'];

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    if (isset($_POST['save'])) {
        $editor->setText($_POST['text']);
    } elseif (isset($_POST['undo'])) {
        $editor->undo();
    }
}
?>

<form method="post">
    <textarea name="text" rows="10" cols="50"><?php echo
htmlspecialchars($editor->getText()); ?></textarea><br>
    <button type="submit" name="save">Save</button>
    <button type="submit" name="undo">Delete</button>
</form>
</body>
</html>

```

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				16
Змн.	Арк.	№ докум.	Підпис	Дата		



## Результат виконання програми:

### Text Editor

Виконуємо лабораторну роботу №4

Save

Delete

### Text Editor

Виконуємо лабораторну роботу

Save

Delete

**Висновок:** під час виконання лабораторної роботи я навчилася реалізовувати структурні шаблони проєктування Ланцюжок відповідальностей, Посередник, Спостерігач, Стратегія, Мemento.

		Кохан Т.О.			ДУ «Житомирська політехніка».22.121.16.000 – Лр4	Арк.
		Фант М. О.				17
Змн.	Арк.	№ докум.	Підпис	Дата		