Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Национальный исследовательский университет ИТМО»

(Университет ИТМО)

Мегафакультет компьютерных технологий и управления  
(МФКТиУ)

Факультет программной инженерии и компьютерной техники  
(ФПИиКт)

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №4

«Проектирование системы администрирования для сайта»

Студент:  
Мухина Е. М.  
гр. P3369

Преподаватель:

Климов И. В.

Санкт-Петербург 2024

**СОДЕРЖАНИЕ**

[Задание 3](#_Toc188499049)

[Описание проектирования системы администрирования 4](#_Toc188499050)

[Код сайта 5](#_Toc188499051)

# Задание

Разработайте систему администрирования для вашего сайта.

# Описание проектирования системы администрирования

Для сайта в работе про проектирование базы данных были обозначены 4 категории, которые использовались для создания базы данных: сообщения, заказы, портфолио, отзывы. Для создания системы администрирования также возьмем их.

Администрирование сообщений, оставленных через формы на сайте должно позволять читать их, сортировать, отмечать прочитанным и удалять.

Администрирования заказов должно позволять также видеть их, сортировать их и менять их статус.

Администрирование портфолио должно позволять не только загружать элементы, но и удалять их.

Администрирование отзывов схоже с администрированием сообщений и заказов.

# Код сайта

Options -Indexes

RewriteEngine On

# Устанавливаем index.php как файл по умолчанию

DirectoryIndex index.php index.html

# Исключить статические файлы из обработки перезаписи

RewriteCond %{REQUEST\_FILENAME} !-f

RewriteCond %{REQUEST\_FILENAME} !-d

RewriteCond %{REQUEST\_URI} !\.(css|js|jpg|jpeg|png|gif|ico)$ [NC]

# Перенаправление запросов к страницам без .php (для красивых URL)

RewriteRule ^([a-zA-Z0-9\_-]+)$ index.php?page=$1 [L,QSA]

# Перенаправление старых URL с page= к красивым URL, если нет красивого URL

RewriteCond %{QUERY\_STRING} ^page=([a-zA-Z0-9\_-]+)$

RewriteCond %{REQUEST\_URI} !^/index\.php$ [NC]

RewriteRule ^index\.php$ /%1? [L,R=301]

# Перенаправление запросов к admin.php

RewriteRule ^admin$ admin.php [L,QSA]

php\_value upload\_max\_filesize 200M

php\_value post\_max\_size 200M

1. .htaccess

<main>

<div class="container\_404">

<h1>404 Не найдено</h1>

<h2>Упс! Страница не найдена.</h2>

<p>К сожалению, запрашиваемая вами страница не существует. Она могла быть удалена или вы указали неправильный адрес.

</p>

<p>Попробуйте перейти <a class="back" href="?page=home">на главную страницу</a> или воспользуйтесь меню навигации.</p>

<button onclick="window.history.back();" class="full-width-button">Назад</button>

</main>

1. 404.php

<main>

<h2 class="center">Обо мне</h2>

<div class="about\_left">

<img src="img/prof\_pic.png" class="profile\_pic">

</div>

<div class="about\_right">

<p class="top">Привет! Я Джон Доу — фотограф, чьи кадры захватывают не просто моменты, но и души людей, их чувства и эмоции. Мой путь в фотографии начался с увлечения деталями, которые часто остаются незамеченными. За годы работы я научился находить красоту в самых разных формах — будь то шумные улицы мегаполисов или тихие уголки природы.</p>

<p>Моя работа — это не просто фиксирование событий на пленке. Это попытка рассказать историю, показать мир через призму моего восприятия. Я верю, что каждый кадр — это маленькая часть большого повествования, которое мы создаем сами, шаг за шагом. Работаю в самых разных жанрах фотографии, включая портреты, уличную и документальную фотографию, а также фокусируюсь на природной эстетике. Мои путешествия по Японии и Южной Корее стали важным этапом, когда я начал замечать уникальные детали в культуре, архитектуре и жизни людей этих стран.</p>

<p>Фотография для меня — это способ поймать момент, когда кажется, что мир стоит на грани чего-то особенного. Будь то спокойная тень дерева в городе или буря эмоций на улице — я всегда ищу такие моменты, которые могут рассказать больше, чем тысяча слов.</p>

<p>Каждый проект, который я начинаю, для меня — это не просто работа, а путешествие, в котором важно увидеть, понять и передать историю того, что я снимаю. Я стремлюсь, чтобы каждая фотография оставалась живой, чтобы она могла рассказать свою историю через много лет.</p>

<p>Будь то работа для крупных брендов, съемка в путешествиях или просто портрет друга, я всегда стараюсь передать душу и атмосферу момента. Я горжусь тем, что могу смотреть на мир через объектив, находя удивительное в привычном.</p>

</div>

<div class="main\_about">

<h2>Опыт</h2>

<section class="timeline">

<ul>

<li>

<div class="timeline-content">

<h3>2024 — Преподавание</h3>

<p>С 2024 года я начал преподавать фотографию, проводя как групповые занятия, так и индивидуальные мастер-классы. В своей работе я уделяю внимание как техническим аспектам съемки, так и развитию творческого подхода, помогая студентам увидеть мир с новой перспективы и научиться работать с камерой на профессиональном уровне.</p>

</div>

</li>

<li>

<div class="timeline-content">

<h3>2017–2024 — Практика в фотостудиях Южной Кореи</h3>

<p>В период с 2017 по 2020 год я работал в нескольких фотостудиях, специализирующихся на коммерческой фотографии, портретной съемке и фотожурналистике. Это был важный этап в моей карьере, где я смог освоить работу с различными жанрами фотографии и наработать практический опыт, который повлиял на мой стиль.</p>

</div>

</li>

<li>

<div class="timeline-content">

<h3>2019 — Курсы в токио</h3>

<p>В 2019 году я прошел курсы повышения квалификации в фотографии, которые проводились ведущими профессионалами отрасли в Токио. Эти курсы охватывали современные техники съемки, постобработку и редактуру, а также новые подходы к созданию концептуальных фотопроектов.</p>

</div>

</li>

<li>

<div class="timeline-content">

<h3>2018 — Магистратура в области визуальных искусств</h3>

<p>В 2018 году я завершил магистратуру в области визуальных искусств, где особое внимание уделялось современным направлениям в фотографии и киноискусстве. Мой дипломный проект исследовал взаимодействие традиционной культуры и современных урбанистических пейзажей, что стало основой для многих моих работ.</p>

</div>

</li>

<li>

<div class="timeline-content">

<h3>2016 — Бакалавр в Университете Сеула</h3>

<p>В 2016 году я завершил обучение в одном из ведущих университетов Южной Кореи, Университете Сеула, по специальности "Фотография и визуальные искусства". Во время учебы я углубленно изучал теорию и практику фотографии, а также искусство визуального повествования, что позволило мне выстроить уникальный взгляд на мир через объектив.</p>

</div>

</li>

</ul>

</section>

</div>

</main>

1. about.php

<main>

<p>Добро пожаловать в систему администрирования</p>

</main>

1. admin\_start.php

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<!-- Шрифты -->

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100..900;1,100..900&family=Oswald:wght@200..700&display=swap" rel="stylesheet">

<!-- Стили -->

<link rel="stylesheet" href="style/main.css">

<title>Фотограф Джон Доу | Администрирование и модерация</title>

</head>

<body>

<?php

session\_start();

if(!empty($\_GET["action"])&&$\_GET["action"]=="logout"){

session\_destroy();

header("Location: login.php");

exit;

}

if(!empty($\_GET["action"])&&$\_GET["action"]=="login"&&!empty($\_POST["log"])&&!empty($\_POST["pass"])){

if(($\_POST["log"]=="admin"&&$\_POST["pass"]=="test")){

$\_SESSION["user"]="admin";

}

}

if(!empty($\_SESSION["user"])&&$\_SESSION["user"]=="admin"){

} else {

header("Location: login.php");

exit;

}

?>

<?php

$routes = [ 'comments' => 'feedback\_adm.php','content' => 'portfolio\_adm.php', 'messages' => 'messages\_adm.php', 'orders' => 'orders\_adm.php', 'start' => 'admin\_start.php', 'login' => 'login.php'];

$page = isset($\_GET['page']) ? htmlspecialchars($\_GET['page']) : 'start';

?>

<div class="main">

<header>

<menu>

<ul>

<li><a href="?page=orders">Заказы</a></li>

<li><a href="?page=messages">Сообщения</a></li>

<li><a href="?page=comments">Комментарии</a></li>

<li><a href="?page=content">Портфолио</a></li>

</ul>

</menu>

</header>

<?php

if (array\_key\_exists($page, $routes) && file\_exists($routes[$page])) {

include $routes[$page];

} else {

include '404.php';

}

?>

<footer>

<p>Работает? Не трогай!</p>

<p><a href="admin.php?action=logout">Выйти из системы администрирования</a></p>

</footer>

</div>

</body>

</html>

1. Admin.php

<main>

<div class="contacts\_main">

<h2 class="center">Как меня найти</h2>

<ul>

<li>bechance — <a href="https://www.behance.net/" target="\_blank">Джон Доу</a></li>

<li>flicr — <a href="https://www.flickr.com/" target="\_blank">Джон Доу</a></li>

<li>Организация лекций и мастер-классов: <a href="mailto:education@jdoy.com?subject=Образовательное сотрудничество">education@jdoy.com</a></li>

<li>Для рекламы и коммерческого сотрудничества, e-mail: <a href="mailto:marketing@jdoy.com?subject=Коммерческое сотрудничество">marketing@jdoy.com</a></li>

<li>Студия в Сеуле расположена по адресу 03789 75, Синчон-ро, Содемун-гу, Сеул (서울 서대문구 신촌로 75 (창천동). <br /> Она открыта для свободного посещения с четверга по субботу с 10 до 18 часов. Телефон студии: <a href="tel:02-323-9284">02-323-9284</a>.</li>

<li>Также мои альбомы и книги можно найти на <a href="https://www.amazon.com/">Amazon</a>. Для заказа отдельных фотографий в печати пишите на почту рекламы и сотрудничества.</li>

</ul>

</div>

<div class="main\_form">

<?php

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

try {

$pdo = new PDO($dsn, $username, $password, $options);

// Получение данных из формы

$name = htmlspecialchars(strip\_tags($\_POST['name']));

$contact = htmlspecialchars(strip\_tags($\_POST['contact']));

$message = htmlspecialchars(strip\_tags($\_POST['message']));

$status = 'new'; // Устанавливаем статус сообщения как "новое"

// Подготовка SQL-запроса

$stmt = $pdo->prepare("INSERT INTO messages (name, contact, text, status, data)

VALUES (:name, :contact, :message, :status, NOW())");

// Привязка параметров и выполнение запроса

$stmt->bindParam(':name', $name);

$stmt->bindParam(':contact', $contact);

$stmt->bindParam(':message', $message);

$stmt->bindParam(':status', $status);

// Выполнение запроса

if ($stmt->execute()) {

} else {

echo "<p>Ошибка при отправке сообщения.</p>";

}

} catch (PDOException $e) {

echo "Ошибка подключения к базе данных: " . $e->getMessage();

}

}

?>

<form method="POST" id="messageform">

<h2>Связаться</h2>

<ul class="form\_ul">

<li>

<label for="main\_name">Ваше имя</label>

<input type="text" id="main\_name" name="name" placeholder="Как к вам обращаться" required>

</li>

<li>

<label for="main\_contact">Контакт для связи</label>

<input type="text" id="main\_contact" name="contact" placeholder="Телефон/Почта" required>

</li>

<li>

<textarea id="main\_message" name="message" rows="4" cols="50" placeholder="Текст вашего сообщения." required></textarea>

</li>

<li class="center">

<input type="submit" value="Написать">

</li>

</ul>

</form>

<script src="scripts/forms.js"></script>

</div>

</main>

1. Contacts.php

<main>

<div class="feddbackadm">

<h2 class="center">Модерация комментариев</h2>

<?php

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

$pdo = new PDO($dsn, $username, $password, $options);

// Обработка формы добавления комментария

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['name'], $\_POST['contact'], $\_POST['feedback'])) {

$name = htmlspecialchars(strip\_tags($\_POST['name']));

$contact = htmlspecialchars(strip\_tags($\_POST['contact']));

$feedback = htmlspecialchars(strip\_tags($\_POST['feedback']));

$stmt = $pdo->prepare("INSERT INTO feedback (name, contact, data, text) VALUES (:name, :contact, NOW(), :feedback)");

$stmt->execute(['name' => $name, 'contact' => $contact, 'feedback' => $feedback]);

header('Location: ' . $\_SERVER['REQUEST\_URI']);

exit;

}

// Условия фильтрации и сортировки

$order = $\_GET['order'] ?? 'data';

$direction = $\_GET['direction'] ?? 'DESC';

$limit = (int)($\_GET['limit'] ?? 10);

$offset = (int)($\_GET['page\_num'] ?? 0) \* $limit;

$statusFilter = $\_GET['comment\_status'] ?? null;

$allowedStatuses = ['new', 'approve', 'hidden'];

$statusFilter = in\_array($statusFilter, $allowedStatuses) ? $statusFilter : null;

$allowedColumns = ['name', 'data', 'text', 'contact'];

$allowedDirections = ['ASC', 'DESC'];

$order = in\_array($order, $allowedColumns) ? $order : 'data';

$direction = in\_array($direction, $allowedDirections) ? $direction : 'DESC';

$limit = in\_array($limit, [10, 20, 50, 100]) ? $limit : 10;

if ($statusFilter) {

$stmt = $pdo->prepare("SELECT id\_feedback, name, contact, data, text, status

FROM feedback WHERE status = :statusFilter

ORDER BY $order $direction LIMIT :limit OFFSET :offset");

$stmt->bindValue(':statusFilter', $statusFilter, PDO::PARAM\_STR);

} else {

$stmt = $pdo->prepare("SELECT id\_feedback, name, contact, data, text, status

FROM feedback

ORDER BY $order $direction LIMIT :limit OFFSET :offset");

}

$stmt->bindValue(':limit', $limit, PDO::PARAM\_INT);

$stmt->bindValue(':offset', $offset, PDO::PARAM\_INT);

$stmt->execute();

$feedbacks = $stmt->fetchAll(PDO::FETCH\_ASSOC);

$totalPages = ceil($pdo->query("SELECT COUNT(\*) FROM feedback")->fetchColumn() / $limit);

$currentPageNum = (int)($\_GET['page\_num'] ?? 0);

// Действия с комментариями

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['action'], $\_POST['feedback\_id'])) {

$feedbackId = (int)$\_POST['feedback\_id'];

$action = $\_POST['action'];

if ($action === 'approve') {

$pdo->prepare("UPDATE feedback SET status = 'approve' WHERE id\_feedback = :id")

->execute(['id' => $feedbackId]);

} elseif ($action === 'hide') {

$pdo->prepare("UPDATE feedback SET status = 'hidden' WHERE id\_feedback = :id")

->execute(['id' => $feedbackId]);

} elseif ($action === 'delete') {

$pdo->prepare("DELETE FROM feedback WHERE id\_feedback = :id")

->execute(['id' => $feedbackId]);

} elseif ($action === 'edit' && isset($\_POST['new\_text'])) {

$newText = htmlspecialchars(strip\_tags($\_POST['new\_text']));

$pdo->prepare("UPDATE feedback SET text = :text WHERE id\_feedback = :id")

->execute(['id' => $feedbackId, 'text' => $newText]);

}

header('Location: ' . $\_SERVER['REQUEST\_URI']);

exit;

}

?>

<div class="pagination">

<form method="GET">

<label for="comment\_status">Статус</label>

<select name="comment\_status" id="comment\_status">

<option value="new" <?= $statusFilter == 'new' ? 'selected' : '' ?>>Новые</option>

<option value="approve" <?= $statusFilter == 'approve' ? 'selected' : '' ?>>Одобренные</option>

<option value="hidden" <?= $statusFilter == 'hidden' ? 'selected' : '' ?>>Скрытые</option>

</select>

<label for="limit">Отзывы на странице:</label>

<select name="limit" id="limit">

<option value="10" <?= $limit == 10 ? 'selected' : '' ?>>10</option>

<option value="20" <?= $limit == 20 ? 'selected' : '' ?>>20</option>

<option value="50" <?= $limit == 50 ? 'selected' : '' ?>>50</option>

<option value="100" <?= $limit == 100 ? 'selected' : '' ?>>100</option>

</select>

<label for="order">Сортировка по:</label>

<select name="order" id="order">

<option value="data" <?= $order == 'data' ? 'selected' : '' ?>>Дате</option>

<option value="name" <?= $order == 'name' ? 'selected' : '' ?>>Имени</option>

<option value="text" <?= $order == 'text' ? 'selected' : '' ?>>Сообщению</option>

<option value="contact" <?= $order == 'contact' ? 'selected' : '' ?>>Контактам</option>

</select>

<label for="direction">Направление сортировки:</label>

<select name="direction" id="direction">

<option value="ASC" <?= $direction == 'ASC' ? 'selected' : '' ?>>По возрастанию</option>

<option value="DESC" <?= $direction == 'DESC' ? 'selected' : '' ?>>По убыванию</option>

</select>

<input type="hidden" name="page\_num" value="<?= $currentPageNum ?>">

<input type="hidden" name="page" value="comments">

<button type="submit">Применить</button>

</form>

<p>

<?php

for ($i = 0; $i < $totalPages; $i++):

$isCurrentPage = ($i == $currentPageNum);

?>

<a href="?page\_num=<?= $i ?>&limit=<?= $limit ?>&order=<?= $order ?>&direction=<?= $direction ?>&comment\_status=<?= htmlspecialchars($statusFilter) ?>" class="<?= $isCurrentPage ? 'current\_p' : '' ?>">

<?= $i + 1 ?>

</a>

<?php endfor; ?>

</p>

</div>

<?php foreach ($feedbacks as $feedback): ?>

<div class="feedback">

<p class="feedback\_name"><?= htmlspecialchars($feedback['name'] ?? '') ?></p>

<p class="feedback\_contact"><?= htmlspecialchars($feedback['contact'] ?? '') ?></p>

<p class="feedback\_date"><?= htmlspecialchars($feedback['data'] ?? '') ?></p>

<p class="feedback\_text"><?= htmlspecialchars($feedback['text'] ?? '') ?></p>

<p class="feedback\_status">Статус: <?= htmlspecialchars($feedback['status'] ?? '') ?></p>

<form method="POST">

<input type="hidden" name="feedback\_id" value="<?= $feedback['id\_feedback'] ?>">

<button type="submit" name="action" value="approve">Одобрить</button>

<button type="submit" name="action" value="hide">Скрыть</button>

<button type="submit" name="action" value="delete">Удалить</button>

</form>

<form method="POST">

<input type="hidden" name="feedback\_id" value="<?= $feedback['id\_feedback'] ?>">

<textarea name="new\_text"><?= htmlspecialchars($feedback['text']) ?></textarea>

<button type="submit" name="action" value="edit">Изменить комментарий</button>

</form>

</div>

<?php endforeach; ?>

</div>

</main>

1. feedback\_adm.php

<?php

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

try {

$pdo = new PDO($dsn, $username, $password, $options);

// Определяем количество отзывов на странице

$reviewsPerPage = 4;

$currentPageNum = isset($\_GET['page\_num']) ? (int)$\_GET['page\_num'] : 1;

$offset = ($currentPageNum - 1) \* $reviewsPerPage;

// Подготовка SQL-запроса с пагинацией и фильтрацией по статусу

$stmt = $pdo->prepare("SELECT \* FROM feedback WHERE status = 'approve' ORDER BY RAND() LIMIT :offset, :reviewsPerPage");

$stmt->bindParam(':offset', $offset, PDO::PARAM\_INT);

$stmt->bindParam(':reviewsPerPage', $reviewsPerPage, PDO::PARAM\_INT);

// Выполнение запроса

$stmt->execute();

$reviews = $stmt->fetchAll();

// Подсчитываем общее количество страниц

$stmtTotal = $pdo->prepare("SELECT COUNT(\*) FROM feedback WHERE status = 'approve'");

$stmtTotal->execute();

$totalReviews = $stmtTotal->fetchColumn();

$totalPages = ceil($totalReviews / $reviewsPerPage);

// Обработка формы отправки отзыва

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

// Получаем данные из формы

$name = htmlspecialchars(strip\_tags($\_POST['name']));

$contact = htmlspecialchars(strip\_tags($\_POST['contact']));

$text = htmlspecialchars(strip\_tags($\_POST['text']));

$status = 'new'; // Статус отзыва — новый

// Подготовка SQL-запроса для вставки отзыва в базу данных

$stmt = $pdo->prepare("INSERT INTO feedback (name, contact, text, status, data)

VALUES (:name, :contact, :text, :status, NOW())");

// Привязываем параметры

$stmt->bindParam(':name', $name);

$stmt->bindParam(':contact', $contact);

$stmt->bindParam(':text', $text);

$stmt->bindParam(':status', $status);

// Выполняем запрос

if ($stmt->execute()) {

// Отзыв успешно добавлен, можно вывести сообщение об успехе

echo "<p>Ваш отзыв успешно отправлен!</p>";

} else {

echo "<p>Ошибка при отправке отзыва.</p>";

}

}

} catch (PDOException $e) {

echo "Ошибка подключения к базе данных: " . $e->getMessage();

}

?>

<main>

<div class="feedback\_display">

<h2 class="center">Отзывы</h2>

<div class="feedbacks\_list">

<?php if (!empty($reviews)): ?>

<?php foreach ($reviews as $review): ?>

<div class="feedback">

<p class="feedback\_name"><?= htmlspecialchars($review['name']); ?></p>

<p class="feedback\_date"><?= date('d.m.Y', strtotime($review['data'])); ?></p>

<p class="feedback\_text"><?= htmlspecialchars($review['text']); ?></p>

</div>

<?php endforeach; ?>

<?php else: ?>

<p>Нет отзывов для отображения.</p>

<?php endif; ?>

</div>

<!-- Пагинация -->

<ol class="carousel\_\_navigation-list center\_nav">

<?php for ($i = 1; $i <= $totalPages; $i++): ?>

<li class="carousel\_\_navigation-item">

<a href="?page=feedback&page\_num=<?= $i ?>" class="carousel\_\_navigation-button"><?= $i ?></a>

</li>

<?php endfor; ?>

</ol>

<div class="feedback\_form" id="feedbackForm">

<form method="post" id="messageform">

<h2 class="center">Оставить отзыв</h2>

<ul class="form\_ul">

<li>

<label>Ваше имя:</label>

<input type="text" name="name" placeholder="Ваше имя" required>

</li>

<li>

<label>Контакт для связи:</label>

<input type="text" name="contact" placeholder="Телефон/Почта">

</li>

<li>

<textarea id="text" name="text" rows="4" cols="50" placeholder="Ваш отзыв" required></textarea>

</li>

<li class="center">

<input type="submit" value="Оставить">

</li>

</ul>

</form>

</div>

</div>

</main>

<script src="scripts/forms.js"></script>

1. Feedback.php

<footer>

<p class="medialinks">

Фотограф Джон Доу <br/>

<a href="https://www.behance.net/" target="\_blank">behance</a>

<a href="https://www.flickr.com/" target="\_blank">Flickr</a>

</p>

</footer>

1. Footer.php

<header>

<menu>

<ul>

<li>

<a href="?page=home" <?php if (array\_key\_exists($page, $routes) && file\_exists($routes[$page]) && $page === 'home') {echo 'class="active\_menu"';} ?>>Главная</a>

</li>

<li>

<a href="?page=about" <?php if (array\_key\_exists($page, $routes) && file\_exists($routes[$page]) && $page === 'about') {echo 'class="active\_menu"';} ?>>Обо мне</a>

</li>

<li>

<a href="?page=portfolio" <?php if (array\_key\_exists($page, $routes) && file\_exists($routes[$page]) && $page === 'portfolio') {echo 'class="active\_menu"';} ?>>Портфолио</a>

</li>

<li>

<a href="?page=prices" <?php if (array\_key\_exists($page, $routes) && file\_exists($routes[$page]) && $page === 'prices') {echo 'class="active\_menu"';} ?>>Цены</a>

</li>

<li>

<a href="?page=order" <?php if (array\_key\_exists($page, $routes) && file\_exists($routes[$page]) && $page === 'order') {echo 'class="active\_menu"';} ?>>Заказать съемку</a>

</li>

<li>

<a href="?page=feedback" <?php if (array\_key\_exists($page, $routes) && file\_exists($routes[$page]) && $page === 'feedback') {echo 'class="active\_menu"';} ?>>Отзывы</a>

</li>

<li>

<a href="?page=contacts" <?php if (array\_key\_exists($page, $routes) && file\_exists($routes[$page]) && $page === 'contacts') {echo 'class="active\_menu"';} ?>>Контакты</a>

</li>

</ul>

</menu>

</header>

1. Header.php

<main>

<div class="main\_left">

<section class="carousel" aria-label="Gallery">

<ol class="carousel\_\_viewport">

<li id="carousel\_\_slide1" tabindex="0" class="carousel\_\_slide">

<img src="https://img.freepik.com/free-photo/people-walking-japan-street-nighttime\_23-2148942945.jpg?t=st=1733831859~exp=1733835459~hmac=a9512ba088cb12577b2f786cba02f77ccbf521a8ce8b50b82ce46a9b63461874&w=2000">

</li>

<li id="carousel\_\_slide2" tabindex="0" class="carousel\_\_slide">

<img src="https://img.freepik.com/free-photo/narrow-japan-street-with-lanterns-daytime\_23-2148942948.jpg?t=st=1733834587~exp=1733838187~hmac=70850814600815b355f750ad36c477f79f448fce60f856adf4cbae7598926fac&w=2000">

</li>

<li id="carousel\_\_slide3" tabindex="0" class="carousel\_\_slide">

<img src="https://img.freepik.com/free-photo/medium-shot-smiley-friends-reunion-cold-weather\_23-2149329365.jpg?t=st=1733834706~exp=1733838306~hmac=0eb829b75020d37bcc498a9f8c752b8959a0a8b1d5f571f9a38097c8ea0cd5d1&w=2000">

</li>

<li id="carousel\_\_slide4" tabindex="0" class="carousel\_\_slide">

<img src="https://img.freepik.com/free-photo/young-woman-new-york-city-daytime\_23-2149488495.jpg?t=st=1733834766~exp=1733838366~hmac=09f4d87216644bbe6458aa47b53f97060a777c5958ab00a622ee2a2fe1997dfd&w=2000">

</li>

</ol>

<aside class="carousel\_\_navigation">

<ol class="carousel\_\_navigation-list">

<li class="carousel\_\_navigation-item">

<a href="#carousel\_\_slide1" class="carousel\_\_navigation-button">Go to slide 1</a>

</li>

<li class="carousel\_\_navigation-item">

<a href="#carousel\_\_slide2" class="carousel\_\_navigation-button">Go to slide 2</a>

</li>

<li class="carousel\_\_navigation-item">

<a href="#carousel\_\_slide3" class="carousel\_\_navigation-button">Go to slide 3</a>

</li>

<li class="carousel\_\_navigation-item">

<a href="#carousel\_\_slide4" class="carousel\_\_navigation-button">Go to slide 4</a>

</li>

</ol>

</aside>

</section>

</div>

<div class="main\_right">

<h2>Джон Доу</h2>

<p>Добро пожаловать в мир Джона Доу — фотографа, который превращает мгновения городской суеты и природной тишины в уникальные истории. Работая на улицах Японии и Южной Кореи, он запечатлевает ритм мегаполисов, неожиданные эмоции случайных прохожих и нежность портретов, наполненных искренностью. Каждый кадр Джона — это мост между прошлым и настоящим, где традиции азиатской культуры переплетаются с динамикой современной жизни.</p>

<p>Вдохновленный эстетикой деревенского края и природными текстурами, Джон ищет красоту в простых моментах — будь то свет утреннего солнца на старой улочке Киото или тень сакуры на стенах современного Сеула. Его работы — это взгляд 30-летнего жителя мегаполиса, который видит гармонию в контрастах и превращает обыденное в искусство..</p>

</div>

<div class="main\_form">

<?php

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

try {

$pdo = new PDO($dsn, $username, $password, $options);

// Получение данных из формы

$name = htmlspecialchars(strip\_tags($\_POST['name']));

$contact = htmlspecialchars(strip\_tags($\_POST['contact']));

$message = htmlspecialchars(strip\_tags($\_POST['message']));

$status = 'new'; // Устанавливаем статус сообщения как "новое"

// Подготовка SQL-запроса

$stmt = $pdo->prepare("INSERT INTO messages (name, contact, text, status, data)

VALUES (:name, :contact, :message, :status, NOW())");

// Привязка параметров и выполнение запроса

$stmt->bindParam(':name', $name);

$stmt->bindParam(':contact', $contact);

$stmt->bindParam(':message', $message);

$stmt->bindParam(':status', $status);

// Выполнение запроса

if ($stmt->execute()) {

} else {

echo "<p>Ошибка при отправке сообщения.</p>";

}

} catch (PDOException $e) {

echo "Ошибка подключения к базе данных: " . $e->getMessage();

}

}

?>

<form method="POST" id="messageform">

<h2>Связаться</h2>

<ul class="form\_ul">

<li>

<label for="main\_name">Ваше имя</label>

<input type="text" id="main\_name" name="name" placeholder="Как к вам обращаться" required>

</li>

<li>

<label for="main\_contact">Контакт для связи</label>

<input type="text" id="main\_contact" name="contact" placeholder="Телефон/Почта" required>

</li>

<li>

<textarea id="main\_message" name="message" rows="4" cols="50" placeholder="Текст вашего сообщения." required></textarea>

</li>

<li class="center">

<input type="submit" value="Написать">

</li>

</ul>

</form>

<script src="scripts/forms.js"></script>

</div>

</main>

1. Home.php

<?php

$routes = ['home' => 'home.php', 'about' => 'about.php', 'portfolio' => 'portfolio.php', 'order' => 'order.php', 'prices' => 'prices.php', 'contacts' => 'contacts.php', 'feedback' => 'feedback.php', 'portfolio\_element' => 'portfolio\_element.php'];

$titles = ['home' => 'Главная', 'about' => 'Обо мне', 'portfolio' => 'Портфолио', 'order' => 'Заказать', 'prices' => 'Цены', 'contacts' => 'Контакты', 'feedback' => 'Отзывы', 'portfolio\_element' => 'Портфолио'];

$page = isset($\_GET['page']) ? htmlspecialchars($\_GET['page']) : 'home';

?>

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<!-- Шрифты -->

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100..900;1,100..900&family=Oswald:wght@200..700&display=swap" rel="stylesheet">

<!-- Стили -->

<link rel="stylesheet" href="style/main.css">

<title>

<?php if (array\_key\_exists($page, $routes) && file\_exists($routes[$page])) { echo $titles[$page].' | ';} ?> Фотограф Джон Доу

</title>

</head>

<body>

<div class="main">

<?php

include 'header.php';

if (array\_key\_exists($page, $routes) && file\_exists($routes[$page])) {

include $routes[$page];

} elseif ($page == 'admin'){

header("Location: admin.php");

} else {

include '404.php';

}

include 'footer.php';

?>

</div>

</body>

</html>

1. Index.php

<form method="POST" id="login\_form" action="admin.php?action=login">

<input type="text" name="log">

<input type="password" name="pass">

<input type="submit" value="Войти">

</form>

<a class="back" href="index.php?page=home">на главную страницу</a>

1. Login.php

<main>

<div class="feddbackadm">

<h2 class="center">Модерация сообщений</h2>

<?php

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

$pdo = new PDO($dsn, $username, $password, $options);

// Фильтрация по статусу

if (isset($\_GET['message\_status'])) {

$statusFilter = htmlspecialchars(strip\_tags($\_GET['message\_status']));

}

// Параметры сортировки

$order = $\_GET['order'] ?? 'data';

$direction = $\_GET['direction'] ?? 'DESC';

$limit = (int)($\_GET['limit'] ?? 10);

$offset = (int)($\_GET['page\_num'] ?? 0) \* $limit;

$statusFilter = $\_GET['message\_status'] ?? null;

$allowedStatuses = ['new', 'readed'];

$statusFilter = in\_array($statusFilter, $allowedStatuses) ? $statusFilter : null;

$allowedColumns = ['name', 'data', 'status'];

$allowedDirections = ['ASC', 'DESC'];

$order = in\_array($order, $allowedColumns) ? $order : 'data';

$direction = in\_array($direction, $allowedDirections) ? $direction : 'DESC';

$limit = in\_array($limit, [10, 20, 50, 100]) ? $limit : 10;

// Запрос с фильтрацией по статусу, если задан фильтр

if ($statusFilter) {

$stmt = $pdo->prepare("SELECT id\_message, name, contact, data, text, status

FROM messages WHERE status = :statusFilter

ORDER BY $order $direction LIMIT :limit OFFSET :offset");

$stmt->bindValue(':statusFilter', $statusFilter, PDO::PARAM\_STR);

} else {

$stmt = $pdo->prepare("SELECT id\_message, name, contact, data, text, status

FROM messages

ORDER BY $order $direction LIMIT :limit OFFSET :offset");

}

$stmt->bindValue(':limit', $limit, PDO::PARAM\_INT);

$stmt->bindValue(':offset', $offset, PDO::PARAM\_INT);

$stmt->execute();

$messages = $stmt->fetchAll(PDO::FETCH\_ASSOC);

// Общее количество сообщений

$totalMessages = $pdo->query("SELECT COUNT(\*) FROM messages")->fetchColumn();

$totalPages = ceil($totalMessages / $limit);

$currentPageNum = isset($\_GET['page\_num']) ? (int)$\_GET['page\_num'] : 0;

// Действия с сообщениями

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['action'], $\_POST['message\_id'])) {

$messageId = (int)$\_POST['message\_id'];

$action = $\_POST['action'];

if ($action === 'readed') {

$pdo->prepare("UPDATE messages SET status = 'readed' WHERE id\_message = :id")

->execute(['id' => $messageId]);

} elseif ($action === 'delete') {

$pdo->prepare("DELETE FROM messages WHERE id\_message = :id")

->execute(['id' => $messageId]);

}

header('Location: ' . $\_SERVER['REQUEST\_URI']);

exit;

}

?>

<!-- Фильтр по статусу сообщений -->

<div class="pagination">

<form method="GET">

<label for="message\_status">Статус</label>

<select name="message\_status" id="message\_status">

<option value="">Все</option>

<option value="new" <?= $statusFilter == 'new' ? 'selected' : '' ?>>Новое</option>

<option value="readed" <?= $statusFilter == 'readed' ? 'selected' : '' ?>>Прочитано</option>

</select>

<label for="limit">Сообщений на странице:</label>

<select name="limit" id="limit">

<option value="10" <?= $limit == 10 ? 'selected' : '' ?>>10</option>

<option value="20" <?= $limit == 20 ? 'selected' : '' ?>>20</option>

<option value="50" <?= $limit == 50 ? 'selected' : '' ?>>50</option>

<option value="100" <?= $limit == 100 ? 'selected' : '' ?>>100</option>

</select>

<label for="order">Сортировка по:</label>

<select name="order" id="order">

<option value="data" <?= $order == 'data' ? 'selected' : '' ?>>Дате</option>

<option value="name" <?= $order == 'name' ? 'selected' : '' ?>>Имени</option>

<option value="status" <?= $order == 'status' ? 'selected' : '' ?>>Статусу</option>

</select>

<label for="direction">Направление сортировки:</label>

<select name="direction" id="direction">

<option value="ASC" <?= $direction == 'ASC' ? 'selected' : '' ?>>По возрастанию</option>

<option value="DESC" <?= $direction == 'DESC' ? 'selected' : '' ?>>По убыванию</option>

</select>

<input type="hidden" name="page\_num" value="<?= $currentPageNum ?>">

<input type="hidden" name="page" value="messages">

<button type="submit">Применить</button>

</form>

</div>

<!-- Пагинация -->

<div class="pagination">

<p>

<?php

for ($i = 0; $i < $totalPages; $i++):

$isCurrentPage = ($i == $currentPageNum);

?>

<a href="?page\_num=<?= $i ?>&limit=<?= $limit ?>&order=<?= $order ?>&direction=<?= $direction ?>&message\_status=<?= htmlspecialchars($statusFilter) ?>" class="<?= $isCurrentPage ? 'current\_p' : '' ?>">

<?= $i + 1 ?>

</a>

<?php endfor; ?>

</p>

</div>

<!-- Вывод всех сообщений -->

<div class="messages">

<?php foreach ($messages as $message): ?>

<div class="message">

<p class="message\_name"><?= htmlspecialchars($message['name']) ?></p>

<p class="message\_contact"><?= htmlspecialchars($message['contact']) ?></p>

<p class="message\_dat">Дата: <?= htmlspecialchars($message['data']) ?></p>

<p class="message\_text">Текст: <?= htmlspecialchars($message['text']) ?></p>

<p class="message\_status">Статус: <?= htmlspecialchars($message['status']) ?></p>

<!-- Формы для изменения статуса -->

<form method="POST">

<input type="hidden" name="message\_id" value="<?= $message['id\_message'] ?>">

<button type="submit" name="action" value="readed">Прочитано</button>

<button type="submit" name="action" value="delete">Удалить</button>

</form>

</div>

<?php endforeach; ?>

</div>

</div>

</main>

1. messages\_adm.php

<main>

<div class="main\_form">

<?php

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

try {

$pdo = new PDO($dsn, $username, $password, $options);

// Получаем данные из формы

$name = htmlspecialchars(strip\_tags($\_POST['name']));

$contact = htmlspecialchars(strip\_tags($\_POST['contact']));

$shooting\_type = htmlspecialchars(strip\_tags($\_POST['type']));

$pref\_date = htmlspecialchars(strip\_tags($\_POST['pref\_date']));

$message = htmlspecialchars(strip\_tags($\_POST['message']));

$status = 'new'; // Устанавливаем статус заказа как "новый"

// Подготовка SQL-запроса

$stmt = $pdo->prepare("INSERT INTO orders (name, contact, type, pref\_date, description, status, date)

VALUES (:name, :contact, :type, :pref\_date, :description, :status, NOW())");

// Привязываем параметры

$stmt->bindParam(':name', $name);

$stmt->bindParam(':contact', $contact);

$stmt->bindParam(':type', $shooting\_type);

$stmt->bindParam(':pref\_date', $pref\_date);

$stmt->bindParam(':description', $message);

$stmt->bindParam(':status', $status);

// Выполняем запрос

if ($stmt->execute()) {

// echo "<p>Ваш заказ успешно отправлен!</p>";

} else {

echo "<p>Ошибка при отправке заказа.</p>";

}

} catch (PDOException $e) {

echo "Ошибка подключения к базе данных: " . $e->getMessage();

}

}

?>

<form method="POST" id="messageform">

<h2 class="center">Заказать съемку</h2>

<ul class="form\_ul">

<li>

<label for="name">Ваше имя</label>

<input type="text" id="name" name="name" placeholder="Ваше имя" required>

</li>

<li>

<label for="contact">Контакт для связи</label>

<input type="text" id="contact" name="contact" placeholder="Телефон/Почта" required>

</li>

<li>

<label for="type">Тип съемки</label>

<select id="type" name="type" required>

<option value="">Тип съемки</option>

<option value="Портретная фотосъемка">Портретная фотосъемка</option>

<option value="Свадебная фотосессия">Свадебная фотосессия</option>

<option value="Корпоративная съемка">Корпоративная съемка</option>

<option value="Фотосессия на открытом воздухе">Фотосессия на открытом воздухе</option>

<option value="Тематическая студийная съемка">Тематическая студийная съемка</option>

</select>

</li>

<li>

<label for="pref\_date">Желаемая дата</label>

<input type="date" id="pref\_date" name="pref\_date" placeholder="дд.мм.гггг" required>

</li>

<li>

<textarea id="message" name="message" rows="4" cols="50" placeholder="Любые дополнительные данные, которые вы хотите сообщить о себе и съемке"></textarea>

</li>

<li class="center">

<input type="submit" value="Написать">

</li>

</ul>

</form>

<div class="price\_container">

<h3 class="center">Цены на пакетные предложения</h3>

<ul class="price-list">

<li>Портретная фотосъемка (за час): <span class="price">50,000 ₩ / $40</span></li>

<li>Свадебная фотосессия (за час): <span class="price">200,000 ₩ / $160</span></li>

<li>Корпоративная съемка (за час): <span class="price">133,333 ₩ / $107</span></li>

<li>Фотосессия на открытом воздухе (за час): <span class="price">125,000 ₩ / $100</span></li>

<li>Тематическая студийная съемка (за час): <span class="price">175,000 ₩ / $140</span></li>

<li>Пакет "День со съемкой": <span class="price">1,500,000 ₩ / $1,200</span></li>

<li>Редактирование и ретушь (за 10 фото): <span class="price">100,000 ₩ / $80</span></li>

<li>Печать фотографий (набор из 20): <span class="price">50,000 ₩ / $40</span></li>

</ul>

<p class="price-note">

Обратите внимание:<br>

- Цены для фотосессий указаны за один час съемки, кроме пакета "День со съемкой".<br>

- Цены указаны без учета транспортных расходов для съемок за пределами Сеула.<br>

- Печать фотографий возможна только на территории Кореи.<br>

- Все услуги могут быть адаптированы под ваши потребности. Свяжитесь для уточнения деталей.

</p>

</div>

</div>

</main>

<script src="scripts/forms.js"></script>

<script>

// Форматируем дату в нужный формат (гггг.мм.дд)

window.onload = function() {

var today = new Date();

var year = today.getFullYear();

var month = ("0" + (today.getMonth() + 1)).slice(-2); // Добавляем ведущий 0 если месяц однозначный

var day = ("0" + today.getDate()).slice(-2); // Добавляем ведущий 0 если день однозначный

var formattedDate = year + "." + month + "." + day;

// Вставляем отформатированную дату в поле формы

document.getElementById('pref\_date').value = formattedDate;

};

</script>

1. Order.php

<main>

<div class="feddbackadm">

<h2 class="center">Модерация заказов</h2>

<?php

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

$pdo = new PDO($dsn, $username, $password, $options);

// Фильтрация по статусу

if (isset($\_GET['order\_status'])) {

$statusFilter = htmlspecialchars(strip\_tags($\_GET['order\_status']));

}

// Параметры сортировки

$order = $\_GET['order'] ?? 'date';

$direction = $\_GET['direction'] ?? 'DESC';

$limit = (int)($\_GET['limit'] ?? 10);

$offset = (int)($\_GET['page\_num'] ?? 0) \* $limit;

$statusFilter = $\_GET['order\_status'] ?? null;

$allowedStatuses = ['new', 'in\_progress', 'completed', 'canceled'];

$statusFilter = in\_array($statusFilter, $allowedStatuses) ? $statusFilter : null;

$allowedColumns = ['name', 'date', 'status', 'type'];

$allowedDirections = ['ASC', 'DESC'];

$order = in\_array($order, $allowedColumns) ? $order : 'date';

$direction = in\_array($direction, $allowedDirections) ? $direction : 'DESC';

$limit = in\_array($limit, [10, 20, 50, 100]) ? $limit : 10;

// Запрос с фильтрацией по статусу, если задан фильтр

if ($statusFilter) {

$stmt = $pdo->prepare("SELECT id\_order, name, contact, date, pref\_date, type, description, status

FROM orders WHERE status = :statusFilter

ORDER BY $order $direction LIMIT :limit OFFSET :offset");

$stmt->bindValue(':statusFilter', $statusFilter, PDO::PARAM\_STR);

} else {

$stmt = $pdo->prepare("SELECT id\_order, name, contact, date, pref\_date, type, description, status

FROM orders

ORDER BY $order $direction LIMIT :limit OFFSET :offset");

}

$stmt->bindValue(':limit', $limit, PDO::PARAM\_INT);

$stmt->bindValue(':offset', $offset, PDO::PARAM\_INT);

$stmt->execute();

$orders = $stmt->fetchAll(PDO::FETCH\_ASSOC);

// Общее количество заказов

$totalOrders = $pdo->query("SELECT COUNT(\*) FROM orders")->fetchColumn();

$totalPages = ceil($totalOrders / $limit);

$currentPageNum = isset($\_GET['page\_num']) ? (int)$\_GET['page\_num'] : 0;

// Обновление статуса заказа

if ($\_SERVER['REQUEST\_METHOD'] === 'POST') {

$idOrder = $\_POST['id\_order'] ?? null;

$newStatus = $\_POST['status'] ?? null;

if ($idOrder && $newStatus) {

$stmt = $pdo->prepare("UPDATE orders SET status = :status WHERE id\_order = :id\_order");

$stmt->bindValue(':status', $newStatus, PDO::PARAM\_STR);

$stmt->bindValue(':id\_order', $idOrder, PDO::PARAM\_INT);

$stmt->execute();

}

}

?>

<!-- Фильтр по статусу заказов -->

<div class="pagination">

<form method="GET">

<label for="order\_status">Статус</label>

<select name="order\_status" id="order\_status">

<option value="">Все</option>

<option value="new" <?= $statusFilter == 'new' ? 'selected' : '' ?>>Новый</option>

<option value="in\_progress" <?= $statusFilter == 'in\_progress' ? 'selected' : '' ?>>В процессе</option>

<option value="completed" <?= $statusFilter == 'completed' ? 'selected' : '' ?>>Завершен</option>

<option value="canceled" <?= $statusFilter == 'canceled' ? 'selected' : '' ?>>Отменен</option>

</select>

<label for="limit">Заказов на странице:</label>

<select name="limit" id="limit">

<option value="10" <?= $limit == 10 ? 'selected' : '' ?>>10</option>

<option value="20" <?= $limit == 20 ? 'selected' : '' ?>>20</option>

<option value="50" <?= $limit == 50 ? 'selected' : '' ?>>50</option>

<option value="100" <?= $limit == 100 ? 'selected' : '' ?>>100</option>

</select>

<label for="order">Сортировка по:</label>

<select name="order" id="order">

<option value="date" <?= $order == 'date' ? 'selected' : '' ?>>Дате</option>

<option value="name" <?= $order == 'name' ? 'selected' : '' ?>>Имени</option>

<option value="status" <?= $order == 'status' ? 'selected' : '' ?>>Статусу</option>

</select>

<label for="direction">Направление сортировки:</label>

<select name="direction" id="direction">

<option value="ASC" <?= $direction == 'ASC' ? 'selected' : '' ?>>По возрастанию</option>

<option value="DESC" <?= $direction == 'DESC' ? 'selected' : '' ?>>По убыванию</option>

</select>

<input type="hidden" name="page\_num" value="<?= $currentPageNum ?>">

<input type="hidden" name="page" value="orders">

<button type="submit">Применить</button>

</form>

</div>

<!-- Пагинация -->

<div class="pagination">

<p>

<?php

for ($i = 0; $i < $totalPages; $i++):

$isCurrentPage = ($i == $currentPageNum);

?>

<a href="?page\_num=<?= $i ?>&limit=<?= $limit ?>&order=<?= $order ?>&direction=<?= $direction ?>&order\_status=<?= htmlspecialchars($statusFilter) ?>" class="<?= $isCurrentPage ? 'current\_p' : '' ?>">

<?= $i + 1 ?>

</a>

<?php endfor; ?>

</p>

</div>

<!-- Вывод всех заказов -->

<?php foreach ($orders as $order): ?>

<div class="order">

<p class="order\_name"><?= htmlspecialchars($order['name']) ?></p>

<p class="order\_contact"><?= htmlspecialchars($order['contact']) ?></p>

<p class="order\_date">Дата отправки: <?= htmlspecialchars($order['date']) ?></p>

<p class="order\_pref\_date">Предпочтительная дата: <?= htmlspecialchars($order['pref\_date']) ?></p>

<p class="order\_type">Тип заказа: <?= htmlspecialchars($order['type']) ?></p>

<p class="order\_description">Описание: <?= htmlspecialchars($order['description']) ?></p>

<p class="order\_status">Статус: <?= htmlspecialchars($order['status']) ?></p>

<!-- Форма для изменения статуса -->

<form method="POST" class="status-form">

<input type="hidden" name="id\_order" value="<?= htmlspecialchars($order['id\_order']) ?>">

<input type="hidden" name="page" value="orders">

<p>Изменить статус:</p>

<button type="submit" name="status" value="new" <?= $order['status'] === 'new' ? 'disabled' : '' ?>>Новый</button>

<button type="submit" name="status" value="in\_progress" <?= $order['status'] === 'in\_progress' ? 'disabled' : '' ?>>В процессе</button>

<button type="submit" name="status" value="completed" <?= $order['status'] === 'completed' ? 'disabled' : '' ?>>Завершен</button>

<button type="submit" name="status" value="canceled" <?= $order['status'] === 'canceled' ? 'disabled' : '' ?>>Отменен</button>

</form>

</div>

<?php endforeach; ?>

</div>

</main>

1. orders\_adm.php

<?php

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

$pdo = new PDO($dsn, $username, $password, $options);

// Обработка загрузки

if ($\_SERVER['REQUEST\_METHOD'] === 'POST') {

$title = $\_POST['title'];

$description = $\_POST['description'];

$uploadedFiles = $\_FILES['photos'];

$link = $\_POST['link'];

$pdo->beginTransaction();

try {

// Добавляем элемент в таблицу портфолио

$stmt = $pdo->prepare("INSERT INTO portfolio (title, description, likes, link) VALUES (?, ?, 0, ?)");

$stmt->execute([$title, $description, $link]);

$portfolioId = $pdo->lastInsertId();

// Создаем папку для фотографий

$folderPath = "photobase/$portfolioId";

if (!mkdir($folderPath, 0777, true)) {

throw new Exception("Не удалось создать папку.");

}

// Сохраняем файлы и добавляем их в таблицу фотографий

foreach ($uploadedFiles['tmp\_name'] as $index => $tmpName) {

if ($uploadedFiles['error'][$index] === UPLOAD\_ERR\_OK) {

// Сохраняем файл с оригинальным именем

$filename = $uploadedFiles['name'][$index];

$destination = "$folderPath/$filename"; // Добавляем имя файла в путь

// Перемещаем файл в нужную папку

if (move\_uploaded\_file($tmpName, $destination)) {

// Добавляем запись в таблицу фотографий с именем файла

$stmt = $pdo->prepare("INSERT INTO photos (portfolio\_id, folder\_id, filename) VALUES (?, ?, ?)");

$stmt->execute([$portfolioId, $portfolioId, $filename]);

} else {

throw new Exception("Ошибка при сохранении файла: $filename");

}

}

}

$pdo->commit();

header("Location: admin.php?page=content");

} catch (Exception $e) {

$pdo->rollBack();

echo "Ошибка: " . $e->getMessage();

}

}

?>

<main>

<form class="form\_ul" action="portfolio\_adm.php" method="post" enctype="multipart/form-data">

<h2 class="center">Портфолио</h2>

<ul>

<li>

<label for="title">Название элемента портфолио:</label>

<input type="text" name="title" required>

</li>

<li>

<label for="link">Ссылка на публикацию:</label>

<input type="text" name="link">

</li>

<li>

<textarea name="description" placeholder="Описание" required></textarea>

</li>

<li>

<label for="photos">Выберите файлы:</label>

<input type="file" name="photos[]" multiple required>

</li>

<li>

<button type="submit">Загрузить</button>

</li>

</ul>

</form>

<?php

if (isset($\_GET['delete\_id'])) {

$deleteId = (int) $\_GET['delete\_id'];

$pdo->beginTransaction();

try {

// Удаляем связанные фотографии из таблицы photos

$stmt = $pdo->prepare("DELETE FROM photos WHERE portfolio\_id = ?");

$stmt->execute([$deleteId]);

// Удаляем запись из таблицы portfolio

$stmt = $pdo->prepare("DELETE FROM portfolio WHERE id = ?");

$stmt->execute([$deleteId]);

$pdo->commit();

echo "Запись успешно удалена!";

} catch (Exception $e) {

$pdo->rollBack();

echo "Ошибка при удалении: " . $e->getMessage();

}

}

// Выводим таблицу с записями из таблицы portfolio

$stmt = $pdo->query("SELECT \* FROM portfolio");

$portfolioItems = $stmt->fetchAll();

?>

<div class="feedback\_display">

<h2 class="center">Управление портфолио</h2>

<table class="admintable">

<thead>

<tr>

<th>Название</th>

<th>Описание</th>

<th>Ссылка</th>

<th>Действия</th>

</tr>

</thead>

<tbody>

<?php foreach ($portfolioItems as $item): ?>

<tr>

<td><?php echo htmlspecialchars($item['title']); ?></td>

<td><?php echo htmlspecialchars($item['description']); ?></td>

<td><?php echo htmlspecialchars(isset($item['link']) ? htmlspecialchars($item['link']) : 'none'); ?></td>

<td>

<!-- Кнопка для удаления -->

<a href="?page=content&delete\_id=<?php echo $item['id']; ?>" onclick="return confirm('Вы уверены, что хотите удалить этот элемент?')">Удалить</a>

</td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

</div>

</main>

1. portfolio\_adm.php

<?php

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

try {

$pdo = new PDO($dsn, $username, $password, $options);

} catch (PDOException $e) {

echo "Connection failed: " . $e->getMessage();

exit;

}

// Получение параметра portph\_el из URL

$portph\_el = isset($\_GET['portph\_el']) ? (int) $\_GET['portph\_el'] : 0;

// Получение информации о конкретном элементе портфолио

$stmt = $pdo->prepare("SELECT \* FROM portfolio WHERE id = :id");

$stmt->bindParam(':id', $portph\_el, PDO::PARAM\_INT);

$stmt->execute();

$portfolioItem = $stmt->fetch();

if ($portfolioItem) {

// Получение фотографий, ассоциированных с этим элементом портфолио

$stmt = $pdo->prepare("SELECT \* FROM photos WHERE portfolio\_id = :portfolio\_id");

$stmt->bindParam(':portfolio\_id', $portph\_el, PDO::PARAM\_INT);

$stmt->execute();

$photos = $stmt->fetchAll();

// Обработка формы лайка

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['like\_id'])) {

$like\_id = (int) $\_POST['like\_id'];

// Обновление количества лайков

$stmt = $pdo->prepare("UPDATE portfolio SET likes = likes + 1 WHERE id = :id");

$stmt->bindParam(':id', $like\_id, PDO::PARAM\_INT);

$stmt->execute();

exit;

}

} else {

echo "Элемент портфолио не найден.";

exit;

}

?>

<main>

<div class="portf\_view">

<div class="portf\_descrip">

<h2 class="center">Портфолио</h2>

<div class="portf\_el\_h3">

<a class="back\_portfolio" href="?page=portfolio">Назад</a>

<h3 class="center"><?php echo htmlspecialchars($portfolioItem['title']); ?></h3>

<div class="like-button">

<p><?php echo htmlspecialchars($portfolioItem['likes']); ?></p>

<form class="like-form" method="POST" id="messageform">

<input type="hidden" name="like\_id" value="<?php echo $portfolioItem['id']; ?>">

<button type="submit">

<svg xmlns="http://www.w3.org/2000/svg" width="18" height="18" fill="black" viewBox="0 0 24 24">

<path d="M12 21.35l-1.45-1.32C5.4 15.36 2 12.28 2 8.5 2 5.42 4.42 3 7.5 3c1.74 0 3.41.81 4.5 2.09C13.09 3.81 14.76 3 16.5 3 19.58 3 22 5.42 22 8.5c0 3.78-3.4 6.86-8.55 11.54L12 21.35z" />

</svg>

</button>

</form>

</div>

</div>

<p id="descript"><?php echo htmlspecialchars($portfolioItem['description']); ?></p>

<div class="back\_portfolio\_center">

<?php if (!empty($portfolioItem['link'])): ?>

<a class="back\_portfolio" href="<?php echo htmlspecialchars($portfolioItem['link']); ?>">Ссылка на публикацию</a>

<?php endif; ?>

</div>

</div>

<div class="gallery\_all">

<?php foreach ($photos as $photo): ?>

<img class="gallery\_item" src="photobase/<?php echo htmlspecialchars($photo['folder\_id']); ?>/<?php echo htmlspecialchars($photo['filename']); ?>" alt="<?php echo htmlspecialchars($portfolioItem['title']); ?>">

<?php endforeach; ?>

</div>

</div>

</main>

<script src="scripts/forms.js"></script>

1. portfolio\_element.php

<?php

// Подключение к базе данных

$dsn = "mysql:host=localhost;dbname=test;charset=utf8";

$username = "Lena";

$password = "Password1!";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

];

try {

$pdo = new PDO($dsn, $username, $password, $options);

// Определяем количество элементов на странице

$itemsPerPage = 12;

$currentPageNum = isset($\_GET['page\_num']) ? (int)$\_GET['page\_num'] : 1;

$offset = ($currentPageNum - 1) \* $itemsPerPage;

// Получаем 12 элементов из таблицы portfolio

$stmt = $pdo->prepare("SELECT \* FROM portfolio LIMIT :offset, :itemsPerPage");

$stmt->bindParam(':offset', $offset, PDO::PARAM\_INT);

$stmt->bindParam(':itemsPerPage', $itemsPerPage, PDO::PARAM\_INT);

$stmt->execute();

$portfolioItems = $stmt->fetchAll();

// Подсчитываем общее количество записей в таблице portfolio для пагинации

$stmtTotal = $pdo->prepare("SELECT COUNT(\*) FROM portfolio");

$stmtTotal->execute();

$totalItems = $stmtTotal->fetchColumn();

$totalPages = ceil($totalItems / $itemsPerPage);

} catch (PDOException $e) {

echo "Ошибка подключения к базе данных: " . $e->getMessage();

}

?>

<main>

<div class="portf\_view">

<div class="portf\_descrip">

<h2 class="center">Мои работы</h2>

<p>Добро пожаловать в мой творческий мир, где каждая фотография — это рассказ о жизни, эмоциях и красоте в самых неожиданных её проявлениях. Здесь собраны мгновения, которые я запечатлел на улицах Японии и Южной Кореи, от оживленных проспектов Токио до уединенных тропинок среди цветущих сакур.</p>

<p>В этом портфолио вы найдете как динамичные стрит-фотографии, передающие ритм мегаполиса, так и проникновенные портреты, в которых живут искренность и характер. Моя цель — показать гармонию контрастов: современного и традиционного, светлого и теневого, городского и природного.</p>

</div>

<!-- <div class="portf\_pagin">

<a>Все</a>

<a>Городской пейзаж</a>

<a>Портреты</a>

<a>События</a>

</div> -->

<div class="portf\_all">

<?php foreach ($portfolioItems as $item): ?>

<?php

// Получаем первую фотографию для каждого элемента портфолио

$stmtImage = $pdo->prepare("SELECT \* FROM photos WHERE portfolio\_id = :portfolio\_id LIMIT 1");

$stmtImage->bindParam(':portfolio\_id', $item['id'], PDO::PARAM\_INT);

$stmtImage->execute();

$image = $stmtImage->fetch();

?>

<a href="?page=portfolio\_element&portph\_el=<?= $item['id'] ?>">

<div class="portf\_item">

<img class="portf\_gallery" src="photobase/<?= $image['folder\_id'] ?>/<?= $image['filename'] ?>" alt="<?= htmlspecialchars($item['title']) ?>">

<div class="portf\_caption"><?= htmlspecialchars($item['title']) ?></div>

</div>

</a>

<?php endforeach; ?>

</div>

<!-- Пагинация -->

<ol class="carousel\_\_navigation-list center\_nav">

<?php for ($i = 1; $i <= $totalPages; $i++): ?>

<li class="carousel\_\_navigation-item">

<a href="?page=portfolio&page\_num=<?= $i ?>" class="carousel\_\_navigation-button"><?= $i ?></a>

</li>

<?php endfor; ?>

</ol>

</div>

</main>

1. Portpholo.php

<main>

<div class="price\_container">

<h2 class="center">Цены</h2>

<h3 class="center">Цены на пакетные предложения</h3>

<ul class="price-list">

<li>Портретная фотосъемка (за час): <span class="price">50,000 ₩ / $40</span></li>

<li>Свадебная фотосессия (за час): <span class="price">200,000 ₩ / $160</span></li>

<li>Корпоративная съемка (за час): <span class="price">133,333 ₩ / $107</span></li>

<li>Фотосессия на открытом воздухе (за час): <span class="price">125,000 ₩ / $100</span></li>

<li>Тематическая студийная съемка (за час): <span class="price">175,000 ₩ / $140</span></li>

<li>Пакет "День со съемкой": <span class="price">1,500,000 ₩ / $1,200</span></li>

<li>Редактирование и ретушь (за 10 фото): <span class="price">100,000 ₩ / $80</span></li>

<li>Печать фотографий (набор из 20): <span class="price">50,000 ₩ / $40</span></li>

</ul>

<p class="price-note">

Обратите внимание:<br>

- Цены для фотосессий указаны за один час съемки, кроме пакета "День со съемкой".<br>

- Цены указаны без учета транспортных расходов для съемок за пределами Сеула.<br>

- Печать фотографий возможна только на территории Кореи.<br>

- Все услуги могут быть адаптированы под ваши потребности. Свяжитесь для уточнения деталей.

</p>

<form id="price-calculator" class="calculator-form">

<h3 class="center">Калькулятор примерных цен</h3>

<ul class="form\_ul">

<li>

<label for="service-select">Выберите услугу:</label> <select id="service-select" name="service">

<option value="50000" data-usd="40">Портретная фотосъемка (за час)</option>

<option value="200000" data-usd="160">Свадебная фотосессия (за час)</option>

<option value="133333" data-usd="107">Корпоративная съемка (за час)</option>

<option value="125000" data-usd="100">Фотосессия на открытом воздухе (за час)</option>

<option value="175000" data-usd="140">Тематическая студийная съемка (за час)</option>

<option value="1500000" data-usd="1200">Пакет "День со съемкой"</option>

<option value="100000" data-usd="80">Редактирование и ретушь (за 10 фото)</option>

<option value="50000" data-usd="40">Печать фотографий (набор из 20)</option>

</select>

</li>

<li>

<label for="quantity">Количество часов/услуг:</label>

<input type="number" id="quantity" name="quantity" min="1" value="1">

</li>

<li>

<button type="button" id="calculate-btn">Рассчитать</button>

</li>

</form>

<div id="result" class="result">

<p>Итоговая стоимость: <span id="total-won">0</span> ₩ / <span id="total-usd">0</span> $</p>

</div>

</div>

<script src="scripts/prices\_calc.js"></script>

</main>

1. Prices.php

document.addEventListener('DOMContentLoaded', () => {

const form = document.getElementById('messageform');

if (window.location.search.includes('submitted=true')) {

form.reset(); // Очищаем форму

}

form.addEventListener('submit', (event) => {

event.preventDefault(); // Останавливаем стандартное поведение отправки формы

// Получаем данные формы

const formData = new FormData(form);

fetch(form.action, {

method: 'POST',

body: formData

})

.then(response => {

if (response.ok) {

// Получаем текущие параметры URL

const currentParams = new URLSearchParams(window.location.search);

// Перенаправляем на тот же URL

window.location.search = currentParams.toString();

} else {

// Обработка ошибки, если отправка не прошла

}

})

.catch(error => {

console.error('Error:', error);

});

});

});

1. Forms.js

document.getElementById('calculate-btn').addEventListener('click', function () {

const serviceSelect = document.getElementById('service-select');

const quantity = parseInt(document.getElementById('quantity').value, 10) || 1;

const selectedOption = serviceSelect.options[serviceSelect.selectedIndex];

const priceWon = parseInt(selectedOption.value, 10);

const priceUsd = parseInt(selectedOption.getAttribute('data-usd'), 10);

const totalWon = priceWon \* quantity;

const totalUsd = priceUsd \* quantity;

document.getElementById('total-won').textContent = totalWon.toLocaleString();

document.getElementById('total-usd').textContent = totalUsd.toLocaleString();

});

1. Prices\_calc.js

body {

font-size: 0;

padding: 0 0 0 0px;

margin: 0 0 0 0px;

font-family: Oswald;

font-weight: 200;

}

.main{

display: flex;

flex-direction: column;

min-height: 100vh;

flex-wrap: nowrap;

}

header {

width: 100%;

height: 60px;

background-color: #000;

color: #fff;

font-size: 18px;

font-family: Oswald;

position:sticky;

top: 0%;

z-index: 1;

}

menu {

margin: 0 auto 0 auto;

padding-top: 15px;

text-align: center;

}

menu ul {

display: inline;

}

menu li {

margin: 0 20px;

padding: 0;

display: inline-block;

}

main {

width: 80%;

min-width: 1000px;

flex: 1;

margin: 0 auto;

display: flex;

flex-direction: row;

flex-wrap: wrap;

}

h2 {

color: black;

font-size: 36px;

font-family: Montserrat;

font-weight: 700;

height: fit-content;

}

h3{

color: black;

font-size: 16px;

font-family: Montserrat;

font-weight: 700;

}

p {

color: black;

font-size: 16px;

font-family: Oswald;

font-weight: 300;

word-wrap: break-word"

}

p.top{

margin-block-start: 0;

}

a{

text-decoration: none;

font-weight: 200;

color: #fff;

}

a:active, a:hover, a:visited{

color: #fff;

}

a.back{

color: #000;

text-decoration: underline;

}

.active\_menu{

font-weight: 400;

}

/\* Домашняя \*/

.main\_left {

min-height: 720px;

width: 50%;

}

.about\_left{

width: 50%;

display: flex;

justify-content: flex-end;

align-items: flex-start;

}

.main\_right {

height: 720px;

width: 50%;

padding-top: 200px;

padding-left: 52px;

}

.about\_right{

width: 50%;

padding-left: 52px;

}

.profile\_pic{

width: 350px;

height: 350px;

object-fit: cover;

object-position: center;

}

.main\_form {

width: 800px;

margin-left: auto;

margin-right: auto;

padding-bottom: 25px;

}

.main\_form h2 {

text-align: center;

}

h2.center{

display: block;

text-align: center;

width: 100%;

}

h3.center{

display: block;

text-align: center;

width: 100%;

}

/\* О фотографе \*/

.main\_about{

width: 800px;

margin-left: auto;

margin-right: auto;

padding-bottom: 25px;

}

.main\_about h2 {

text-align: center;

}

.experience\_timeline{

list-style: none;

}

.experience\_timeline\_element{

font-weight: 600;

}

/\* Форма \*/

form {

width: 100%;

font-size: 16px;

}

.form\_ul {

list-style: none;

}

.form\_ul li {

display: flex;

justify-content: flex-end;

min-height: 50px;

align-items: baseline;

align-content: center;

}

label {

font-size: 16px;

font-weight: 300;

}

input {

font-family: Oswald;

font-weight: 200;

height: 40px;

margin-top: 8px;

border: #000 solid 1px;

border-radius: 10px;

padding-left: 10px;

}

input[type="submit"] {

background-color: #fff;

border: #000 solid 1px;

border-radius: 10px;

height: 40px;

}

select{

font-family: Oswald;

font-weight: 300;

color: #000;

height: 40px;

margin-top: 8px;

border: #000 solid 1px;

border-radius: 10px;

padding-left: 10px;

}

textarea {

width: 100%;

min-height: 100px;

resize: none;

margin-top: 8px;

font-weight: 200;

font-family: Oswald;

border: #000 solid 1px;

border-radius: 10px;

padding: 5px;

padding-left: 10px;

}

.form\_ul label {

flex: 0.9;

text-align: right;

margin-right: 8px;

vertical-align: middle;

}

.form\_ul input {

flex: 2;

}

.form\_ul select{

flex: 2;

}

.form\_ul .center {

justify-content: center;

}

.form\_ul input[type="submit"] {

flex: 0.6;

}

.form\_ul button{

font-family: Oswald;

font-weight: 200;

height: 40px;

flex: 0.3;

margin: 8px auto;

border: #000 solid 1px;

border-radius: 10px;

padding-left: 10px;

background-color: #fff;

border: #000 solid 1px;

border-radius: 10px;

height: 40px;

}

/\* Карусель \*/

@keyframes tonext {

75% {

left: 0;

}

95% {

left: 100%;

}

98% {

left: 100%;

}

99% {

left: 0;

}

}

@keyframes tostart {

75% {

left: 0;

}

95% {

left: -300%;

}

98% {

left: -300%;

}

99% {

left: 0;

}

}

@keyframes snap {

96% {

scroll-snap-align: center;

}

97% {

scroll-snap-align: none;

}

99% {

scroll-snap-align: none;

}

100% {

scroll-snap-align: center;

}

}

\* {

box-sizing: border-box;

scrollbar-color: transparent transparent;

/\* thumb and track color \*/

scrollbar-width: 0px;

}

\*::-webkit-scrollbar {

width: 0;

}

\*::-webkit-scrollbar-track {

background: transparent;

}

\*::-webkit-scrollbar-thumb {

background: transparent;

border: none;

}

\* {

-ms-overflow-style: none;

}

ol,

li {

list-style: none;

margin: 0;

padding: 0;

}

.carousel {

position: relative;

height: 720px;

}

.carousel\_\_slide img {

height: 100%;

max-width: 100%;

object-fit: cover;

object-position: center;

}

.carousel\_\_viewport {

position: absolute;

top: 0;

right: 0;

bottom: 0;

left: 0;

display: flex;

overflow-x: scroll;

scroll-behavior: smooth;

scroll-snap-type: x mandatory;

}

.carousel\_\_slide {

position: relative;

flex: 0 0 100%;

width: 100%;

background-color: #DFDFDF;

}

.carousel\_\_slide:nth-child(even) {

background-color: #DFDFDF;

}

.carousel\_\_slide:before {

position: absolute;

top: 50%;

left: 50%;

transform: translate3d(-50%, -40%, 70px);

color: #fff;

font-size: 18px;

}

.carousel\_\_snapper {

position: absolute;

top: 0;

left: 0;

width: 100%;

height: 100%;

scroll-snap-align: center;

}

@media (hover: hover) {

.carousel\_\_snapper {

animation-name: tonext, snap;

animation-timing-function: ease;

animation-duration: 4s;

animation-iteration-count: infinite;

}

.carousel\_\_slide:last-child .carousel\_\_snapper {

animation-name: tostart, snap;

}

}

@media (prefers-reduced-motion: reduce) {

.carousel\_\_snapper {

animation-name: none;

}

}

.carousel:hover .carousel\_\_snapper, .carousel:focus-within .carousel\_\_snapper {

animation-name: none;

}

.carousel\_\_navigation {

position: absolute;

right: 0;

bottom: 24px;

left: 0;

text-align: center;

}

.carousel\_\_navigation-list, .carousel\_\_navigation-item {

display: inline-block;

}

.carousel\_\_navigation-button {

display: inline-block;

width: 1rem;

height: 1rem;

background-color: #000000;

background-clip: content-box;

border: 0.25rem solid transparent;

border-radius: 50%;

font-size: 0;

transition: transform 0.2s;

}

.carousel\_\_navigation-button:hover {

transform: scale(1.2);

background-color: #555;

}

/\* Таймлайн \*/

/\* Обертка для таймлайна \*/

.timeline {

margin: 0;

position: relative;

box-sizing: border-box;

}

.timeline h3{

margin-top: 0;

margin-block-start: 0;

margin-block-end: 0.5em;

}

.timeline p{

margin-block-start: 0.5em;

}

/\* Список элементов \*/

.timeline ul {

list-style: none;

padding: 0;

margin: 0;

position: relative;

}

/\* Элементы списка \*/

.timeline ul li {

position: relative;

margin: 0;

padding-bottom: 15px;

width: 50%;

display: flex;

justify-content: flex-start;

align-items: center;

box-sizing: border-box;

}

/\* Четные элементы - слева \*/

.timeline ul li:nth-child(even) {

margin-right: calc(50% - 2px);

padding-left: 20px; /\* отступ от линии \*/

flex-direction: row-reverse; /\* меняем направление для выравнивания слева \*/

border-right: 2px #000000 solid;

}

.timeline ul li:nth-child(even) .timeline-content {

float: right;

text-align: right;

padding-right: 20px;

}

/\* Нечетные элементы - справа \*/

.timeline ul li:nth-child(odd) {

margin-left: calc(50% - 2px);

padding-right: 20px; /\* отступ от линии \*/

border-left: 2px #000000 solid;

}

.timeline ul li:nth-child(odd) .timeline-content {

float: left;

text-align: left;

padding-left: 20px;

}

/\* Точки на линии \*/

.timeline ul li:nth-child(odd)::after,

.timeline ul li:nth-child(even)::before {

content: '';

position: absolute;

top: 5px;

width: 10px;

height: 10px;

background-color: #000;

border-radius: 50%;

transform: translateY(-50%);

z-index: 1;

}

/\* Линия между точками \*/

.timeline ul li:nth-child(odd)::after {

left: -6px; /\* точка по левому краю \*/

}

.timeline ul li:nth-child(even)::before {

right: -6px; /\* точка по правому краю \*/

}

/\* Убираем линии после последнего элемента \*/

.timeline ul li:last-child, .timeline ul li:last-child{

border-left: none;

border-right: none;

}

.timeline ul li:last-child::after {

left: -4px; /\* точка по левому краю \*/

}

.timeline ul li:last-child::before {

right: -4px; /\* точка по правому краю \*/

}

/\* Footer \*/

footer{

padding-top: 10px;

padding-bottom: 20px;

min-height: fit-content;

background-color: #000;

color: #fff;

bottom: 0px;

}

footer p{

color: #fff;

font-size: 16px;

text-align: center;

}

.medialinks{

margin-left: auto;

margin-right: auto;

text-align: center;

}

.medialinks a{

text-decoration: underline;

padding-right: 15px;

}

/\* Contacts \*/

.contacts\_main{

width: 1000px;

padding-left: 250px;

padding-right: 250px;

margin-left: auto;

margin-right: auto;

padding-bottom: 25px;

font-size: 16px;

}

.contacts\_main li{

margin-bottom: 20px;

}

.contacts\_main a{

color: #000;

text-decoration: underline;

text-decoration-color: #000;

}

/\* Portfolio \*/

.portf\_descrip{

width: 60%;

min-width: 1000px;

margin-left: auto;

margin-right: auto;

font-size: 16px;

margin-bottom: 50px;

}

.portf\_pagin{

width: 100%;

min-width: 1000px;

text-align: center;

margin-bottom: 5px;

}

.portf\_pagin a{

font-size: 20px;

font-weight: 3500;

color: #000;

padding: 5px;

}

.portf\_pagin a:hover{

color: #2e2e2e;

font-weight: 400;

}

.portf\_all{

width: 100%;

min-width: 1000px;

display: grid;

grid-template-columns: repeat(4, 200px);

gap: 20px;

align-content: center;

justify-content: center;

margin-bottom: 15px;

}

.portf\_item {

position: relative;

width: 200px;

height: 200px;

}

.portf\_gallery{

width: 100%;

height: 100%;

object-fit: cover;

transition: transform 0.3s ease-in-out;

position: relative;

z-index: 0;

}

.portf\_gallery:hover{

transform: scale(1.1);

z-index: 1;

}

.center\_nav{

text-align: center;

min-width: 1000px;

display: block;

margin-left: auto;

margin-right: auto;

object-fit: cover;

object-position: center;

padding-bottom: 30px;

}

.portf\_caption {

position: absolute;

bottom: 0;

left: 0;

width: 100%;

height: 30%;

background: rgba(0, 0, 0, 0.6);

color: #fff;

font-size: 16px;

display: flex;

align-items: center;

justify-content: center;

text-align: center;

opacity: 0;

transition: opacity 0.2s ease-in-out;

transition: transform 0.3s ease-in-out;

z-index: 2;

pointer-events: none;

}

.portf\_item:hover .portf\_caption {

opacity: 1;

bottom: -5%;

left: -5%;

width: 110%;

height: 40%;

}

.portf\_view{

display: flex;

flex-flow: column;

flex-wrap: nowrap;

margin: 0 auto;

}

/\* Съемка \*/

.gallery\_all{

width: 100%;

min-width: 1000px;

display: grid;

grid-template-columns: repeat(3, 300px);

gap: 20px;

align-content: center;

justify-content: center;

margin-bottom: 35px;

}

.gallery\_item{

width: 300px;

height: 300px;

object-fit: cover;

object-position: center;

transition: transform 0.3s ease-in-out;

position: relative;

z-index: 0;

}

.gallery\_item:hover{

transform: scale(1.1);

z-index: 1;

}

a.back\_portfolio{

color: #000;

text-decoration: underline;

}

.back\_portfolio\_center{

text-align: center;

color: #000;

}

.portf\_el\_h3 {

display: flex;

flex-flow: row;

align-items: center;

align-content: center;

flex-wrap: nowrap;

}

.like-button{

display: inline-flex;

flex-wrap: nowrap;

flex-direction: row;

margin-bottom: 0;

align-items: center;

}

.like-button form{

margin-bottom: 0;

}

.like-button button{

cursor: pointer;

border: none;

background-color: inherit;

}

.like-form{

display: inline;

}

/\* Цены \*/

.price-list {

list-style-type: none;

padding: 0;

margin: 0;

}

.price-list li {

margin-bottom: 10px;

font-size: 16px;

}

.price {

font-weight: bold;

}

.price\_container{

display: flex;

flex-flow: column;

flex-wrap: nowrap;

min-width: 600px;

margin-left: auto;

margin-right: auto;

justify-content: flex-start;

align-content: center;

align-items: center;

flex-direction: column;

}

.price-note {

border-top: #000 solid 1px;

padding-top: 10px;

font-size: 16px;

color: #555;

margin-top: 20px;

line-height: 1.5;

margin-bottom: 25px;

}

.calculator-form {

width: 100%;

margin: 20px auto;

padding: 15px;

}

.result {

text-align: center;

margin-top: 15px;

font-size: 16px;

}

/\* 404 \*/

.container\_404{

width: 800px;

margin-left: auto;

margin-right: auto;

padding-bottom: 25px;

flex: 1;

}

.full-width-button {

display: block;

width: 100%;

height: 50px;

padding: 10px 0;

margin-top: 20px;

font-size: 16px;

font-weight: 300;

font-family: Oswald;

background-color: #fff;

color: #000;

border: solid #000 1px;

border-radius: 5px;

cursor: pointer;

}

.full-width-button:hover {

background-color: #000;

color: #fff;

}

/\* Feedback \*/

.feedback\_display{

display: flex;

flex-flow: column;

flex-wrap: nowrap;

width: 1000px;

margin: 0 auto;

}

.feedbacks\_list{

display: grid;

grid-template-columns: repeat(4, 220px);

gap: 20px;

align-items: start;

justify-content: center;

align-content: center;

margin-bottom: 25px;

}

.feedback{

height: 400px;

word-break: keep-all;

text-wrap: balance;

padding: 5px 20px;

border: #000 solid 1px;

border-radius: 5px;

}

.feedback\_name{

font-family: Montserrat;

font-size: 16px;

font-weight: 600;

margin-bottom: 5px;

}

.feedback\_date{

font-family: Oswald;

font-size: 14px;

font-weight: 200;

color: #2e2e2e;

margin-bottom: 20px;

margin-top: 0;

}

.feedback\_text{

font-family: Oswald;

font-size: 16px;

font-weight: 300;

color: #2e2e2e;

}

.feedback\_form{

display: flex;

width: 600px;

margin-left: auto;

margin-right: auto;

padding-bottom: 25px;

}

.feedback\_form input[type="submit"] {

flex: 0.3;

}

/\* Admin \*/

table.admintable{

font-size: 16px;

font-family: Oswald;

text-align: center;

color: #000;

border: solid #000 1px;

border-radius: 5px;

border-collapse: collapse;

margin-bottom: 50px;

}

.admintable tr{

border-bottom: #000 solid 1px;

}

.admintable a{

color: #000;

text-decoration: underline;

}

.feddbackadm{

color: #000;

width: 1200px;

margin: 10px auto;

padding-bottom: 25px;

font-size: 16px;

}

.feddbackadm p{

color: #000;

}

.feddbackadm a{

color: #000;

}

.feedback{

margin-bottom: 10px;

border: 1px #d5d5d5 solid;

border-radius: 5px;

padding-left: 10px;

padding-top: 10px;

padding-right: 10px;

padding-bottom: 25px;

}

.feedback\_name{

font-weight: 600;

font-size: 16px;

margin-bottom: 0;

margin-top: 16px;

}

.feedback\_text{

font-size: 14px;

word-wrap: break-word;

margin-top: 0;

}

.feedback\_date{

font-size: 12px;

color: #acacac;

padding-top: 0;

padding-bottom: 0;

margin-top: 0;

margin-bottom: 6px;

}

.pagination{

margin-bottom: 15px;

}

.pagination a{

padding-right: 5px;

text-decoration: none;

}

.pagination select{

border-radius: 5px;

border: 2px #acacac solid;

background-color: #fff;

font-family: "Montserrat", sans-serif;

font-optical-sizing: auto;

font-weight: 400;

font-style: normal;

font-size: 14px;

}

.pagination button{

border-radius: 5px;

border: 2px #acacac solid;

background-color: #fff;

font-family: "Montserrat", sans-serif;

font-optical-sizing: auto;

font-weight: 400;

font-style: normal;

font-size: 14px;

}

.orders {

display: flex;

flex-direction: column;

}

.order {

border: 1px solid #ddd;

padding: 10px;

margin-bottom: 10px;

}

.current\_p {

font-weight: bold;

}

.messages {

display: flex;

flex-direction: column;

}

.message {

border: 1px solid #ddd;

padding: 10px;

margin-bottom: 10px;

}

1. Main.css