

Функции js на странице index.html

- Обработка события нажатия на кнопку

```
document.querySelector('.btn-new').addEventListener('click', function() {
```

- Получение данных из формы

```
const form = document.getElementById('regForm');
```

- Проверка, что все поля формы заполнены

```
const inputs = form.querySelectorAll('input[required]');
```

- Инициализация переменной для проверки и цикл для прохода по полям формы

```
let allFilled = true;

inputs.forEach(input => {
  if (!input.value || !input.checkValidity()) {
    allFilled = false; // Если есть незаполненные поля});
```

- При правильном заполнении переход на страницу теста, иначе вывод сообщения об ошибке

```
if (allFilled) {
  window.location.href = 'test.html'; // Переход к тесту
}
else {
  alert('Пожалуйста, заполните все поля и проверьте корректность!');
}
});
```

Функции js на странице test.html

- Создание и управление всплывающего окна с приветствием

```
const welcomeModalText = `Привет, давай познакомимся!  
Представь, что ты пишешь небольшое сочинение на тему: "Кем мне стать?".  
Чтобы облегчить тебе задачу, предлагаю ответить на пару вопросов в  
свободной форме. Отвечай искренне, это может наилучшим образом помочь  
тебе.  
Впереди тебя ждут три блока вопросов: "прошлое", "настоящее", "будущее" и  
один блок на свободное написание.`;  
  
const welcomeModal = document.createElement('div');  
welcomeModal.className = 'welcome-modal';  
welcomeModal.setAttribute('role', 'dialog');  
welcomeModal.setAttribute('aria-modal', 'true');  
welcomeModal.setAttribute('aria-labelledby', 'welcomeTitle');  
welcomeModal.innerHTML = `  
    <button class="close-btn" aria-label="Закреть  
приветствие">&times;</button>  
    <p id="welcomeTitle" style="font-weight:bold; font-size:20px; margin-  
bottom:15px;">Привет, давай познакомимся!</p>  
    <p>${welcomeModalText.replace(/\n/g, '<br>')}</p>  
`;  
document.body.appendChild(welcomeModal);
```

- Блокировка блоков, пока приветственное окно на экране

```
window.addEventListener('load', () => {  
    welcomeModal.style.display = 'block';  
    textarea.disabled = true;  
    completeBtn.disabled = true;  
    completeBtn.classList.remove('active');  
    blockTitle.textContent = 'Блок: ...';  
});
```

- Заккрытие приветственного окошка и переход к первому блоку эссе

```
closeWelcomeBtn.addEventListener('click', () => {  
    welcomeModal.style.display = 'none';  
    currentBlockIndex = 0;  
    blockTitle.textContent = `Блок: ${blocks[currentBlockIndex]}`;  
    textarea.disabled = false;  
    textarea.value = '';  
    textarea.focus();  
});
```

- Высвечивание подсказок при нажатии на помогающую кнопку. И закрытие подсказок при нажатии на кнопку/любую область экрана

```
helpBtn.addEventListener('click', () => {
  const isVisible = tooltip.classList.toggle('visible');
  tooltip.setAttribute('aria-hidden', !isVisible);
});
document.addEventListener('click', (e) => {
  if (!tooltip.contains(e.target) && !helpBtn.contains(e.target)) {
    tooltip.classList.remove('visible');
    tooltip.setAttribute('aria-hidden', 'true');
  }
});
```

- Активация кнопки перехода к следующему блоку, если в поле ответа есть текст

```
textarea.addEventListener('input', () => {
  if (textarea.value.trim().length > 0) {
    completeBtn.disabled = false;
    completeBtn.classList.add('active');
  } else {
    completeBtn.disabled = true;
    completeBtn.classList.remove('active');
  }
});
```

- Переход между блоками и обработка кнопки на переход к другой страницы

```
const blocks = ['Прошное', 'Настоящее', 'Будущее', 'Свободное написание'];
let currentBlockIndex = 0;
completeBtn.addEventListener('click', () => {
  if (completeBtn.disabled) return;

  // Можно здесь сохранить эссе блока, например в localStorage или
  // переменной
  console.log(`Эссе блока "${blocks[currentBlockIndex]}":`,
    textarea.value.trim());

  currentBlockIndex++;
  if (currentBlockIndex < blocks.length) {
    blockTitle.textContent = `Блок: ${blocks[currentBlockIndex]}`;
    textarea.value = '';
    textarea.focus();
    completeBtn.disabled = true;
    completeBtn.classList.remove('active');
  } else {
    // Все блоки пройдены – переход на результат
    window.location.href = 'result.html';
  }
});
```

Основные функции js на странице result.html:

- Хранение данных

```
// Пример данных — в реальном проекте можно получать с сервера или из localStorage
const resultData = {
  faculty: "ИСАУ",
  directions: [
    "Информационные системы и технологии",
    "Управление в технических системах",
    "Программная инженерия",
    "Искусственный интеллект"
  ]
};
```

- Получение элементов со страницы

```
const facultyTitle = document.getElementById('facultyTitle');
const directionsList = document.getElementById('directionsList');
const exitBtn = document.getElementById('exitBtn');
const surveyBtn = document.getElementById('surveyBtn');
```

- Заполнение страницы имеющимися данными

```
function renderResult(data) {
  facultyTitle.textContent = `Факультет ${data.faculty}`;
  directionsList.innerHTML = '';
  data.directions.forEach(direction => {
    const li = document.createElement('li');
    li.textContent = direction;
    directionsList.appendChild(li);
  });
}
```

- Обработка кнопок перехода к опросу и закрытия (переход на страницу университета)

```
exitBtn.addEventListener('click', () => {
  window.location.href = 'https://uni-dubna.ru/';
});

surveyBtn.addEventListener('click', () => {
  window.location.href = 'opros.html';
});
```

Основные функции js на странице opros.html

- «База» для хранения все вопросов

```
const allQuestions = [  
  {id: 1, text: "Мне нравится руководить работой других людей и  
организовать людей для выполнения рабочих задач и мероприятий"},  
  {id: 2, text: "Мне нравится обслуживать приборы, следить за их  
работой, регулировать и настраивать"},  
  {id: 3, text: "Мне нравится изготавливать по чертежам (проектам) детали,  
изделия, здания и сооружения"},  
  {id: 4, text: "Мне нравится обучать людей, преподавать в учебных  
заведениях, сообщать, разъяснять людям нужные им сведения "},  
  {id: 5, text: "Мне нравится выводить новые сорта растений, новые  
породы животных"},  
  {id: 6, text: "Мне нравится управлять персоналом в организации:  
осуществлять отбор, помогать адаптироваться, обучать, мотивировать на  
эффективную работу "}, .....  
  {id: 35, text: "Мне нравится изобретать и производить новые химические  
соединения (лаки, краски, бытовую химию и др.)"}  
];
```

- Выбор случайных 10 вопросов из «базы»

```
const selectedQuestions = allQuestions  
  .sort(() => Math.random() - 0.5)  
  .slice(0, 10);
```

- Отображение выбранных вопросов на страницу в форму

```
function renderQuestions() {  
  const container = document.getElementById('questions-  
container');  
  
  selectedQuestions.forEach(question => {  
    const questionHTML = `  
      <div class="question-card" data-id="${question.id}">  
        <h3>${question.text}</h3>  
        <div class="rating-scale">  
          ${Array.from({length: 5}, (_, i) =>  
            `<div class="rating-point"  
              data-value="${i + 1}"  
              onclick="setRating(this,  
${question.id})"></div>`  
          )}.join('')  
        </div>  
      </div>  
    `;  
    container.innerHTML += questionHTML;  
  });  
}
```

- Обработка выбора ответа и закрашивание кружочков

```
function setRating(element, questionId) {
    const value = parseInt(element.dataset.value);
    const points = element.parentElement.children;
    Array.from(points).slice(0, value).forEach(point =>
        point.classList.add('active'));
}
```

- Проверка ответов на все вопросы и обработка кнопки перехода к следующей странице

```
function validateForm() {
    const answeredQuestions = Object.keys(answers).length;
    const allFilled = answeredQuestions ===
selectedQuestions.length;

    if (allFilled) {
        window.location.href = 'end.html';
    } else {
        alert('Пожалуйста, ответьте на все вопросы!');
    }
}
```

- Высвечивание подсказок при нажатии на помогающую кнопку. И закрытие подсказок при нажатии на кнопку/любую область экрана

```
document.querySelector('.help-btn').addEventListener('click', (e) => {
    document.querySelector('.help-popup').style.display =
        document.querySelector('.help-popup').style.display ===
'block' ? 'none' : 'block';
});
document.addEventListener('click', (e) => {
    if (!e.target.closest('.help-btn') &&
        !e.target.closest('.help-popup')) {
        document.querySelector('.help-popup').style.display =
'none';
    }
});
```

Основные функции js на странице end.html

- «База» для возможных результатов

```
const resultsData = {
  technical: {
    name: "Информатика и вычислительная техника",
    description: "Идеально подходит для работы с технологиями и инновациями. Рекомендуемые области: программирование, инженерия, робототехника.",
    recommendations: [
      "Курс по машинному обучению",
      "Программирование на Python",
      "Основы кибербезопасности",
      "Системное администрирование"
    ]
  },
  humanitarian: {
    name: "Клиническая психология",
    description: "Подходит для работы с людьми и социальными системами. Перспективные области: психология, социология, международные отношения.",
    recommendations: [
      "Курс по межкультурной коммуникации",
      "Основы психологии личности",
      "Публичные выступления",
      "Социальная антропология"
    ]
  },
  creative: {
    name: "Прикладная информатика",
    description: "Идеально для реализации в искусстве и креативных индустриях. Перспективные направления: дизайн, медиа, современное искусство.",
    recommendations: [
      "Графический дизайн для начинающих",
      "Цифровая иллюстрация",
      "Фотография и композиция",
      "Сценарное мастерство"
    ]
  }
};
```

- Логика выбора результата

```
function getResult() {
  const types = Object.keys(resultsData);
  const randomType = types[Math.floor(Math.random() * types.length)];
  return resultsData[randomType];
}
```

- Добавление выбранного результата на страницу

```
function renderResults() {
    const result = getResult();
    document.getElementById('directionName').textContent =
result.name;
    document.getElementById('directionDescription').textContent =
result.description;

    const recommendationsList =
document.getElementById('recommendationsList');
    recommendationsList.innerHTML = result.recommendations
        .map(item => `<li>${item}</li>`)
        .join('');
}
```

- Обработка кнопки перехода на страницу университета

```
function redirectToUniversity() {
    window.location.href = "https://uni-dubna.ru/";
}
```