**Ответы на вопросы**

1. **Для чего служат формы на веб-страницах?**

Для получения данных от пользователя

1. **Каким типом задаётся поле ввода в форме текста?**

Type =’text’

1. **Какими способами можно обратиться к элементу формы?**

Через свойство elements:   
document.forms.name формы.elements

1. **Что такое событие submit для формы и как его можно использовать?**

Событие, которое отправляет данные на сервер и перезагружает страницу

1. **Восстановить значения умолчания в полях формы можно только по кнопке reset?**

Нет, существуют несколько способов восстановить значения полей формы к значениям по умолчанию.

1. С помощью кнопки "Reset": в HTML форма может содержать элемент кнопки со значением "Reset". При нажатии на эту кнопку, все значения полей формы будут сброшены к значениям по умолчанию.

<form id="myForm">

<input type="text" name="name" value="Default Name">

<input type="reset" value="Reset">

</form>

2. С помощью JavaScript: с использованием JavaScript можно изменять значения полей формы вручную. Например, можно использовать метод `defaultValue` для установки значения поля по умолчанию.

<form id="myForm">

<input type="text" name="name" value="Default Name">

<input type="button" value="Reset" onclick="resetForm()">

</form>

<script>

function resetForm() {

document.getElementById("myForm").reset();

document.getElementById("myForm").name.defaultValue = "Default Name";

}

</script>

3. Используя JavaScript и HTML5: HTML5 вводит атрибут `placeholder`, который может быть использован для определения текста-подсказки, отображаемого в поле формы, пока оно не заполнено значением. Если атрибут `placeholder` указан для поля формы, можно очистить значение поля, введя `""` или значение атрибута `placeholder` с помощью JavaScript.

<form id="myForm">

<input type="text" name="name" placeholder="Default Name">

<input type="button" value="Reset" onclick="resetForm()">

</form>

<script>

function resetForm() {

document.getElementById("myForm").reset();

document.getElementById("myForm").name.value = "";

}

</script>

1. **Каким способом можно отправить данные из формы на сервер?**

Указать доступ к серверу в теге form и указать method = ‘post’:

<form name="form" action="https://server.com/form" method="POST">

1. **Как можно полностью перехватить и обработать js-функцией событие submit (отправку данных на сервер)?**

Отменить стандартное событие:

form.addEventListener('submit', function (evt) { evt.preventDefault();})

1. **Какие есть недостатки у стандартного способа задания валидации через HTML5?**

Он срабатывает еще до того, как пользователь ввел данные в поля формы. Еще один минус этого способа — его ограниченность. Проверять мы можем только факт заполнения, но не корректность введённых данных.

1. **Какие свойства доступны в объекте validity для проверки валидности элемента формы?**

valid: false //Поле не является валидным

customError: false //Не установлено специальное сообщение об ошибке

patternMismatch: false //Значение не удовлетворяет шаблону, указанному в атрибуте pattern

rangeOverflow: false //Значение не превышает максимальное значение, указанное в атрибуте max

rangeUnderflow: false //Значение должно быть больше или равно минимальному значению, указанному в атрибуте min

stepMismatch: false //Значение соответствует указанному шагу

tooLong: false //Значение не слишком длинное

tooShort: false //Значение не слишком короткое

typeMismatch: false //Значение соответствует указанному атрибуту type

valueMissing: false //Не отсутствует обязательное значение

1. **Как можно проверить, является ли элемент формы валидным с использованием объекта validity?**

Вызывать в функции и проверять верность введения. Например:  
if (**validity.patternMismatch**) {errors.push('Неверный формат заполнения'); }

1. **Какие флаги могут быть использованы при создании регулярных выражений и какова их роль?**

Основные флаги:

i - флаг, указывающий на регистронезависимый поиск: нет разницы между A и a

g - флаг, указывающий на глобальный поиск всех совпадений, без него будет найдено только первое совпадение

y - режим поиска на конкретной позиции в тексте

m - многострочный режим

1. **Что делает метод match() при работе с регулярными выражениями? Какие аргументы он принимает?**

Строка.match(регулярное выражение) – этот метод возвращает массив совпадений или null, если совпадения не найдены

1. **Что делает метод test() при проверке регулярного выражения на совпадение? Какие значения он возвращает?**

Регулярное выражение.test(строка) – метод возвращает true, если есть хотя бы одно совпадение с регулярным выражением в строке, или false, если совпадений не найдено