

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор высшей школы
программной инженерии
А.В. Малеев**

	Рабочая программа дисциплины (модуля)
по дисциплине:	Веб-верстка и язык Javascript
по направлению:	Программная инженерия
профиль подготовки:	Разработка программно-информационных систем высшая школа программной инженерии высшая школа программной инженерии МФТИ - Яндекс
курс:	1
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 2 (весенний) - Дифференцированный зачет

Аудиторных часов: 45 всего, в том числе:

лекции: 0 час.

семинары: 45 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 27 час.

Всего часов: 72, всего зач. ед.: 2

Программу составил: А.В. Малеев, заведующий кафедрой

Программа обсуждена на заседании высшей школы программной инженерии МФТИ - Яндекс 06.06.2022

Аннотация

В курсе изучаются основы и тонкости языка программирования JavaScript, а также особенности его применения. Для освоения дисциплины требуются базовые знания принципов программирования, уделяется внимание разработке серверной части веб-приложений, организации и оптимизации вёрстки, организации окружения.

1. Цели и задачи

Цель дисциплины

- получение практических знаний и прикладных навыков WEB-разработки с использованием функционала языка JavaScript.

Задачи дисциплины

- обучить студентов навыкам программирования на языке JavaScript, познакомить студентов с его основными функциями и возможностями;
- познакомить студентов с особенностями WEB-вёрстки с использованием инструментария языка JavaScript и обучить современным методикам и инструментам WEB-вёрстки.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.1 Знает алгоритмы решения типовых задач, области и способы их применения
	ОПК-6.2 Умеет применять языки программирования для решения прикладных задач
	ОПК-6.4 Имеет навыки программирования и тестирования программных продуктов
	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
ОПК-7 Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1 Обладает навыками создания и выполнения тестовых сценариев для выявления ошибок в программном обеспечении
	ОПК-7.2 Понимает принципы работы баз данных и умеет проектировать структуру данных для эффективного хранения информации
	ОПК-7.3 Умеет выявлять узкие места в процессе разработки и предлагать методы и инструменты для его оптимизации
ПК-3 Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение	ПК-3.1 Различает синтаксис языков программирования, особенности программирования на этих языках, стандартные библиотеки языков программирования
	ПК-3.4 Знает, как определять оптимальные методы и средства проектирования программного обеспечения и структур данных

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны знать:

- основы работы с языком JavaScript и его основной инструментарий;
- определения: встроенные типы данных, базовое определение функции, прототипы, фреймворк.

уметь:

- использовать функционал языков JavaScript и HTML для задач фронтенд- и бэкенд-разработки;
- использовать подходы объектно-ориентированного программирования для предотвращения возникновения проблем при масштабировании программ;
- находить требуемый элемент в дереве DOM;
- управлять отображением данных при помощи CSS;
- использовать встроенные в браузер механики HTML элементов.

владеть:

- основными инструментами, функциями и методиками веб-разработки с использованием языков JavaScript и HTML.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Основы работы с языком JavaScript		2		1
2	Типы данных в языке JavaScript		3		1
3	Функции, области видимости и замыкания		10		5
4	Объектно-ориентированное и функциональное программирование в JavaScript		10		5
5	Введение в Html		6		5
6	Фреймворки для веб-программирования		10		5
7	Публичные интерфейсы браузера		4		5
Итого часов			45		27
Подготовка к экзамену		0 час.			
Общая трудоёмкость		72 час., 2 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 2 (Весенний)

1. Основы работы с языком JavaScript

Назначение языка JavaScript, обзор переменных, операторов, выражения и инструкции, управляющие структуры и встроенные функции, циклы while, for, do while, if – else.

2. Типы данных в языке JavaScript

Встроенные типы данных, Boolean, создание структур из примитивов, передача данных по значению и ссылке, оператор typeof.

3. Функции, области видимости и замыкания

Базовое определение функции, замыкания и выделения памяти, декларация функций, аргументы функции, области видимости, функция-выражение, анонимная функция.

4. Объектно-ориентированное и функциональное программирование в JavaScript

Конструкторы объектов, прототипы, классы, инкапсуляция, наследование и полиморфизм в JavaScript.

5. Введение в Html

Функционал, использование, элементы форматирования текста, атрибуты, разметка, синтаксис и семантика.

6. Фреймворки для веб-программирования

Фреймворки CSS Препроцессоры CSS: LESS и SASS, Фреймворки Javascript Библиотека jQuery, Фреймворк AngularJS. Фреймворк ReactJS.

7. Публичные интерфейсы браузера

Управление деревом представления DOM. Модель событий, селекторы HTML элементов.

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Компьютер, мультимедийное оборудование (проектор).

6.Перечень рекомендуемой литературы

Основная литература

1. Технологии разработки программного обеспечения, Электрон. версия печ. публикации / . — Санкт-Петербург, Питер, 2012
2. Разработка приложений Java EE 6 в NetBeans 7 [Электронный ресурс], Электрон. версия печ. публикации / Д. Хеффельфингер ; пер. с англ. Е. Н. Карышева. — М., ДМК Пресс, 2013

Дополнительная литература

Рекомендуемая литература для самостоятельного изучения:

Немцова, Т. И. Компьютерная графика и web-дизайн : учеб. пособие / Т.И. Немцова, Т.В. Казанкова, А.В. Шнякин / под ред. Л.Г. Гагариной. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2017. — 400 с.

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Не требуются

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

не предусмотрено.

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Программа курса в разделе «самостоятельная работа» обозначает минимально необходимое время для работы студента над темой. Самостоятельная работа включает в себя:

- проработку учебного материала (по учебной и научной литературе), подготовку ответов на вопросы, предназначенных для самостоятельного изучения, доказательство отдельных утверждений, свойств;
- выполнение домашних заданий.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Программная инженерия
профиль подготовки:	Разработка программно-информационных систем высшая школа программной инженерии МФТИ - Яндекс высшая школа программной инженерии
курс:	1
квалификация:	бакалавр
Семестр, формы промежуточной аттестации: 2 (весенний) - Дифференцированный зачет	
Разработчик:	А.В. Малеев, заведующий кафедрой

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.1 Знает алгоритмы решения типовых задач, области и способы их применения
	ОПК-6.2 Умеет применять языки программирования для решения прикладных задач
	ОПК-6.4 Имеет навыки программирования и тестирования программных продуктов
	ОПК-6.3 Знает методы тестирования программного кода на ошибки и способен проводить тестирование на различных уровнях (модульное, интеграционное, системное)
ОПК-7 Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1 Обладает навыками создания и выполнения тестовых сценариев для выявления ошибок в программном обеспечении
	ОПК-7.2 Понимает принципы работы баз данных и умеет проектировать структуру данных для эффективного хранения информации
	ОПК-7.3 Умеет выявлять узкие места в процессе разработки и предлагать методы и инструменты для его оптимизации
ПК-3 Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение	ПК-3.1 Различает синтаксис языков программирования, особенности программирования на этих языках, стандартные библиотеки языков программирования
	ПК-3.4 Знает, как определять оптимальные методы и средства проектирования программного обеспечения и структур данных

2. Показатели оценивания компетенций

В результате изучения дисциплины «Веб-верстка и язык Javascript» обучающийся должен:

знать:

- основы работы с языком JavaScript и его основной инструментарий;
- определения: встроенные типы данных, базовое определение функции, прототипы, фреймворк.

уметь:

- использовать функционал языков JavaScript и HTML для задач фронтенд- и бэкенд-разработки;
- использовать подходы объектно-ориентированного программирования для предотвращения возникновения проблем при масштабировании программ;
- находить требуемый элемент в дереве DOM;
- управлять отображением данных при помощи CSS;
- использовать встроенные в браузер механики HTML элементов.

владеть:

- основными инструментами, функциями и методиками веб-разработки с использованием языков JavaScript и HTML.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

С целью контроля освоения обучающимися учебного материала проводится устный опрос в начале занятия по теме прошлого занятия.

Примерные задания для проведения мини-контрольных в рамках учебных занятий

1) Опрос по теории языка HTML5:

1. С помощью какого тега загрузятся стили для принтера?

Варианты ответов:

- `<link href="styles.css" rel="print">`
- `<link href="styles.css" rel="stylesheet" type="text/css">`
- `<link href="styles.css" rel="stylesheet" target="_print">`
- `<link href="styles.css" media="print" rel="stylesheet">` (верный)

2. Для чего применяется тег `<base href="http://ya.ru/pages">`?

Варианты ответов:

- Он определяет базовый адрес для всех ссылок документа (верный)
- Для задания адреса текущей страницы
- В целях оптимизации сайта для поисковых роботов
- Для задания адреса, с которого пользователь пришел на сайт

3. Где можно расположить тег `<tbody>`?

Варианты ответов:

- Где угодно в таблице
- Непосредственно внутри `<table>` (верный)
- Непосредственно внутри `<table>`, но строго после `<thead>` и до `<tfoot>` (в случае их наличия)

4. Укажите все теги, предназначенные для вывода частей программного кода.

Варианты ответов:

- `<samp>`
- `<output>`
- `<source>`
- `<code>` (верный)
- `<kbd>`

5. Чем примечателен атрибут `title`?

Варианты ответов:

- Добавляет подпись под картинкой/таблицей
- Добавляет окно подсказки при наведении на элемент (верный)
- Устанавливает заголовок страницы

6. Какие из перечисленных элементов НЕ входят в спецификацию HTML?

Варианты ответов:

17

- `<section>`, `<nav>`, `<article>`
- `<hgroup>`, `<address>`, `<main>`
- `<spinner>`, `<foot>`, `<dropdown>` (верный)
- `<header>`, `<footer>`, `<aside>`

7. Что произойдет, если указать URL, начинающийся с `"file:"`?

Варианты ответов:

- Браузер попытается открыть файл на вашем компьютере локально (верный)
- Браузер попытается сохранить страницу в указанный файл
- Браузер загрузит файл на сервер
- Эта запись браузеру не известна, он ничего не сделает

8. Что из перечисленного встраивается в `iframe`?

Варианты ответов:

- Java-апплеты или flash-контент (верный)
- Окна для расширения в социальных сетях (верный)
- Аудио- и видеофайлы (верный)

9. Можно ли использовать `schema.org` и микроформаты в одном документе?

Варианты ответов:

- Можно без ограничений (верный)

- Можно, но с ограничениями, т.к. некоторые используемые атрибуты совпадают
- Можно, но не рекомендуется, т.к. поисковые системы считают это спамом
- Нельзя, эти форматы не совместимы

10. Что будет, если мы не укажем размеры картинки с помощью width и height?

Варианты ответов:

- До загрузки картинка будет иметь минимальный размер, а после загрузки вставится на страницу, но не превысит ширины экрана
- До загрузки картинка будет иметь минимальный размер, а после загрузки вставится на страницу, но не превысит высоты экрана
- До загрузки картинка будет иметь исходный размер и белый фон, а после загрузки изображение отобразится в выделенном под нее пространстве
- До загрузки картинка будет иметь минимальный размер, а после загрузки вставится на страницу в исходном размере (верный)

2) Опрос по теории языка CSS.

1. На странице есть ссылки:

`yandex.ru`

`ya.ru`

`ссылка`

Какой селектор окрасит в синий цвет ссылку, адрес которой содержит слово «yandex»?

Варианты ответов:

- `[href="yandex"] { color: blue }`
- `{href+="yandex"} { color: blue }`
- `[href*="yandex"] { color: blue }` (верный)
- `(href?="yandex") { color: blue }`

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Назвать все известные конструкторы объектов.
2. Какие существуют прототипы и классы.
3. Какие есть механизмы наследования и полиморфизм в JavaScript
4. Базовое определение функции, декларации функций, аргументов функции.
5. Что такое области видимости, функция-выражение, анонимная функция.
6. Какие бывают переменные, операторы, циклы.
7. Назвать варианты элементов форматирования текста, атрибуты, разметка, синтаксис и семантика на HTML.

Критерии оценивания

- оценка «отлично (10)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений
- оценка «отлично (9)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений
- оценка «отлично (8)» выставляется студенту, показавшему всесторонние систематизированные, глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, и правильное обоснование принятых решений
- оценка «хорошо (7)» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «хорошо (6)» выставляется студенту, если он знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;

- оценка «хорошо (5)» выставляется студенту, если он знает материал, и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;
- оценка «удовлетворительно (4)» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;
- оценка «удовлетворительно (3)» выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет фрагментарно основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;
- оценка «неудовлетворительно (2)» выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач
- оценка «неудовлетворительно (1)» выставляется студенту, который не знает формулировок основных понятий дисциплины

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Дифференцированный зачет проводится по итогам текущей успеваемости и сдачи заданий и других видов работ, предусмотренных программой дисциплины и (или) путем организации специального опроса, проводимого в устной и (или) письменной форме.