# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

HTML5+CSS3: Table element

Проект: Закрытие страниц сайта для незалогиненного пользователя

Задача по Монго: #3

1) <tr> - создает строку (вправо), <td> - создает столбец (вниз), <th> - заголовок столбца

<https://www.w3schools.com/html/html_tables.asp>

2) Пункт 10.8 пособия

3) Найти единорогов без статистики убитых вампиров:

**db.unicorns.find({vampires:{$exists:false}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 2

1. HTML5+CSS3: sup, sub, strong, em elements

2. Проект: Функциональность logout

3. Задача по Монго: #4

1) <sup> - superscript – текст НАД обычным текстом (102)

<sub> - subscript – текст ПОД обычным текстом (H2O)

<strong> - работает как <b>, то есть делает жирный шрифт, но используется для важного текста – когда экранный диктор читает текст и натыкается на strong текст он выделяет его интонационно

<em> - делает шрифт курсивом, но используется для важного текста.

2) Пункт 10.7 пособия

3) Найти белых единорогов которые весят меньше 500:

**db.unicorns.find({color:"white",weight:{$lt:500}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 3

1. HTML5+CSS3: Unordered, Ordered, Nested списки

2. Проект: Обработка ошибки аутентификации

3. Задача по Монго: #5

1) <ul> - Unordered List – неупорядоченный список, пункт добавляется через <li> внутри тега

<ol> - Ordered List – упорядоченный список, пункт добавляется через <li> внутри тега

Nested List – Вложенный список, добавляется через <ul> или <ol> после <li>

Чтоб убрать цифры или точки перед каждым пунктом в CSS нужно ввести следующее:

**list-style-type: none;**

2) Пункт 10.6 пособия

3)Найти сколько единорогов не весят 600:

db.unicorns.find({weight:{$ne:600}}).count()

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 4

1. HTML5+CSS3: Внешние, внутренние ссылки, ссылки внутри экрана

2. Проект: Глобальная переменная user

3. Задача по Монго: #6

1) <a> - создает ссылку, href – добавляет значение на которое будет переходить ссылка, target="\_blank" будет переходить на ссылку в новой вкладке

Внешняя ссылка – ссылка на сайт, например «https://github.com»

Внутренняя ссылка – ссылка на смежную страницу, например «index.html»

Ссылка внутри экрана – должна отсылаться на id, например если у вас есть элемент с id="test", то href в ссылке должен выглядеть следующим образом href="#test"

2) Пункт 10.5 пособия

3) Найти черных единорогов которые любят арбузы или весят больше 900:

**db.unicorns.find({color:"black",$or:[{loves:"watermelon"},{weight:{$gt:900}}]})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 5

1. HTML5+CSS3: Форма - input, submit elements

2. Проект: Логика пользователя

3. Задача по Монго: #7

1) <input type="submit"> используется внутри <form action="/action\_page">, при нажатии кнопки информация отправляется в action\_page

2) Пункт 10.4 пособия

3) Найти сколько единорогов любят яблоки или морковку:

**db.unicorns.find({$or:[{loves:'apple'},{loves:"carrot"}]}).count()**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 6

1. HTML5+CSS3: Форма - textarea, radio, checkbox elements

2. Проект: Подготовка данных для передачи на сервер

3. Задача по Монго: #8

1) <textarea> - добавляет поле в котором можно написать любой текст

<input type=”radio”> - добавляет выбираемое поле, radio input может быть несколько, но можно выбрать лишь один radio input одного типа. Тип задается через value

<input type=”checkbox”> - добавляет выбираемое поле, checkbox input может быть несколько, выбрать можно несколько checkbox radio одного типа. Тип задается через value

2) Пункт 10.3 пособия

3) Найти сколько единорогов не любят яблоки и не любят морковку:

**db.unicorns.find({$nor:[{loves:'apple'},{loves:"carrot"}]})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 7

1. HTML5+CSS3: inline, internal, external css

2. Проект: Создание модели User

3. Задача по Монго: #9

1) inline css – стили которые задаются внутри html именно для того элемента внутри которого оно пишется. <p style=”color: green”>something</p>

internal css – стили которые задаются внутри html, но работает как отдельный файл. Пишется так же как и css, но в тэге head в тэге style.

External css – стили, которые задаются в отдельном файле с расширением .css. Подключается в тэге head: <link rel="stylesheet" href="относительный путь к CSS-файлу">

2) Пункт 10.2 пособия

3) Найти единорогов, которые любят виноград или морковку:

**db.unicorns.find({$or:[{loves:”grape”},{loves:”carrot”}]})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 8

1. HTML5+CSS3: element, class, id selectors

2. Проект: Создание страницы регистрации

3. Задача по Монго: #9

1) element selector – затрагивает все тэги которые будут заданы. p {color: green;} окрасит все тэги p в зеленый цвет

Class selector – затрагивает все тэги с заданным классом. <p class=”test”> - HTML. .test {color: green;} – окрасит все элементы с классом test в зеленый цвет. ID работает так же, как и class, но задаётся как <p id=”test”>, а в CSS выбирается как #test {color: green;}. ID уникален.

2) Пункт 10.1 пособия

3) Найти единорогов, которые любят виноград или морковку:

**db.unicorns.find({$or:[{loves:”grape”},{loves:”carrot”}]})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 9

1. HTML5+CSS3: name, rgb, rgba, hex colors

2. Проект: Глобальная переменная для навигации

3. Задача по Монго: #10

1) Цвет задаётся через “color:”, name – вместо значений пишется название цвета, например green, gray, black и т.д.

RGB – задаётся через 3 значения - Red-Green-Blue. Например, color: rgb(255, 255, 255) – белый цвет.

RGBA – задаётся через 4 значения – Red-Green-Blue-Opacity. Opacity – прозрачность. Например, color: rgba(0, 0, 0, 0) – черный цвет, но т.к. opacity = 0 то он не будет показываться.

Hex colors – задаётся через 3 или 6 значные значения цвета. Например, color: #ff0000 либо color: #f00. Оба значения являются одинаковыми.

2) Пункт 9.5 пособия

3) Найти единорогов которые любят яблоки или убили вампиров не меньше 63:

**db.unicorns.find({$or:[{loves:"apple"},{vampires:{$gte:63}}]})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 10

1. HTML5+CSS3: percent, em, rem, VH, VW values

2. Проект: Создание счетчика поснщения страниц сайта

3. Задача по Монго: #11

1) percent – задаёт длину в процентах от родителя, то есть если родитель – сам экран, то 50% займет половину экрана, а если родитель какой-то квадрат, то 50% займет половину этого блока.

em – отношение к стандартному размеру шрифта элемента, 2em – в два раза больше стандарта

rem – отношение к стандартному размеру шрифта страницы, 2em – в два раза больше стандарта страницы (не отношение к родителю)

vh – viewport height – viewport это размер окна браузера. vh – 1% высоты от размера окна браузера.

vw – viewport width - viewport это размер окна браузера. vh – 1% ширины от размера окна браузера.

2) Пункт 9.4 пособия

3) Единорог Roodles похудел на 10:

**db.unicorns.update({name:"Roodles"},{$inc:{weight:-10}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 11

1. HTML5+CSS3: font-family, font-weight, font-style

2. Проект: Создание session

3. Задача по Монго: #12

1) font-family устанавливает шрифт для элемента, значений может быть несколько, если первое значение нет возможности установить то будет произведена попытка установить 2 шрифт, и т.д.

font-weight – задаёт толщину шрифта

font-style – задаёт наклонённость шрифта. Есть 3 значения: normal – обычное, italic – наклонённое, oblique -также наклонённое. Отличие italic от oblique в том, что italic должен предоставить разработчик шрифта и без этого условия оно не будет работать, а oblique просто наклоняет обычный шрифт.

2) Пункт 9.1 пособия

3) Единорог Pilot убил еще  2 вампиров:

**db.unicorns.update({name:"Pilot"},{$inc:{vampires:2}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 12

1. HTML5+CSS3: padding, margin, border properties

2. Проект: Cookie, записи в cookie

3. Задача по Монго: #13

1) padding – добавляет отступ с любой из 4 сторон. Отступ происходит от границ родителя, т.е. если элемент находится в блоке, то padding добавит отступ от него.

margin – добавляет отступ. Отступ происходит от других элементов, а не от родителя.

border-property – добавляет какой вид границы стоит показывать:

dotted, dashed, solid, double, groove, ridge, inset, outset, none, hidden. Значений может быть несколько.

2) Пункт 9.2 пособия

3) Единорог Aurora полюбил сахар”

**db.unicorns.update({name:"Aurora"},{$push:{loves:"sugar"}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 13

1. HTML5+CSS3: box-sizing border-box

2. Проект: Заполнение меню из базы

3. Задача по Монго: #14

1) Свойства width и height включают в себя значения полей и границ, но не отступов

2) Пункт 8.5 пособия

3) Добавить информацию, что Aurora привита (vacсinated: true):

**db.unicorns.update({name:"Aurora"},{$set:{vaccinated:true}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 14

1. HTML5+CSS3: background images

2. Проект: Обработка параметра в адресе, извлечение данных их базы

3. Задача по Монго: #15

1) background-image – добавляет задний фон в виде картинки. Картинка может быть как сохранена локально на компьютере, так и взята напрямую из интернета благодаря ссылке. Значений у background-image может быть несколько.

2) Пункты 8.2-8.3 пособия

3) Проверить какие единороги привиты:

**db.unicorns.find({vaccinated:true})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 15

1. HTML5+CSS3: background-repeat, background-size, background-position, background-attachment

2. Проект: Посев данных приложения

3. Задача по Монго: #15

1) background-repeat – задаёт повтор заднего фона, повтор может быть горизонтальный, вертикальный, оба и пр.

background-size – задаёт размера заднего фона, значения могут быть либо auto, то есть браузер автоматически выставит размер, либо пользователь сам может указать размеры. Порядок следующий: ширина, высота. Значениями также могут являться ключевые слова contain и cover. Contain изменит размер таким образом, что картинка будет полностью видна, т.е. не обрезана. Cover изменит размер таким образом, что картинка покроет родительский контейнер полностью.

background-position – задаёт начальное положение заднего фона, значения могут быть следующие: проценты (0% 0% это левое верхнее положение, если задано только одно значение, то другое равно 50%), x;y (0 0 это левое верхнее положение), left top, left center, left bottom, right top, right center, right bottom, center top, center center, center bottom

background-attachment – задаёт будет ли задний фон прокручиваться вместе с остальной страницей либо же будет закреплено. Значения: scroll (прокручивается – дефолт), fixed (закреплено), local (прокручивается с содержанием элемента)

2) Пункт 6.3 пособия

3) Проверить какие единороги привиты:

**db.unicorns.find({vaccinated:true})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 16

1. HTML5+CSS3: static, relative, absolute, fixed positions

2. Проект: Закрытие страниц сайта для незалогиненного пользователя

3. Задача по Монго: #14

1) static – стандартное значение. На него не действует значение right, left, top, bottom

relative – значение относительное к static. Можно затронуть через right, left, top, bottom.

fixed – значение относительное к viewport (окно браузера), т.е. всегда остаётся в одном и том же месте даже если пользователь скроллит страницу.

absolute - значение относительное к родителю (parent), если родителя нет, то работает как fixed.

sticky – значение которое работает как relative и fixed. Пока пользователь не проскроллит до определенной точки оно работает как relative, после этой точки оно «прилипает» и работает как fixed.

2) Пункт 10.8 пособия

3) Добавить информацию, что Aurora привита (vacсinated: true):

**db.unicorns.update({name:"Aurora"},{$set:{vaccinated:true}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 17

1. HTML5+CSS3: media queries

2. Проект: Функциональность logout

3. Задача по Монго: #17

1) media queries – использует @media для того чтоб изменять стили только если какое-то условие выполняется. К примеру @media only screen and (max-width: 600px) будет работать только на устройствах где ширина экрана не более 600 пикселей, т.е. на мобильных устройствах.

2) Пункт 10.7 пособия

3) Вес единорога Aurora увеличился в два раза:

**db.unicorns.update({name:"Aurora"},{$mul:{weight: 2}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 18

1. HTML5+CSS3: pseudo-class selectors

2. Проект: Обработка ошибки аутентификации

3. Задача по Монго: #16

1) Псевдо-класс используется чтоб определить особое состояние элемента, например когда на него наводятся или нажимают на него (:hover один из псевдо-классов).

Все pseudo-class - https://www.w3schools.com/css/css\_pseudo\_classes.asp

2) Пункт 10.6 пособия

3) Единорог Dunx разлюбил виноград:

**db.unicorns.update({name:"Dunx"},{$pull:{loves:"grape"}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 19

1. HTML5+CSS3: transform trasition(), scale(), rotate(), skew()

2. Проект: Глобальная переменная user

3. Задача по Монго: #11

1) transform – позволяет повернуть, увеличить, подвинуть и перекосить элемент

transition - позволяет контролировать скорость анимации при изменении свойств CSS. Вместо того, чтобы изменения свойства вступали в силу немедленно, можно сделать так, чтобы изменения свойства происходили в течение определенного периода времени.

scale – позволяет изменить размер элемента.

rotate – позволяет повернуть элемент.

skew – позволяет исказить элемент, например растянуть по диагонали и т.д.

2) Пункт 10.5 пособия

3) Единорог Roodles похудел на 10:

**db.unicorns.update({name:"Roodles"},{$inc:{weight:-10}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 20

1. HTML5+CSS3: animation

2. Проект: Логика пользователя

3. Задача по Монго: #14

1) CSS animation позволяет анимировать элементы без использования JavaScript.

Чтобы создать анимацию в CSS нужно стилизовать элемент, который хотите анимировать, с помощью свойства анимации или его подсвойств. Это позволяет настроить время, продолжительность и другие сведения о том, как должна выглядеть анимация. Это не настраивает фактический внешний вид анимации, это делается с помощью @keyframe. Подсвойства:

animation-delay - определяет задержку между загрузкой элемента и началом анимации.

animation-direction – определяет, должна ли первая итерация анимации идти в обратную сторону, и должны ли последующие итерации менять направление при каждом выполнении последовательности или сбрасываться в начальную точку и повторяться.

animation-duration – задает время, за которое анимация завершает один цикл

animation-fill-mode – определяет, как анимация применяет стили к своей цели до и после запуска.

animation-iteration-count – указывает, сколько раз должна повторяться анимация.

animation-name – указывает имя для @keyframes

animation-play-state – указывает, следует ли приостановить или воспроизвести анимацию.

animation-timing-function – определяет, как анимация переходит через ключевые кадры, путем создания кривых ускорения.

2) Пункт 10.4 пособия

3) Добавить информацию, что Aurora привита (vacсinated: true):

**db.unicorns.update({name:"Aurora"},{$set:{vaccinated:true}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 21

1. HTML5+CSS3: css variables

2. Проект: Подготовка данных для передачи на сервер

3. Задача по Монго: #9

1) Переменные CSS — это объекты, определенные разработчиками, которые содержат определенные значения для повторного использования в документе. Они задаются с помощью обозначения пользовательских свойств (например, --main-color: black;) и доступны с помощью функции var() (например, color: var(--main-color);).

2) Пункт 10.3 пособия

3) Найти единорогов, которые любят виноград или морковку:

**db.unicorns.find({$or:[{loves:”grape”},{loves:”carrot”}]})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 22

1. HTML5+CSS3:font-awesome icons

2. Проект: Создание модели User

3. Задача по Монго: #1

1)

2) Пункт 10.2 пособия

3) Найти черных единорогов, которые весят больше 700:

**db.unicorns.find({color:"black",weight:{$gt: 700}})**

# ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 23

1. HTML5+CSS3: возможности emmet

2. Проект: Создание страницы регистрации

3. Задача по Монго: #2

1) Emmet — это бесплатное дополнение для VSCode, которое позволяет ввести несколько фраз, которые затем превращаются в полноценные фрагменты кода. К примеру, вместо того чтоб создавать каждый элемент списка по отдельности можно просто прописать «li.item$\*4» и emmet сделает все нужное автоматически.

2) Пункт 10.1 пособия

3) Найти не черных единорогов вес которых не меньше 701:

**db.unicorns.find({color:{$ne:"black"},weight:{$gte:701}})**