

@Проект#1

Библиотека за DOM манипулация,
Контрола за управление на календар

Резюме:

Първият проект от курса ще ви постави в ситуацията на разработчици, на реален продукт. Ще имате серия от задания която трябва да изпълните като се базирате на предварително дефинираните изисквания за имплементация.

Пожелаваме ви приятна и успешна работа по първият ви проект от Стажантската програма.

Библиотека за DOM манипулация

Най-лесният начин да работите с елементите в web страницата е да ползвате абстракция върху DOM дървото с която лесно можете да постигнете преизползваемост между отделните браузъри както и да спестите количеството код което трябва да използвате. Някой от най-популярните решения за манипулация на DOM дървото са библиотеки като jQuery и Zepto

Да се разработи библиотека за манипулиране на DOM дървото, която да включва следните функционалности:

@функционалности

1. Вземане на съществуващ DOM елемент от страницата.
2. Добавяне на не съществуващ елемент към произволен елемент, вече съществуващ на HTML страницата.
3. Изтриване на съществуващ елемент от HTML страницата
4. Промяна на свойствата на избран елемент, в това число:
 - a. Промяна на атрибутите на елемента (id / class / data / name)
 - b. Промяна и връщане на текстово съдържание
 - c. Промяна и връщане на HTML съдържание на елемента
 - d. Промяна на съществуващи стилове, както и добавяне на множество стилове под формата на обект.
5. Контрол на траверсирането спрямо селектираният елемент в това число
 - a. Достъп до родител (parent element)
 - b. Достъп до роднина, над елемент (sibling element)
 - c. Достъп до роднина под елемент (sibling element)
 - d. Достъп до всички деца на елемента (children elements)
6. Имплементирайте събитийен модел който да ползва вградените в системата обекти за събития.

@изисквания към библиотеката

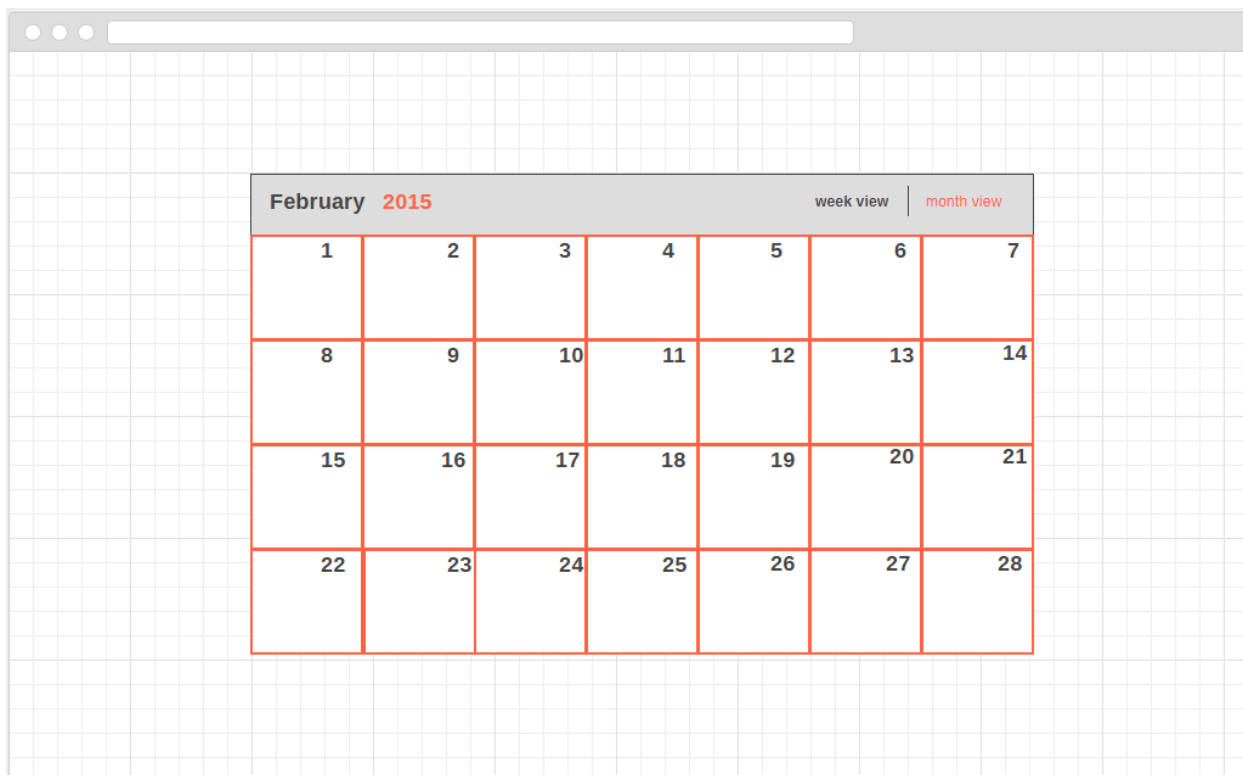
За да бъде напълно завършена библиотеката е необходимо да отговаря на няколко допълнително дефинирани изисквания за функционалност и качество

1. Методите на библиотеката трябва да поддържат възможност за верижно извикване, където това е възможно, (**method chaining**)
2. Библиотеката трябва да отговаря на **изискванията** за качествен програмен код в JS, както и да имплементира правилен Обектно ориентиран модел, + необходимите шаблони за работа с JS където това се налага.
3. Всеки един от елементите в DOM достъпен от библиотеката трябва да може да работи, с имплементираните в нея методи, без значение дали е родител, наследник, или дете на селектираният елемент.

UI контрола Календар

Използвайте DOM библиотеката за манипулация, която разработихте и ползвайте функционалностите и в имплементацията на UI контролата календар.

Контролата трябва да се визуализира на екрана, с помощта на подходящите за тази цел, DOM елементи като избира им е изцяло във ваши ръце. Можете да ползвате таблично, списъчно или блоково представяне на визията.



На фигурата е изобразена примерна визия на контролата, но всяка визуализация която отговаря на предварително дефинираните изисквания за реализация, ще бъде напълно валидна. Илюстрацията има за цел да послужи като вдъхновение, а не като спецификация за визуализацията на контролата.

@функционалности

Контролата трябва да покрива следните изисквания

1. Трябва да изобразява всички дни, от определен месец
2. Да има възможност за свободен предох между предишният и следващият месец
3. Да има възможност за преход между месеците в текущата година.
4. Модифицирайте горната операция така че да може да има свободен преход между предходната и предстоящата година, като се запази възможността за избор на месеците.
5. Трябва да показва текущият ден, от месеца, по начин отличаващ го от останалите дни.

@разширена функционалност

1. Добавете възможност дните от месеца до получават информационни обекти, описващи събития, под формата на JSON обекти, събитията трябва да се изобразяват, в полето на съответният ден, по начин определен от вас.
2. При кликване върху ден от седмицата в диалогов прозорец, намиращ се в рамките на страницата, трябва да се изобразява информация за този ден, ако има събития в него те трябва да се листват, ако няма, то трябва да се изведе подходящо съобщение.
3. Да се съставят различни изгледи към календара, освен стандартния да се поддържат изгледи за отделните дни от текущата седмици, да се предоставя възможност за преминаване между различните изгледи.

@допълнителна функционалност

1. Когато контролата е закачена към **input** поле тя трябва да се превръща, в **datepicker**, реализиращ единствено, функционалност за визуализиране на текущ месец, дните в текущият месец и следващите месеци от текущата година.
2. Когато потребителя избере дата от datepicker контролата, датата трябва да се визуализира, в полето.
3. Контролата се активира при, приемане на фокуса на полето, и се скрива при напускане на фокуса.

@изисквания към контролата

1. Всички специфични аргументи, за визуализирането на контролата се подават като опционални аргументи, при инициализирането и.
2. Задължително е контролата да се закача към произволен елемент, в страницата
3. Опциите които трябва да бъдат имплементирани, ще бъдат избрани от вас, такива опции могат да бъдат, формат на датата, начин по които се подават, масивите с данни.
4. HTML и CSS трябва да са валидни, а визуализацията потребителски издържана.