avaScript-библиотека Chart.js позволяет генерировать на стороне клиента привлекательные графики и диаграммы с использованием средств HTML5 (canvas). Библиотека поддерживает создание линейных графиков, столбцовых, круговых и радиальных диаграмм. Chart.js поддерживает использование анимированных эффектов.

Большим достоинством Chart.js является ее небольшой размер и отсутствие каких-либо зависимостей. К тому же имеется возможность ещё уменьшить размер библиотеки путем включения в нее только тех модулей, которые необходимы в конкретном случае. К примеру, если нужно создать только диаграмму кругового типа (pie chart), то можно подключить ядро библиотеки Chart.js и модуль, с помощью которого создаются диаграммы подобного типа. Таким образом, будет уменьшен общий размер библиотеки Chart.js и увеличена скорость загрузки страницы в целом. Другим достоинством библиотеки является адаптивность диаграммы, позволяющая изменять свой размер при изменении размеров окна браузера.

В отличие от библиотеки JFreeChart, которая используется только в java-приложении и может быть интерактивной только под управлением JVM, библиотека Chart.js исполняется в браузере, обеспечивая всплывающие подсказки и анимированные эффекты.

Функция randomScalingFactor генерит значения для 2-х наборов данных barChartData. По оси абсцисс равномерно откладываются метки labels. Для каждой метки в наборе данных data имеется свое значение. При желании повторить данный пример необходимо вместо многоточий вставить оставшиеся 10 вызовов функций randomScalingFactor.

Набор Dataset1 без заполнения (fill=false) окрашен в красный цвет. В наборе Dataset2 используется опция заполнения по умолчанию (fill=true) - вся область от оси абсцисс до кривой закрашивается в синий цвет. На следующем скриншоте представлено созданное библиотекой Chart.js изображение диаграммы для рассмотренного примера.

По умолчанию флаг всплывающей подсказки установлен. Поэтому при наведении курсора на одно из значений появляется изображение всплывающей подсказки с указанием наименования набора и его значения в текущей точке. Легенда по умолчанию размещается сверху и может быть перенесена в любую область диаграммы (слева, справа, внизу). Оси можно именовать.

Библиотека Chart.js имеет большое количество опций для настройки интерфейса диаграммы. На сайте описание наиболее значимых опций с примерами представлено здесь. Желающие могут перейти к полной англоязычной версии документации.

Пример использования библиотеки Chart.js для динамического формирования страницы сайта с использованием сервлета. В дистрибутив примера, выполненного в виде проекта Eclipse, включена библиотека Chart.js и статическая страница chart.html.

Теперь мы предоставим данные, а также параметры конфигурации, которые нам нужны для построения линейной диаграммы.

|  |  |
| --- | --- |
| 1 | **var** speedData = { |
| 2 | labels: ["0s", "10s", "20s", "30s", "40s", "50s", "60s"], |
| 3 | datasets: [{ |
| 4 | label: "Car Speed", |
| 5 | data: [**0**, **59**, **75**, **20**, **20**, **55**, **40**], |
| 6 | }] |
| 7 | }; |
| 8 |  |
| 9 | **var** chartOptions = { |
| 10 | legend: { |
| 11 | display: **true**, |
| 12 | position: 'top', |
| 13 | labels: { |
| 14 | boxWidth: **80**, |
| 15 | fontColor: 'black' |
| 16 | } |
| 17 | } |
| 18 | }; |