1. **Модульная сетка** — это система, которая делит пространство на равные части (модули), что позволяет организовать контент на странице более структурированно и упорядоченно. Она помогает дизайнерам и разработчикам создавать адаптивные и отзывчивые макеты, которые хорошо выглядят на различных устройствах.
2. **Система модульной верстки Skeleton** — это легковесный CSS-фреймворк, который предоставляет базовые стили и сетку для быстрого создания адаптивных веб-страниц. Skeleton предлагает простую и понятную структуру, позволяя разработчикам сосредоточиться на контенте, а не на сложной настройке стилей.
3. **Использование системы Skeleton** можно начать с подключения CSS-файла Skeleton к вашему проекту. Затем вы можете использовать классы, предоставляемые Skeleton, для создания сетки, кнопок, форм и других элементов интерфейса. Skeleton позволяет быстро разрабатывать макеты, которые автоматически адаптируются к различным размерам экранов.
4. **Стили в файле skeleton.css** определяют базовые стили для различных элементов HTML, таких как заголовки, абзацы, списки, таблицы и формы. Они также включают стили для сетки, которые помогают организовать контент на странице.
5. **Этапы реализации системы модульной верстки Skeleton** включают:
   * Подключение CSS-файла Skeleton к проекту.
   * Создание HTML-структуры с использованием классов Skeleton.
   * Настройка сетки и элементов интерфейса в соответствии с требованиями проекта.
   * Тестирование адаптивности на различных устройствах.
6. **Grid-верстка** — это метод компоновки элементов на веб-странице с использованием CSS Grid Layout. Она позволяет создавать сложные макеты, определяя строки и колонки, а также управляя расположением элементов в сетке.
7. **Свойство grid-template-areas** используется для определения именованных областей в сетке. Это позволяет легко управлять расположением элементов, используя имена областей вместо числовых индексов.
8. **Свойство grid-template-columns** задает количество и ширину колонок в сетке. Оно позволяет разработчикам контролировать, как элементы будут распределены по горизонтали.
9. **Свойство grid-area** используется для указания, в какой области сетки должен находиться элемент. Оно может принимать имя области, заданное в grid-template-areas, или числовые значения для указания позиции элемента.
10. **Свойство grid-row** используется для определения, в каких строках сетки должен находиться элемент. Оно позволяет управлять вертикальным расположением элементов.
11. **Запись grid-template-columns: 20px 20px 20px;** означает, что в сетке будет три колонки, каждая шириной 20 пикселей.
12. **Запись grid-row-end: 4;** указывает, что элемент должен занимать пространство до четвертой строки сетки, то есть он будет растянут на три строки, если его начало находится в первой строке.
13. **Свойство grid-gap** (или gap) используется для задания расстояния между строками и колонками в сетке. Это позволяет создавать отступы между элементами без необходимости добавления дополнительных маргинов.
14. **Свойства align-self и justify-self** используются для управления выравниванием отдельных элементов внутри их ячеек сетки. align-self отвечает за вертикальное выравнивание, а justify за горизонтальное выравнивание.