## 第一题

HTML 和 CSS 的中文全称：

* + HTML 的全称是“超文本标记语言”（HyperText Markup Language）。
  + CSS 的全称是“层叠样式表”（Cascading Style Sheets）。
  + **HTML 的作用**：HTML 是用于构建网页结构和内容的标记语言。它定义了网页的结构，包括文本、图像、超链接等的排布方式，并且通过标记语言的形式告诉浏览器如何呈现内容。
  + **CSS 的作用**：CSS 是一种样式表语言，用于描述网页内容的表现样式和布局。它能够控制页面元素的样式、大小、颜色、间距等外观效果，从而实现网页的美化和布局的控制。

网页的常见构成元素：

* + 结构内容（HTML）：标题、段落、图像、超链接等。
  + 样式和布局（CSS）：颜色、字体、间距、布局等外观样式。
  + 交互行为（JavaScript）：网页的交互性和动态效果。

静态网页和动态网页的区别：

* + **静态网页**：指固定内容呈现，内容在服务器上预先创建好，用户访问时显示相同内容的网页。内容更新需要手动修改源代码再上传到服务器。通常使用HTML、CSS等静态技术构建。
  + **动态网页**：指根据用户请求、用户操作或其他参数生成不同内容的网页。内容可以根据用户的需求、数据库中的数据等动态生成。通常使用服务器端脚本语言（如PHP、Python、Node.js）或前端 JavaScript 框架（如React、Vue）来实现。

HTML5 标准文档的结构：

* + **<!DOCTYPE html> 声明**：定义文档类型和版本。
  + **<html> 标签**：整个 HTML 文档的根元素。
  + **<head> 标签**：包含元数据和引用的外部资源。
  + **<title> 标签**：定义网页的标题。
  + **<body> 标签**：包含页面的实际内容，如文本、图像、链接等。

这些部分共同构成了一个 HTML5 文档的结构。**<!DOCTYPE html>** 声明文档类型，**<html>** 包含整个文档内容，**<head>** 包含与文档相关的配置和引用，**<title>** 给出页面标题，而 **<body>** 则包含页面可见的实际内容。

## 第二题

**HTML标记的三种分类**：

* + 块级标签（Block-level Elements）：这些标签用于创建页面的结构，它们会以块的形式显示在页面上，独占一行或者一块空间。例如 **<div>**、**<p>**、**<h1>** 到 **<h6>** 等。
  + 行内标签（Inline Elements）：这些标签用于标记文档中的行内元素，它们不会独占一行，而是在同一行内显示。例如 **<span>**、**<a>**、**<strong>**、**<em>** 等。
  + 行内块级标签（Inline-block Elements）：这些标签在外观上类似于行内元素，但可以具备块级元素的特性。例如 **<img>**、**<input>**。

1. **常见的单标记**：
   * **<br>**：表示换行，常用于在文本中换行。
   * **<hr>**：表示水平线，常用于分割段落或内容。
   * **<img>**：用于插入图像。
2. **HTML和CSS中注释的语法格式**：
   * HTML注释的语法格式：**<!-- 这是HTML注释 -->**
   * CSS注释的语法格式：**/\* 这是CSS注释 \*/**
3. **标题标记的语法格式**：
   * 语法格式：**<h1>标题内容</h1>** 到 **<h6>标题内容</h6>**
   * 特点：标题标签用于定义标题，数字表示标题级别，从 **<h1>**（最高级别）到 **<h6>**（最低级别），依次递减。通常用于页面的结构化和内容组织。
4. **图像标记的语法格式**：
   * 语法格式：**<img src="image.jpg" alt="描述性文本" width="宽度" height="高度">**
   * 必须属性：**src** 属性（图片路径）、**alt** 属性（图像的描述性文本）。
   * 属性作用：**src** 属性定义图像的路径，**alt** 属性提供替代文本，用于在图像无法显示时描述图像。
5. **列表标记**：
   * 有序列表（Ordered List）：**<ol>**，用于创建有序列表，使用 **<li>** 包裹每个列表项。
   * 无序列表（Unordered List）：**<ul>**，用于创建无序列表，同样使用 **<li>** 包裹每个列表项。
6. **超链接标记的语法格式**：
   * 语法格式：**<a href="URL" target="\_blank">链接文本</a>**
   * 特点：超链接用于创建页面间或页面内的链接。**href** 属性定义链接目标的URL地址，**target="\_blank"** 是在新标签页中打开链接的属性设置。

## 第三题

1. **CSS的语法由哪几个部分组成？** CSS语法主要由以下几个部分组成：
   * **选择器（Selectors）**：指定要应用样式的HTML元素。
   * **属性（Properties）**：要改变的样式属性。
   * **值（Values）**：属性所需的设置或值。
2. **CSS的三种引用方法**：
   * **内联样式表**：直接在HTML元素中使用**style**属性来定义样式。
   * **内部样式表**：使用**<style>**标签在HTML文档头部定义样式。
   * **外部样式表**：创建一个独立的**.css**文件，在HTML文档中通过**<link>**标签引用。
   * **引用方法的语法格式**：
     + **内联样式表**：**<tag style="property: value;">**
     + **内部样式表**：**<style> /\* CSS 样式代码 \*/ </style>**
     + **外部样式表**：**<link rel="stylesheet" type="text/css" href="styles.css">**
3. **CSS的基础选择器和它们的语法格式**：
   * **基础选择器**包括：
     + **元素选择器**：**element**，例如 **p**、**div**。
     + **类选择器**：**.class**，例如 **.example**。
     + **ID选择器**：**#id**，例如 **#uniqueID**。
   * **类选择器和ID选择器的区别**：
     + 类选择器用**.**开头，可以用于多个元素，可以重复使用。
     + ID选择器用**#**开头，对应HTML文档中唯一的一个元素，不可重复使用。
4. **CSS复合选择器**：
   * **复合选择器**是由多个基本选择器组合而成，包括：
     + **后代选择器**：**A B**，例如 **div p**。
     + **子选择器**：**A > B**，例如 **ul > li**。
     + **相邻兄弟选择器**：**A + B**，例如 **h2 + p**。
     + **通用兄弟选择器**：**A ~ B**，例如 **h2 ~ p**。
5. **选择器的优先级**：
   * 选择器的优先级用于确定应用于元素的样式，一般来说：
     + 内联样式的优先级最高。
     + ID选择器的优先级高于类选择器和元素选择器。
     + 类选择器的优先级高于元素选择器。
6. **CSS字体样式属性**：
   * **font-family**（字体族）、**font-size**（字体大小）、**font-weight**（字体粗细）、**font-style**（字体样式）、**line-height**（行高）等。
7. **CSS文本外观样式属性**：
   * **color**（文本颜色）、**text-align**（文本对齐方式）、**text-decoration**（文本装饰效果，如下划线）、**text-transform**（文本转换为大写或小写）等。
8. **伪类**：
   * 伪类是用于选择元素的特殊状态或位置的选择器，常见的伪类有：
     + **:hover** 表示鼠标悬停在元素上时的状态，常用于创建鼠标悬停效果。

## 第四题

**盒子模型的作用和属性**：

* + **作用**：盒子模型用于定义和布局HTML元素。每个元素都被看作是一个矩形的盒子，它包含内容、内边距、边框和外边距。
  + **属性**：盒子模型的六个属性包括：
    - **width**（宽度）
    - **height**（高度）
    - **padding**（内边距）
    - **margin**（外边距）
    - **border**（边框）
    - **content**（内容）

1. **padding和margin的参数顺序和含义**：
   * 参数顺序：上、右、下、左（顺时针方向）
   * 参数个数的含义：
     + 1个参数：设置的是四个方向上的相同数值。
     + 2个参数：第一个值代表上下，第二个值代表左右。
     + 3个参数：分别代表上、左右、下。
     + 4个参数：依次代表上、右、下、左。
2. **background属性**：
   * **background** 属性用于设置元素的背景效果，包括背景颜色、背景图片等。
   * 背景颜色属性：**background-color: #RRGGBB;**
   * 背景图片属性：**background-image: url("image.jpg");**
3. **HTML元素的类型**：
   * HTML元素类型分为块级元素、内联元素和内联块级元素。
   * **块级元素**：独占一行，例如**<div>**、**<p>**、**<h1>**等。
   * **内联元素**：不独占一行，按行内顺序显示，例如**<span>**、**<a>**、**<strong>**等。
   * **内联块级元素**：显示为块级元素的盒子但在行内显示，例如**<img>**、**<input>**。
4. **元素类型转换**：
   * 可以使用CSS的 **display** 属性来进行元素类型的转换。例如，将元素从块级元素转换为内联元素，或者相反。
5. **CSS的三大特性和优先级**：
   * **特性**：层叠性（Cascading）、继承性（Inheritance）、优先级（Specificity）。
   * **优先级顺序**：内联样式 > ID选择器 > 类选择器 > 元素选择器。

## 第五题

1. **CSS布局的三种机制**：
   * **流动布局（Flow Layout）**：元素按照其在HTML中的顺序自上而下流动排列。通过**display: block**或**display: inline**实现。
   * **浮动布局（Float Layout）**：元素通过**float**属性脱离标准流，使得其他元素可以在其周围进行布局。
   * **定位布局（Positioning Layout）**：元素相对于其在标准流中的位置进行定位，使用**position**属性。
2. **标准流**：
   * **标准流**指的是HTML元素默认的布局方式，元素按照其在HTML文档中的顺序自上而下依次排列，形成标准的文档流。
3. **浮动流的作用和属性**：
   * **作用**：浮动元素会脱离标准流，使得其他元素可以环绕其周围布局。
   * **浮动属性**：通过 **float** 属性设置元素的浮动，取值可以是 **left**、**right**。
4. **float的取值**：
   * **left**：元素向左浮动。
   * **right**：元素向右浮动。
   * **none**：默认值，元素不浮动。
   * **inherit**：继承父元素的浮动值。
5. **常用的定位模式**：
   * 绝对定位（**position: absolute**）
   * 相对定位（**position: relative**）
   * 固定定位（**position: fixed**）
6. **相对定位的语法格式和定位对象**：
   * **语法格式**：**position: relative;**
   * **定位对象**：相对自身位置进行定位，通过**top**、**right**、**bottom**、**left**属性调整位置。
7. **绝对定位的语法格式和定位对象**：
   * **语法格式**：**position: absolute;**
   * **定位对象**：相对于其最近的已定位的祖先元素进行定位，若没有已定位的祖先元素，则相对于文档进行定位。
8. **常见的布局形式**：
   * **单列布局**：内容在一列中排列。
   * **双列布局**：内容在两列中排列。
   * **响应式布局**：根据不同设备或屏幕尺寸变化而自适应调整的布局。
9. **网页初始化的常见设置**：
   * 清除内外边距：**body { margin: 0; padding: 0; }**
   * 设置文字颜色和大小：**body { color: #000; font-size: 16px; }**
   * 去掉列表样式：**ul, ol { list-style: none; }**
   * 去掉超链接下划线：**a { text-decoration: none; }**

窗体顶端