

以下是根据您提供的资料中第1章内容对 Review Questions 的回答。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
1.1	1960年代美国国防部（DoD）提出的新国家计算机网络的基本要求之一是什么？	One fundamental requirement was that the network be sufficiently robust so that even if some network nodes were lost... the network would continue to function .	2	一个基本要求是网络必须足够健壮，以便即使某些网络节点丢失.....网络仍能 继续运行 。
1.2	互联网上所有计算机连接使用什么协议？	The innovation that allows all these diverse devices to communicate with each other is a single, low-level protocol named Transmission Control Protocol/Internet Protocol (TCP/IP) .	3	允许所有这些不同设备相互通信的创新是一种名为 传输控制协议/互联网协议（TCP/IP） 的单一、低级协议。
1.3	IP 地址的形式是什么？	IP addresses usually are written (and thought of) as four 8-bit numbers, separated by periods .	4	IP 地址通常被写成（并被认为）是 四个 8 位数字，用点分隔 。
1.4	描述一个完全限定域名。	The host name and all the domain names are together called a fully qualified domain name .	5	主机名和所有域名一起被称为一个完全限定域名 。
1.5	DNS 命名服务器的任务是什么？	Name servers serve a collection of machines on the Internet and are operated by organizations... They implement the Domain Name System (DNS) .	5	命名服务器 服务于互联网上的一组机器.....它们 实现了域名系统（DNS） 。
1.6	telnet 的目的是什么？	telnet, which was developed to allow a user on one computer on the Internet to log onto and use another computer on the Internet .	5	telnet 的开发是为了让互联网上的一台计算机上的用户可以 登录并使用互联网上的另一台计算机 。
1.7	在 Web 的最初提议中，要交换的信息形式是什么？	The intent of this new system, which the group named the World Wide Web, was to allow scientists around the world to use the Internet to exchange documents describing their work .	6	这个新系统（该小组将其命名为万维网）的目的是允许世界各地的科学家使用互联网 交换描述其工作的文档 。
1.8	什么是超文本？	hypertext, which is text with embedded links to text , either in the same document or in another document, to allow nonsequential browsing of textual material.	6	超文本是嵌入了文本链接的文本 ，这些链接可以指向同一文档或另一文档中的文本，从而允许非顺序地浏览文本材料。
1.9	1993 年推出的哪种浏览器导致了 Web 使用的巨大扩展？	In early 1993, things changed with the release of Mosaic, the first browser with a graphical user interface .	7	1993 年初，情况随着 Mosaic 的发布 而改变，它是 第一个具有图形用户界面的浏览器 。
1.10	在什么常见情况下，Web 服务器返回的文档是在接收到请求后创建的？	Sometimes a browser directly requests the execution of a program stored on the server. The output of the program is then returned to the browser .	7	有时浏览器直接请求执行存储在服务器上的程序。 程序的输出随后返回给浏览器 。
1.11	Web 服务器的文档根目录是什么？	The root of one of these [directories] is the document root . The file hierarchy that grows from the document root stores the Web documents to which the server has direct access and normally serves to clients .	9	其中一个（目录）的根是 文档根目录 。从文档根目录增长的文件层次结构 存储了服务器可以直接访问并通常提供给客户端的 Web 文档 。
1.12	什么是虚拟文档树？	The secondary areas from which documents can be served are called virtual document trees .	10	可以从中提供文档的辅助区域被称为 虚拟文档树 。
1.13	Web 服务器的服务器根目录是什么？	The root of the other directory is the server root . This directory, along with its descendant directories, stores the server and its support software .	9	另一个目录的根目录是 服务器根目录 。该目录及其后代目录 存储了服务器及其支持软件 。
1.14	什么是虚拟主机？	Such secondary hosts are called virtual hosts .	10	这种辅助主机被称为 虚拟主机 。
1.15	什么是代理服务器？	Some servers can serve documents that are in the document root of other machines on the Web; these are called proxy servers .	10	某些服务器可以提供位于 Web 上其他机器文档根目录中的文档；这些被称为 代理服务器 。
1.16	文件协议指定了什么？	The file protocol means that the document resides on the machine running the browser .	12	文件协议 意味着 文档驻留在运行浏览器的机器上 。
1.17	文档的部分路径如何在 Web 服务器中工作？	The path to the document is relative to some base path that is specified in the configuration files of the server .	13	文档的路径是 相对于服务器配置文件中指定的某个基本路径 的。
1.18	当浏览器请求一个目录但没有给出其名称时，Web 服务器通常返回的文件的名称是什么？	The server then searches at the top level of the directory in which servable documents are normally stored for something it recognizes as a home page. By convention, this page is often a file named index.html .	13	服务器随后会在通常存储可提供文档的目录的顶层搜索它识别为首页的文件。按照惯例，该页面通常是一个名为 index.html 的文件。
1.19	在浏览器和服务器之间的请求-响应事务中，MIME 类型规范的目的是什么？	When the browser receives the document from a Web server, it uses the included MIME format specification to determine what to do with the document .	13	当浏览器从 Web 服务器接收到文档时，它 使用包含的 MIME 格式规范来确定如何处理该文档 。
1.20	当 Web 服务器返回一个带有实验性 MIME 类型的文档时，它必须向浏览器提供什么？	The Web provider must supply a program that the browser can call when it needs to display the contents of the database.	14	Web 提供者必须提供一个 浏览器可以调用的程序 ，以便在需要显示数据库内容时使用。
1.21	描述五个最常用的 HTTP 方法的用途。	GET : Returns the contents of a specified document. HEAD : Returns the header information for a specified document.	15	GET ：返回指定文档的内容。 HEAD ：返回指定文档的头部信息。 POST ：执

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
		POST : Executes a specified document, using the enclosed data. PUT : Replaces a specified document with the enclosed data. DELETE : Deletes a specified document.		行指定文档, 使用随附的数据。 PUT : 用随附的数据替换指定文档。 DELETE : 删除指定文档。
1.22	HTTP 请求中 Accept 字段的目的是什么?	The Accept field ... specifies a preference of the browser for the MIME type of the requested document.	16	Accept 字段 ... 指定了浏览器对所请求文档 MIME 类型 的偏好。
1.23	哪个响应头字段是最常必需的?	The only essential field of the header is Content-type .	17	头部中唯一必需的字段是 Content-type 。
1.24	标记语言缺少什么重要功能?	HTML is not a programming language—it cannot be used to describe computations .	20	HTML 不是一种编程语言——它 不能用于描述计算 。
1.25	使用公钥加密方法解决了什么问题?	This problem was solved... who developed public-key encryption , a process in which a public key and a private key are used, respectively, to encrypt and decrypt messages.	19	这个问题通过开发 公钥加密 得到了解决, 公钥加密是一种分别使用公钥和私钥来加密和解密消息的过程。
1.26	从给定的公钥计算出相关的私钥在实践中是否可行?	The technical wording for this situation is that it is computationally infeasible to determine the private key from its public key.	19	这种情况的技术措辞是, 从公钥确定私钥是 计算上不可行的 。
1.27	病毒和蠕虫有什么区别?	Viruses are programs that often arrive in a system... and attach to other programs. Worms damage memory, like viruses, but spread on their own, rather than being attached to other files.	20,	病毒是程序 , 通常进入系统后附着到其他程序上。蠕虫像病毒一样破坏内存, 但它们是自行传播的, 而不是附着到其他文件上。
1.28	是什么促使黑客创建和传播病毒?	The incentive for these people apparently is simply the feeling of pride and accomplishment they derive from being able to cause huge amounts of damage by outwitting the designers of Web software systems.	20	这些人的动机显然只是因为他们通过智胜 Web 软件系统的设计者而能够 造成巨大破坏 所获得的成就感和自豪感。
1.29	什么是插件 (plug-in) ?	Plug-ins are programs that can be integrated with a word processor . Plug-ins add new capabilities to the word processor...	21, 22	插件是可与文字处理器集成的程序。插件为文字处理器添加新功能...
1.30	什么是过滤器 HTML 转换器 (filter HTML converter) ?	The second kind of tool is a filter, which converts an existing document in some form... to HTML .	22	第二种工具是 过滤器 , 它将某种形式的现有文档... 转换为 HTML。
1.31	为什么过滤器生成的代码在使用前通常必须手动修改?	The HTML output produced by either filters or plug-ins often must be modified, usually with a simple text editor, to perfect the appearance of the displayed document in the browser.	22	过滤器或插件生成的 HTML 输出通常必须 手动修改 , 通常使用简单的文本编辑器, 以完善显示文档在浏览器中的外观。
1.32	XML 在描述数据方面相对于 HTML 的巨大优势是什么?	The great advantage of XML is that application programs can be written to use the meanings of the tags in the given markup language to find specific kinds of data and process it accordingly.	23	XML 的巨大优势在于可以 编写应用程序来利用给定标记语言中标签的含义 , 从而找到特定类型的数据并进行相应的处理。
1.33	在基于 XML 的标记语言中, 预定义了多少种不同的标签?	XML is not a markup language; it is a meta-markup language that specifies rules for creating markup languages. As a result, XML includes no elements .	279	XML 不是一种标记语言; 它是一种元标记语言, 用于指定创建标记语言的规则。因此, XML 不包含任何元素 (标签) 。
1.34	Java 和 JavaScript 之间有什么关系?	The main similarity between Java and JavaScript is the syntax of their expressions, assignment statements, and control statements .	23	Java 和 JavaScript 之间最主要的 相似之处在于它们的表达式、赋值语句和控制语句的语法 。
1.35	JavaScript 最常见的应用是什么?	JavaScript is a client-side scripting language whose primary uses in Web programming are to validate form data, to build Ajax-enabled HTML documents and to create dynamic HTML documents .	23	JavaScript 是一种客户端脚本语言, 它在 Web 编程中的主要用途是 验证表单数据、构建启用 Ajax 的 HTML 文档以及创建动态 HTML 文档 。
1.36	JavaScript 最常在哪里被解释, 是在服务器上还是在浏览器上?	The JavaScript code in an HTML document is interpreted by an interpreter embedded in the browser on the client .	23	HTML 文档中的 JavaScript 代码由 嵌入在客户端浏览器中的解释器 解释。
1.37	Flash 的主要用途是什么?	Flash is now the leading technology for delivering graphics and animation on the Web .	24	Flash 现在是在 Web 上提供图形和动画的领先技术 。
1.38	Flash 影片在哪里被解释, 在服务器上还是在浏览器上?	These documents are served by Web servers to browsers, which use the Flash player plug-in to display the documents .	24	这些文档由 Web 服务器提供给浏览器, 浏览器使用 Flash 播放器插件来显示这些文档 。
1.39	Servlet 在哪里执行, 在服务器上还是在浏览器上?	A Java class called a servlet can be used for these applications. An object of which is executed on the server system when requested by the HTML document...	25	一种称为 servlet 的 Java 类可用于这些应用程序。它的对象 在服务器系统上执行 ...
1.40	Servlet 是用什么语言编写的?	A Java class called a servlet can be used for these applications.	25	一种称为 servlet 的 Java 类 可用于这些应用程序。
1.41	JSP 文档与 Servlet 在哪方面是相反的?	JSP takes an opposite approach to that of servlets: Instead of embedding HTML in Java code that provides dynamic documents, code of some form is embedded in HTML documents to provide the dynamic parts of a document.	25-26	JSP 采取了与 Servlet 相反的方法 : Servlet 是在 Java 代码中嵌入 HTML 来提供动态文档, 而 JSP 则是将 某种形式的代码嵌入到 HTML 文档 中来提供文档的动态部分。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
1.42	JSF 的目的是什么？	JavaServer Faces (JSF) adds another layer to the JSP technology. The most important contribution of JSF is an event-driven user interface model for Web applications .	26	JavaServer Faces (JSF) 为 JSP 技术添加了另一层。JSF 最重要的贡献是为 Web 应用程序提供了一个 事件驱动的用户界面模型 。
1.43	ASP.NET 的目的是什么？	Active Server Pages .NET (ASP.NET) is a Microsoft framework for building server-side dynamic documents .	26	Active Server Pages .NET (ASP.NET) 是一个用于 构建服务器端动态文档 的微软框架。
1.44	ASP.NET 文档中的代码通常是用什么语言编写的？	In most cases, it is done in either Visual Basic .NET or C# .	26	在大多数情况下，它使用 Visual Basic .NET 或 C# 。
1.45	PHP 代码在哪里被解释，在服务器上还是在浏览器上？	PHP code is embedded in HTML documents... the code is interpreted on the server before the HTML document is delivered to the requesting client.	24	PHP 代码嵌入在 HTML 文档中.....代码在 HTML 文档交付给请求客户端之前 在服务器上被解释 。
1.46	PHP 在哪些方面与 JavaScript 相似？	PHP is similar to JavaScript, both in terms of its syntactic appearance and in terms of the dynamic nature of its strings and arrays .	24	PHP 与 JavaScript 相似，无论是在 语法外观上，还是在其字符串和数组的动态特性上 。
1.47	Ruby 在哪些方面比 Java 更面向对象？	The primary characterizing feature of Ruby is that it is a pure object-oriented language , just as is Smalltalk. Every data value is an object and all operations are via method calls.	26	Ruby 的主要特征是它是一种 纯粹的面向对象语言 ，就像 Smalltalk 一样。 每个数据值都是一个对象 ，所有操作都通过方法调用进行。
1.48	Ruby 是在哪个国家开发的？	Ruby is culturally interesting because it is the first programming language designed in Japan that has achieved relatively widespread use outside that country.	27	Ruby 是在文化上很有趣，因为它是第一个在 日本设计 并在该国以外获得相对广泛使用的编程语言。
1.49	Rails 的目的是什么？	Rails is a development framework for Web-based applications that access databases.	27	Rails 是一个开发框架 ，用于访问数据库的 Web 应用程序。
1.50	Rails 是为哪种特定类型的 Web 应用程序设计的？	Rails was designed to make the construction of Web applications relatively quick and easy .	vi	Rails 旨在使 Web 应用程序的构建 相对快速和简单 。
1.51	Ajax 应用程序中使用了哪些编程语言？	Ajax uses JavaScript as its primary programming language. The x in Ajax represents eXtensible Markup Language (XML) .	404,	Ajax 使用 JavaScript 作为其主要的编程语言。Ajax 中的 x 代表 可扩展标记语言 (XML) 。
1.52	Ajax Web 应用程序与传统 Web 应用程序在根本上有何不同？	The communication from the browser to the server is asynchronous ... the browser need not wait for the server to respond. Second, the document provided by the server usually is only a relatively small part of the displayed document.	25	浏览器与服务器之间的通信是 异步的 浏览器不必等待服务器响应。其次，服务器提供的文档通常 只是显示文档中相对较小的一部分 。

好的，根据您提供的第2章内容，我将回答Review Questions中可见的问题（2.1至2.39）。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
2.1	HTML 的标签或属性被弃用（deprecated）是什么意思？	Deprecating a feature is a warning to users to stop using it because it will be dropped from HTML in the future.	161	弃用某个功能是警告用户停止使用它，因为它将在未来从 HTML 中删除。
2.2	HTML 注释的格式是什么？	They are written in HTML in the following form: <code><!-- anything except two adjacent dashes --></code>	172	它们在 HTML 中的书写形式如下： <code><!-- anything except two adjacent dashes --></code> （即： <code><!--</code> 任何内容，除了两个相邻的破折号 <code>--></code> ）。
2.3	浏览器如何处理待显示的文本中的换行符？	Browsers ignore line breaks . The browser puts as many words as will fit on the lines in the browser window . The browser supplies a line break at the end of each line .	173, 179	浏览器会 忽略换行符 。浏览器会 在浏览器窗口的行上放置尽可能多的单词 。浏览器会 在每行的末尾提供一个换行符 。
2.4	段落元素和换行元素的效果有什么区别？	If a paragraph tag appears at a position other than the beginning of the line, the browser breaks the current line and inserts a blank line . The break tag causes an explicit line break without the preceding blank line .	182, 183	如果段落标签出现在不是行首的位置，浏览器会 中断当前行并插入一个空行 。 换行标签 会造成 明确的换行，而不会在前面插入空行 。
2.5	哪些标题元素（heading elements）使用的字体大小小于普通文本字体大小？	h5 and h6 use smaller sizes.	185	h5 和 h6 使用较小的字号。
2.6	浏览器通常如何设置块引用（block quotations）使其不同于普通文本？	In most cases the block of text simply is indented on both sides .	187	在大多数情况下，这块文本通常是 两侧缩进的 。
2.7	<code>code</code> 元素为其内容指定了什么？	The code element, code, is used to specify a monospace font , usually for program code.	190	code 元素（标签名为 code）用于指定等宽字体 ，通常用于程序代码。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
2.8	JPEG 和 GIF 图像格式之间有什么区别？	GIF uses 8-bit color representations... JPEG format uses 24-bit color representations. The compression algorithm used by JPEG is better at shrinking an image. JPEG does not support transparency, while GIF images can be made to appear transparent.	199, 200, 201	GIF 使用 8 位颜色表示... JPEG 格式使用 24 位颜色表示。JPEG 使用的压缩算法在缩小图像方面优于 GIF。JPEG 不支持透明度，而 GIF 图像可以设置为透明。
2.9	img 元素的两个必需属性是什么？	The img element includes two attributes: src, which specifies the file containing the image; and alt, which specifies text to be displayed when it is not possible to display the image.	202	img 元素包含两个属性：src，用于指定包含图像的文件；和 alt，用于指定在无法显示图像时要显示的文本。
2.10	img 元素的 alt 属性的目的是什么？	The alt attribute is displayed whenever the browser cannot or has been instructed not to display the image.	203	alt 属性会在浏览器无法或已被指示不显示图像时显示。
2.11	使用哪个标签来定义链接？	Links are specified in an attribute of an anchor element, a.	212	链接是在锚点元素 a 的属性中指定的。
2.12	所有锚点标签中必需的属性是什么？	For creating links, only one attribute is required: href (an acronym for hypertext reference).	213	对于创建链接，只需要一个属性：href（是 hypertext reference 的首字母缩写）。
2.13	HTML 允许嵌套链接吗？	An anchor element... cannot be another anchor element.	214	锚点元素不能包含另一个锚点元素。
2.14	当链接的目标在当前显示的文档中但不在其开头时，通常如何识别该目标？	If the target element has an id attribute, that value can be used to specify the target.	217	如果目标元素具有 id 属性，则该值可用于指定目标。
2.15	当目标是链接所在文档之外的文档片段时，锚点标签中 href 属性的值是什么格式？	When the target is an element in another document, the value of that element's id is specified at the end of the URL, separated by a pound sign (#), as in aidan1.html#avionics.	219	当目标是另一个文档中的元素时，该元素 id 的值位于 URL 的末尾，并用一个井号 (#) 分隔，例如 aidan1.html#avionics。
2.16	无序列表（unordered list）中项目默认的项目符号形式是什么？	When displayed, each list item is implicitly preceded by a bullet.	222	当显示时，每个列表项前面隐式地带有项目符号。
2.17	有序列表（ordered list）中项目的默认顺序值是什么？	The default sequential values are Arabic numerals, beginning with 1.	223	默认的顺序值是阿拉伯数字，从 1 开始。
2.18	在定义列表（definition list）中，使用哪些标签来定义术语及其定义？	Each term to be defined in the definition list is given as the content of a dt element. The definitions themselves are specified as the content of dd elements.	227	定义列表中的每个术语是作为 dt 元素的内容给出的。定义本身是作为 dd 元素的内容指定的。
2.19	th 元素的 colspan 属性的目的是什么？	The colspan attribute specification in a table header or table data tag tells the browser to make the cell as wide as the specified number of rows below it in the table.	235	colspan 属性的规范（在表头或表数据标签中）告诉浏览器使单元格的宽度达到表中其下方指定行数所占的宽度。
2.20	td 元素的 rowspan 属性的目的是什么？	The rowspan attribute of the table heading and table data tags does for rows what colspan does for columns. (It makes the cell as high as the specified number of rows.)	235	表头和表数据标签的 rowspan 属性对行所做的操作与 colspan 对列所做的操作相同。（它使单元格的高度达到指定的行数）。
2.21	什么是控件（controls）？	The commonly used objects on a screen form... are called controls or widgets or components.	242	屏幕表单上常用的对象...被称为控件、小部件或组件。
2.22	本章讨论的哪些控件是使用 input 元素创建的？	The input element includes those for text, passwords, checkboxes, radio buttons, plain buttons, ranges of numbers, URLs, electronic mail addresses, reset, submit, and image.	246	input 元素包括用于文本、密码、复选框、单选按钮、普通按钮、数字范围、URL、电子邮件地址、重置、提交和图像的控件。
2.23	文本控件的文本框的默认大小是多少？	The default size of a text box is often 20 characters.	248	文本框的默认大小通常是 20 个字符。
2.24	文本控件的 input 元素的 size 和 maxlength 属性之间有什么区别？	The size attribute specifies the visible length of the box. The maxlength attribute specify the maximum number of characters that the browser will accept in the box.	249	size 属性指定框的可见长度。maxlength 属性指定浏览器在框中将接受的最大字符数。
2.25	一组复选框按钮和一组单选按钮在行为上有何不同？	Only one radio button can be on or pressed at any time. Every time a radio button is pressed, the button in the group that was previously on is turned off.	258–259	任何时候只能有一个单选按钮处于开启或按下状态。每当按下其中一个单选按钮时，该组中先前处于开启状态的按钮就会关闭。
2.26	在什么情况下使用菜单（menu）而不是单选按钮组？	If the number of choices is large, the form may become too long to display. In these cases, a menu should be used.	266	如果选项的数量很大，表单可能会变得太长而无法显示。在这种情况下，应该使用菜单。
2.27	如何为 textarea 控件指定滚动条？	The text typed into the area created by textarea is not limited in length, and there is implicit scrolling when needed, both vertically and horizontally.	270	输入到 textarea 创建的区域中的文本长度没有限制，并且在需要时会隐式地进行垂直和水平滚动。
2.28	解释 url 类型的 input 元素的行为。	The browser is supposed to check to determine whether the input could possibly be a valid URL. This check is to make sure the input includes a colon and that the colon is both preceded and followed by at least one character.	261, 262	浏览器应该检查以确定输入是否可能是一个有效的 URL。此检查旨在确保输入包含一个冒号，并且该冒号前后都至少有一个字符。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
2.29	解释 email 类型的 input 元素的行为。	The browser is supposed to determine whether the input could possibly be a valid electronic mail address . This check is to make sure the input includes an at-sign (@) and that it is both preceded and followed by at least one character.	263	浏览器应该确定输入是否可能是一个 有效的电子邮件地址 。此检查旨在确保输入包含一个 艾特符号 (@) ，并且该符号前后都至少有一个字符。
2.30	input 元素的 placeholder 属性的目的是什么？	The placeholder attribute ... is used to provide initial values to those elements, which serve as hints to the user as to what the input should be.	265–266	placeholder 属性用于为这些元素提供 初始值 ，作为 对用户关于输入内容的提示 。
2.31	在 HTML5 之前，当浏览器显示 Web 文档时，如何播放声音片段？	Until the arrival of HTML5 there was no standard way of doing that without a plug-in, such as Flash or Microsoft's Media Player .	278	在 HTML5 出现之前，如果没有 插件，例如 Flash 或微软的 Media Player ，就没有标准的方法来实现这一功能。
2.32	什么是音频编解码器 (audio codec) ？	Audio information is coded into digital streams with encoding algorithms called audio codecs .	278	音频信息是使用编码算法编码成数字流的，这些算法被称为 音频编解码器 。
2.33	什么是音频容器 (audio container) ？	A container can be thought of as a zip file; it is a way to pack data into a file, but the encoding of the data in the file is irrelevant to the container.	279	容器 可以被视为一个 zip 文件；它是一种将数据打包到文件中的方式，但文件中数据的编码方式与容器无关。
2.34	为什么一个 audio 元素会包含多个 source 元素？	Different browsers are capable of playing different audio container/codecs combinations.	280	不同的浏览器能够播放不同的音频容器/编解码器组合 。
2.35	在 HTML5 之前，当显示 Web 文档时，播放视频片段最常用的方法是什么？	The most common approach to video on the Web was the use of the Flash plug-in .	282	Web 上最常见的视频方法是使用 Flash 插件 。
2.36	video 元素的 autoplay 属性有什么作用？	The autoplay attribute specifies that the video plays automatically as soon as it is ready .	285	autoplay 属性指定视频在准备好后立即 自动播放 。
2.37	在 HTML5 之前，哪些 HTML 元素用于组织文档？	The primary elements for this in those languages [HTML 4.01 and XHTML 1.0] were division (div) and paragraph (p) .	287	在这些语言 (HTML 4.01 和 XHTML 1.0) 中，用于组织文档的主要元素是 division (div) 和 paragraph (p) 。
2.38	article 元素的目的是什么？	The article element is used to encapsulate a self-contained part of a document that comes from some external source , such as a post from a forum or a newspaper article.	291	article 元素用于封装文档中来自 外部源的独立部分 ，例如论坛帖子或报纸文章。
2.39	datetime 属性值的日期部分的格式是什么？	The date part of datetime is given as a four-digit year, a dash, the two-digit month, a dash, and the two-digit day of the month , for example, "2011-02-14" .	292	datetime 的日期部分格式为 四位数的年份、一个破折号、两位数的月份、一个破折号和两位数的月份中的日期 ，例如 "2011-02-14" 。

以下是第3章（Cascading Style Sheets，层叠样式表）的 Review Questions 的回答，所有内容均根据您提供的资料生成。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
3.1	文档级样式表相对于行内样式表的优势是什么？	Document style specifications appear in the document head section and apply to the entire body of the document . This is obviously an effective way to impose a uniform style on the presentation of all the content of a single document .	98	文档样式规范出现在文档的头部部分，并 应用于整个文档的主体 。这显然是 对单个文档的所有内容施加统一风格 的有效方式。
3.2	外部样式表的目的是什么？	In many cases, it is desirable to have a style sheet apply to more than one document .	98	在许多情况下，希望样式表 应用于多个文档 。
3.3	链接到外部样式表时需要哪些属性？	The rel attribute is used to specify the relationship of the linked-to document to the document in which the link appears. The href attribute... is used to specify the URL of the style sheet document.	98	rel 属性用于 指定 链接文档与包含该链接的文档之间的 关系 。 href 属性... 用于指定样式表文档的 URL 。
3.4	行内样式表的格式是什么？	Inline style specifications appear as values of the style attribute of a tag ... style = "property_1 : value_1; property_2 : value_2;...; property_n : value_n ;".	98	行内样式规范显示为 标签的 style 属性 的值... style = "property_1 : value_1; property_2 : value_2;...; property_n : value_n ;"。
3.5	文档级样式表的格式是什么，以及该样式表出现在哪里？	Document style specifications appear as the content of a style element within the header of a document ... <style type = "text/css"> rule_list </style>.	98, 99	文档样式规范显示为 文档头部内的 style 元素 的内容... <style type = "text/css"> rule_list </style>。
3.6	外部样式表的格式是什么？	An external style sheet consists of a list of style rules of the same form as in document style sheets . The <style> tag is not included .	99	外部样式表由 与文档样式表格式相同的样式规则列表 组成。其中 不包含 <style> 标签 。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
3.7	文档级样式表规则列表中注释的格式是什么？	CSS comments are introduced with / and terminated with / .	99	CSS 注释以 /* 开始，并以 */ 结束。
3.8	样式类选择器的目的是什么？	Class selectors are used to allow different occurrences of the same element to use different style specifications .	100	样式类选择器 用于 允许同一元素的不同出现使用不同的样式规范 。
3.9	通用类的目的是什么？	Sometimes it is convenient to have a class of style specifications that applies to the content of more than one kind of element .	100	有时，方便的做法是拥有一类样式规范，它 适用于不止一种元素的内容 。
3.10	选择器 ol ul 和 ol > ul 的区别是什么？	ol ul applies its style only to the content of ordered list elements that are descendants of unordered list elements... ol > ul applies to ul elements only if they are children of ol elements.	101, 102	ol ul 仅将其样式应用于作为无序列表元素的 后代 的有序列表元素的内容... ol > ul 仅适用于作为 ol 元素的 子元素 的 ul 元素。
3.11	描述专门用于链接的两个伪类。	The two pseudo classes for styling links are link and visited . The link pseudo class is used to style a link that has not been selected ; visited is used to style a link that previously has been selected .	103	专用于链接样式的两个伪类是 link 和 visited 。 link 伪类用于设置尚未被选中的链接的样式； visited 用于设置先前已被选中的链接的样式。
3.12	关键字属性值是区分大小写还是不区分大小写？	Keyword property values are not case sensitive , so Small, SmAll, and SMALL are all the same as small.	104	关键字属性值不区分大小写 ，因此 Small、SmAll 和 SMALL 都与 small 相同。
3.13	为什么 font-family 属性的值是一组字体名称列表？	The browser uses the first font in the list that it supports .	105	浏览器使用 列表中它支持的第一个字体 。
3.14	五种通用字体是什么？	The five generic fonts are Serif, Sans-serif, cursive, fantasy, and monospace .	106	这五种通用字体是 Serif、Sans-serif、cursive、fantasy 和 monospace 。
3.15	为什么使用 em 设置字体大小比使用 pt 更好？	One problem with using points and picas for font sizes is that they do not display in the same size on different computers . If a relative size is given [such as em], the font size will be scaled relative to a new default set by the user .	107	使用磅和皮卡设置字体大小的一个问题是它们 在不同的计算机上显示尺寸不同 。如果给出了相对大小，字体大小将相对于用户设置的新默认值进行缩放。
3.16	属性值必须以什么顺序出现在 font 属性的列表中？	The order must be as follows: The font names must be last , the font size must be second to last , and the font style, font variant, and font weight... can be in any order but must precede the font size .	108	顺序必须如下： 字体名称必须放在最后 ， 字体大小必须倒数第二 ，而字体样式、字体变体和字体粗细...可以按任何顺序排列，但 必须排在字体大小之前 。
3.17	可以通过 text-decoration 以哪些方式修改文本？	The available values are line-through, overline, underline , and none.	110	可用的值有 line-through （删除线）、 overline （上划线）、 underline （下划线）和 none 。
3.18	什么是字距（tracking）和行距（leading）？	The letter-spacing property controls the amount of space between the letters in words. This spacing is called tracking . The space between lines of text can be controlled with the line-height property. This spacing is called leading .	111	letter-spacing 属性控制单词中字母之间的空间量。这种间距被称为 字距 （tracking）。文本行之间的空间可以通过 line-height 属性控制。这种间距被称为 行距 （leading）。
3.19	list-style-type 属性如何与无序列表一起使用？	The list-style-type property of an unordered list can be set to disc (the default), circle, square, or none .	113	无序列表的 list-style-type 属性可以设置为 disc （默认）、 circle、square 或 none 。
3.20	list-style-type 属性与有序列表一起使用时，可能的取值有哪些？	Possible values include decimal, decimal-leading-zero, lower-alpha, upper-alpha, lower-roman, upper-roman , lower-greek, lower-latin, upper-latin, armenian, georgian, and None.	115	可能的值包括 decimal （阿拉伯数字）、 decimal-leading-zero （零开头的阿拉伯数字）、 lower-alpha （小写字母）、 upper-alpha （大写字母）、 lower-roman （小写罗马数字）和 upper-roman （大写罗马数字）等。
3.21	如果你希望文本围绕图像的右侧流动，那么 float 属性应该被赋予 right 还是 left 值？	To specify this condition [text flowing around the left side of the image while the image is on the right side of the display], the float property of the image is set to right .	118	要指定这种情况，则图像的 float 属性设置为 right 。
3.22	为什么必须谨慎选择背景图像？	Text over a background image can be difficult or even impossible to read if the image has areas that are nearly the same color as the text.	128	如果背景图像的某些区域颜色与文本颜色相近， 文本在背景图像上可能会变得难以阅读甚至无法阅读 。
3.23	text-align 属性可能的取值有哪些？	The most commonly used keyword values are left, center, right, and justify .	118	最常用的关键字值是 left、center、right 和 justify 。
3.24	text-indent 属性的作用是什么？	The text-indent property can be used to indent the first line of a paragraph .	117	text-indent 属性可用于 缩进段落的首行 。
3.25	用于设置元素周围边距的属性有哪些？	The margin properties are named margin , which applies to all four sides of an element, margin-left, margin-right, margin-top, and margin-bottom .	124	边距属性命名为 margin （适用于元素的四边），以及 margin-left、margin-right、margin-top 和 margin-bottom 。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
3.26	颜色属性值可以通过哪三种方式指定？	Color property values can be specified as color names , as six-digit hexadecimal numbers , or in RGB form .	104	颜色属性值可以指定为 颜色名称 、 六位十六进制数字 或以 RGB 形式 。
3.27	如果你希望背景图像垂直重复但不水平重复，需要将哪个属性设置为哪个值？	The background-repeat property ... repeat-y means that the image is to be repeated vertically.	128	background-repeat 属性 必须设置为 repeat-y ，这意味着图像将垂直重复。
3.28	如果文本框的左边框是红色且细的，右边框是蓝色且粗的，你需要使用哪些属性和值来在其周围设置虚线边框？	必须使用 border-style 属性设置为 dotted （虚线）。左边框需设置 border-left-color 为 red ， border-left-width 为 thin 。右边框需设置 border-right-color 为 blue ， border-right-width 为 thick 。	121	The border-style must be set to dotted . The left border requires border-left-color to red and border-left-width to thin . The right border requires border-right-color to blue and border-right-width to thick .
3.29	边框样式的简写属性是什么？	There is shorthand for setting the style properties of all four borders of an element, border .	121	设置元素四边样式属性的简写是 border 。
3.30	border-collapse 属性的目的是什么？	To get just one border between table cells [instead of the default double borders]... we set the table property border-collapse to collapse .	122	为了使表格单元格之间只有一条边框，我们将表格属性 border-collapse 设置为 collapse 。
3.31	标签本身向浏览器指示了哪些布局信息？	Unlike most other tags, there is no default layout for the content of .	128	与大多数其他标签不同， 的内容没有默认的布局 。
3.32	元素的作用是什么？	Its primary use is to specify presentation details for a section or division of a document .	129	它的主要用途是 为文档的某个部分或划分指定呈现细节 。
3.33	id 选择器和通用选择器（*）哪个具有更高的优先级？	id selectors [have the highest precedence in specificity]... followed by Class and pseudo class selectors, Contextual selectors, and Universal selectors . 因此， id 选择器 具有更高的优先级。	131	id selectors （ID 选择器）具有更高的优先级，其后是类和伪类选择器、上下文选择器，最后是 通用选择器 。
3.34	用户来源规范和浏览器规范哪个具有更高的优先级？	The multi-stage sorting process specifies that Normal declarations with user origin take precedence over Any declarations with browser (or other user agent) origin .	131	多阶段排序过程规定， 用户来源的普通声明 优先于 浏览器（或其他用户代理）来源的任何声明 。
3.35	如果文档级样式表中有两条相互冲突的规范，哪一条具有优先权？	If there are still conflicts, they are resolved by giving precedence to the most recently seen specification .	132	如果仍然存在冲突，则通过赋予 最近出现的规范 优先权来解决。

以下是根据您提供的第4章内容对 Review Questions 的回答。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
4.1	简要描述 Java 和 JavaScript 之间的三个主要区别。	One important difference is support for object-oriented programming . Another important difference between Java and JavaScript is that objects in Java are static ... JavaScript objects are dynamic . Java is a strongly typed language . Variables in JavaScript need not be declared and are dynamically typed .	139	一个重要的区别是对 面向对象编程 的支持。Java 和 JavaScript 之间的另一个重要区别是 Java 中的对象是 静态的而 JavaScript 对象是 动态的 。Java 是一种 强类型语言 。JavaScript 中的变量不需要声明，并且是 动态类型 的。
4.2	简要描述 JavaScript 在客户端的三个主要用途。	Client-side JavaScript can serve as an alternative for some of what is done with server-side programming . Interactions with users through form elements , such as buttons and menus, can be conveniently described in JavaScript. Another interesting capability of JavaScript was made possible by the development of the Document Object Model (DOM), which allows JavaScript scripts to access and modify the style properties and content of the elements of a displayed HTML document, making formally static documents highly dynamic .	139, 140	客户端 JavaScript 可以替代部分 服务器端编程 的工作。与用户通过 表单元素 （例如按钮和菜单）进行的交互，可以用 JavaScript 方便地描述。JavaScript 的另一个有趣功能是通过文档对象模型（DOM）的开发而实现的，它允许 JavaScript 脚本访问和修改所显示 HTML 文档的元素样式属性和内容，从而使原本静态的文档变得高度 动态 。
4.3	简要描述事件驱动计算的基本过程。	Much of what JavaScript scripts typically do is event driven, meaning that the actions often are executed in response to the browser user's actions , among them mouse clicks and form submissions.	140	JavaScript 脚本通常做的事情大部分是 事件驱动的，这意味着操作通常是响应浏览器用户的行为而执行的 ，其中包括鼠标点击和表单提交。
4.4	JavaScript 中属性的两类是什么？	Each property is either a data property or a function or method property .	141	每个属性要么是 数据属性 ，要么是 函数或方法属性 。
4.5	为什么 JavaScript 有两类数据变量，即原始值（primitives）和对象（objects）？	Primitives are used because they often can be implemented directly in hardware, resulting in faster operations on their values (faster than if they were treated as objects).	141	使用原始值是因为它们通常可以直接在硬件中实现，从而对它们的值进行更快的操作 （比将它们视为对象更快）。
4.6	描述将 JavaScript 脚本嵌入 HTML 文档的两种方式。	There are two different ways to embed JavaScript in an HTML document: implicitly and explicitly .	140	有两种不同的方式将 JavaScript 脚本嵌入 HTML 文档： 隐式嵌入 和 显式嵌入 。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
4.7	JavaScript 注释的两种形式是什么？	Two adjacent slashes (//) appear on a line, the rest of the line is considered a comment. Second, /* may be used to introduce a comment, and */ to terminate it, in both single- and multiple-line comments.	143	行上出现两个相邻的斜杠 (//) , 该行的其余部分被视为注释。其次, /* 可用于引入注释, 而 */ 可用于终止注释, 适用于单行和多行注释。
4.8	为什么 JavaScript 脚本有时通过将它们放入 HTML 注释中来隐藏？	A few browsers that are still in use are so old that they do not recognize the <code><script></code> tag. Such browsers would display the contents of the script element as if it were just text. It has been customary to enclose the contents of all script elements in HTML comments to avoid this problem.	143	仍然有一些在用的浏览器非常老旧, 它们无法识别 <code><script></code> 标签。此类浏览器会将脚本元素的内容显示为普通文本。通常的做法是 将所有脚本元素的内容都包含在 HTML 注释中以避免此问题。
4.9	JavaScript 中的五种原始数据类型是什么？	JavaScript has five primitive types: Number, String, Boolean, Undefined, and Null.	145	JavaScript 有五种原始类型: Number、String、Boolean、Undefined 和 Null。
4.10	单引号字符串字面值与双引号字符串字面值是否有任何不同的特性？	There is no difference between single-quoted and double-quoted literal strings.	146	单引号和双引号字面字符串之间 没有区别。
4.11	在什么情况下变量的值会是 undefined ？	If a variable has been explicitly declared, but not assigned a value , it has the value undefined.	146	如果一个变量已被 显式声明, 但未赋值 , 则它的值是 undefined。
4.12	如果 undefined 值用作布尔表达式, 它会被解释为 true 还是 false ？	If undefined is used as a Boolean, it is false .	158	如果 undefined 用作布尔值, 则为 false 。
4.13	语言中的运算符优先级规则有什么作用？	The precedence rules of a language specify which operator is evaluated first when two operators with different precedences are adjacent in an expression.	147	语言的 优先级规则 规定了当表达式中 两个具有不同优先级的运算符相邻 时, 哪个运算符 先被评估 。
4.14	语言中的运算符结合性规则有什么作用？	The associativity rules of a language specify which operator is evaluated first when two operators with the same precedence are adjacent in an expression.	148	语言的 结合性规则 规定了当表达式中 两个具有相同优先级的运算符相邻 时, 哪个运算符 先被评估 。
4.15	描述 NaN 的用途和特征。	Any arithmetic operation that results in an error... or that produces a value that cannot be represented... returns the value "not a number," which is displayed as NaN . If NaN is compared for equality against any number, the comparison fails. Surprisingly, in a comparison, NaN is not equal to itself .	149	任何导致错误的算术运算.....或产生无法表示的值.....都会返回 "非数字" (not a number) 值, 该值显示为 NaN 。如果将 NaN 与任何数字进行相等性比较, 则比较失败。令人惊讶的是, 在比较中, NaN 不等于它自己 。
4.17	如果操作数是对象, typeof 返回什么值？	If the operand is an object or null , <code>typeof</code> produces "object" .	152	如果操作数是 对象或 null , <code>typeof</code> 产生 "object" 。
4.18	传递给 <code>document.write</code> 的字符串操作数通常的行尾标点符号是什么？	Because <code>write</code> is used to create markup, the only useful punctuation in its parameter is in the form of HTML tags. Therefore, the parameter of <code>write</code> often includes .	154	因为 <code>write</code> 用于创建标记, 所以其参数中唯一有用的标点符号是 HTML 标签形式的。因此, <code>write</code> 的参数通常包含。
4.19	传递给 <code>alert</code> 的字符串操作数通常的行尾标点符号是什么？	The string parameter of <code>alert</code> is not HTML code; it is plain text. Therefore, the string parameter of <code>alert</code> may include <code>\n</code> but never should include <code>
</code> .	155	<code>alert</code> 的字符串参数不是 HTML 代码; 它是纯文本。因此, <code>alert</code> 的字符串参数可以包含 <code>\n</code> 但绝不应包含 <code>
</code> 。
4.20	描述 <code>prompt</code> 方法的操作。	The prompt method creates a dialog window that contains a text box used to collect a string of input from the user, which prompt returns as its value . This window also includes two buttons: OK and Cancel. <code>prompt</code> takes two parameters: the string that prompts the user for input and a default string in case the user does not type a string before pressing one of the two buttons.	156	prompt 方法创建一个对话框窗口, 其中包含一个用于从用户收集字符串输入的文本框, prompt 将其作为值返回 。该窗口还包括两个按钮: OK 和 Cancel。 <code>prompt</code> 接受两个参数: 提示用户输入的字符串, 以及在用户未在按下两个按钮之一之前输入字符串时的默认字符串。
4.21	什么是控制结构 (control construct) ？	A control construct is a control statement together with the statement or compound statement whose execution it controls .	158	控制结构是控制语句以及其控制执行的语句或复合语句的组合。
4.22	JavaScript 中控制表达式的三种可能形式是什么？	The expressions upon which statement flow control can be based include primitive values, relational expressions, and compound expressions .	158	语句流控制所依据的表达式包括 原始值、关系表达式和复合表达式 。
4.23	<code>==</code> 和 <code>===</code> 之间有什么区别？	The operators <code>==</code> and <code>!=</code> allow type conversion of the operands if they are of different types. The operators <code>===</code> and <code>!==</code> disallow type conversion of either operand.	159	运算符 <code>==</code> 和 <code>!=</code> 允许对类型不同的操作数进行类型转换。运算符 <code>===</code> 和 <code>!==</code> 禁止 对任何操作数进行类型转换。
4.24	表达式的短路求值是什么意思？	<code>**&&</code> and	are short-circuit operators <code>**</code> . This means that if the value of the first operand of either	or <code>&&</code> determines the value of the expression, the second operand is not evaluated and the Boolean operator does nothing.

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
4.25	描述 <code>break</code> 语句的语义。	The break statement... transfers control out of the compound statement in which it appears.	160	break 语句.....将控制权转移到其所在的复合语句之外。
4.26	<code>while</code> 语句和 <code>do-while</code> 语句之间有什么区别?	The difference is that the test for completion is logically (and physically) at the end [of the <code>do-while</code>], rather than at the beginning [of the <code>while</code>], of the loop construct.	164	区别在于完成测试在循环结构的 末尾 (<code>do-while</code>) , 而不是在 开头 (<code>while</code>) 。
4.27	JavaScript 构造函数何时被调用?	Objects are often created with a new expression , which must include a call to a constructor method .	165	对象通常是通过 new 表达式 创建的, 该表达式必须包含对 构造函数方法 的调用。
4.28	Java 和 JavaScript 中的构造函数有什么区别?	In an object-oriented language such as Java... the constructor initializes members but does not create them. In JavaScript, however, the new operator creates a blank object... The constructor both creates and initializes the properties.	165	在像 Java 这样的面向对象语言中.....构造函数初始化成员但不创建它们。但在 JavaScript 中, <code>new</code> 运算符创建一个空白对象..... 构造函数既创建又初始化属性。
4.29	使用 <code>new</code> 运算符和 <code>Object</code> 构造函数创建的对象属性是什么?	The following statement creates an object that has no properties : <code>var my_object = new Object();</code>	165	以下语句创建了一个 没有属性 的对象: <code>var my_object = new Object();</code>
4.30	描述引用对象属性的两种方式。	First, any property can be accessed in the same way it is assigned a value, namely, with the object-dot-property notation . Second, the property names of an object can be accessed as if they were elements of an array... the property name (as a string literal) is used as a subscript .	165, 166	首先, 任何属性都可以以分配给它的相同方式访问, 即使用 对象-点-属性记法 。其次, 对象的属性名可以像数组的元素一样访问..... 属性名 (作为字符串字面值) 用作下标。
4.31	如何创建对象的新属性?	A property for an object is created by assigning a value to its name .	165	通过 给其名称赋值 来创建对象的一个属性。
4.32	描述 <code>for-in</code> 语句的语义。	The variable, <code>prop</code> , takes on the values of the properties of the my_car object, one for each iteration .	166	变量 <code>prop</code> 在每次迭代中取 my_car 对象的属性值, 每个属性一个值。
4.33	创建 <code>Array</code> 对象的两种方式是什么?	The usual way to create any object is to apply the new operator to a call to a constructor . The second way... is with a literal array value, which is a list of values enclosed in brackets .	166, 167	创建任何对象的通常方式是对 构造函数的调用应用 new 运算符 。第二种方式.....是使用 数组字面值, 它是包含在方括号中的值列表。
4.34	<code>Array</code> 的 <code>length</code> 属性值与对象中实际存在的元素数量之间有什么关系?	The length of an array is the highest subscript to which a value has been assigned, plus 1 . The length property of an array is not necessarily the number of elements allocated.	167	数组的长度是已赋值的最高下标加 1 。数组的 <code>length</code> 属性不一定是已分配的元素数量。
4.35	描述 <code>Array</code> 的 <code>join</code> 方法的语义。	The <code>join</code> method converts all the elements of an array to strings and catenates them into a single string . If a string parameter is provided, it is used as the element separator.	169	<code>join</code> 方法 将数组的所有元素转换为字符串, 并将它们连接成一个单一的字符串 。如果提供了字符串参数, 它将用作元素分隔符。
4.36	描述 <code>slice</code> 方法在只给定一个参数时的语义。	If <code>slice</code> is given just one parameter, the array that is returned has all the elements of the object, starting with the specified index .	169	如果 <code>slice</code> 只给定一个参数, 返回的数组包含 从指定索引开始的该对象的所有元素 。
4.37	嵌套数组字面值的形式是什么?	A two-dimensional array is implemented in JavaScript as an array of arrays . This can be done... with nested array literals .	170	二维数组在 JavaScript 中实现为 数组的数组 。这可以通过 嵌套数组字面值 来完成。
4.38	如果函数不包含 <code>return</code> 语句, 它会返回什么值?	If there are no return statements in a function or if the specific return that is executed does not include an expression, the value returned is undefined .	171	如果函数中没有 <code>return</code> 语句, 或者执行的特定 <code>return</code> 不包含表达式, 则返回的值是 undefined 。
4.39	变量未在函数中声明时, 其作用域在嵌入 HTML 文档的 JavaScript 脚本中是如何定义的?	Variables that are implicitly declared have global scope—that is, they are visible in the entire HTML document (or entire file if the script is in its own file) —even if the implicit declaration occurs within a function definition.	172	隐式声明的变量具有 全局作用域——即它们在整个 HTML 文档中可见 (如果脚本在自己的文件中, 则在整个文件中可见) ——即使隐式声明发生在函数定义内。
4.40	JavaScript 函数中是否可以引用全局变量?	If a name that is defined both as a local variable and as a global variable appears in a function, the local variable has precedence, effectively hiding the global variable with the same name .	172	如果一个名称在函数中既被定义为局部变量又被定义为全局变量, 则 局部变量具有优先权, 有效地隐藏了同名的全局变量 。
4.41	在函数中使用局部变量有什么好处?	When you make up their names, you need not be concerned that a global variable with the same name may exist somewhere in the collection of scripts in the HTML document.	172	当你给它们命名时, 你不需要担心 同名的全局变量可能存在于 HTML 文档中的脚本集合中的某个地方 。
4.42	JavaScript 使用哪种参数传递方法?	JavaScript uses the pass-by-value parameter-passing method.	172	JavaScript 使用 按值传递 的参数传递方法。
4.43	JavaScript 是否检查实际参数的类型与其对应形式参数的类型?	The number of parameters in a function call is not checked against the number of formal parameters in the called function.	173	函数调用中的参数 数量不会被检查 , 以匹配被调用函数中形式参数的数量。
4.44	函数如何访问那些与形式参数不对应的实际参	All parameters are communicated through a property array, arguments , that... stores the values of the	173	所有参数都通过一个属性数组 arguments 进行通信.....它存储了传

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
	数值？	parameters that are passed.		递的参数值。
4.45	原始变量可以按引用传递给函数的一种方式是什么？	One inelegant way is to put the value in an array and pass the array .	174	一种不优雅的方式是 将值放入数组中并传递该数组 。
4.46	在 JavaScript 中，构造函数究竟做了什么？	JavaScript constructors are special functions that create and initialize the properties of newly created objects .	177	JavaScript 构造函数是特殊的函数，用于 创建和初始化新创建的对象 的属性。
4.47	模式中的字符类是什么？	Such classes are defined by placing the desired characters in brackets . Dashes can appear in character class definitions, making it easy to specify sequences of characters.	179	此类（字符）是通过 将所需的字符放入方括号 中来定义的。破折号可以出现在字符类定义中，从而方便指定字符序列。
4.48	什么是预定义的字符类，它们是什么意思？	\d () matches A digit ; \D (o-9) matches Not a digit ; \w ([A-Za-z_o-9]) matches A word character (alphanumeric); \W (a-za-z_o-9) matches Not a word character ; \s ([\r\t\n\f]) matches A white-space character ; \S ([^ \r\t\n\f]) matches Not a white-space character .	179	\d () 匹配 一个数字 ； \D (o-9) 匹配 非数字 ； \w ([A-Za-z_o-9]) 匹配 一个单词字符 （字母数字）； \W (a-za-z_o-9) 匹配 非单词字符 ； \s ([\r\t\n\f]) 匹配 一个空白字符 ； \S ([^ \r\t\n\f]) 匹配 非空白字符 。
4.49	什么是符号量词（symbolic quantifiers），它们是什么意思？	The three symbolic quantifiers are asterisk (*) , plus (+) , and question mark (?) . An asterisk means zero or more repetitions , a plus sign means one or more repetitions , and a question mark means one or none .	180	三个符号量词是 星号 (*) 、 加号 (+) 和 问号 (?) 。星号表示 零或多次重复 ，加号表示 一次或多次重复 ，问号表示 一次或零次重复 。
4.50	描述两种行尾锚点（end-of-line anchors）。	A pattern is tied to a position at one of the ends of a string with an anchor. It can be specified to match only at the beginning of the string by preceding it with a circumflex (^) anchor. A pattern can be specified to match at the end of a string only by following the pattern with a dollar sign (\$) anchor.	180, 181	模式通过锚点绑定到字符串的一端位置。可以通过在模式前面加上 脱字符 (^) 锚点来指定它仅匹配字符串的开头。可以通过在模式后面加上 美元符号 (\$) 锚点来指定它仅匹配字符串的末尾。
4.51	i 模式修饰符有什么作用？	The i modifier makes the letters in the pattern match either uppercase or lowercase letters in the string.	181	i 修饰符使模式中的字母匹配字符串中的 大写或小写字母 。
4.52	String 方法 replace 究竟做了什么？	The replace method is used to replace substrings of the String object that match the given pattern . The replace method takes two parameters: the pattern and the replacement string.	181	replace 方法用于 替换匹配给定模式的 String 对象的子字符串 。 replace 方法接受两个参数：模式和替换字符串。
4.53	String 方法 match 究竟做了什么？	The match method... takes a single parameter: a pattern. It returns an array of the results of the pattern-matching operation . If the pattern has the g modifier, the returned array has all the substrings of the string that matched.	182	match 方法.....接受一个参数：模式。它 返回一个包含模式匹配操作结果的数组 。如果模式带有 g 修饰符，则返回的数组包含所有匹配的子字符串。

以下是针对第 5 章（JavaScript 和 HTML 文档）的 Review Questions 的回答，所有内容均基于您提供的资料中的第 5 章内容。

5.1

- **中文翻译：** JavaScript 中的全局变量是哪个对象的属性？
- **英文回答：** The **properties of the Window object are visible to all JavaScript scripts** that appear either implicitly or explicitly in the window's HTML document, so they include all the global variables.
- **依据页码：** 194
- **中文回答：** **Window 对象的属性**对所有在该窗口的 HTML 文档中隐式或显式出现的 JavaScript 脚本可见，因此它们包括了所有全局变量。

5.2

- **中文翻译：** 在 JavaScript 绑定到 DOM 中，HTML 元素和属性是如何表示的？
- **英文回答：** In the JavaScript binding to the DOM, the **elements of a document are objects**, with both data and operations. The data are called **properties**, and the operations are, naturally, called methods. The **property names in JavaScript are the same as their corresponding attribute names in HTML**.
- **依据页码：** 196
- **中文回答：** 在 JavaScript 绑定到 DOM 中，**文档的元素是对象**，包含数据和操作。数据被称为**属性**，操作自然被称为方法。**JavaScript 中的属性名称与其在 HTML 中的相应属性名称相同**。

5.3

- **中文翻译：** 什么是事件（event）？
- **英文回答：** An **event is a notification that something specific has occurred**, either in the browser, such as the completion of the loading of a document, or a browser user action, such as a mouse click on a form button.
- **依据页码：** 201
- **中文回答：** **事件是关于特定情况发生的一种通知**，可能发生在浏览器中（例如文档加载完成），或者由浏览器用户行为引起（例如在表单按钮上点击鼠标）。

5.4

- 中文翻译: 什么是事件处理程序 (event handler) ?
- 英文回答: An **event handler is a script that is implicitly executed in response to the appearance of an event.**
- 依据页码: 202
- 中文回答: 事件处理程序是为响应事件出现而隐式执行的脚本。

5.5

- 中文翻译: DOM o 事件模型的起源是什么?
- 英文回答: The **HTML 4.0 standard provided the first specification of an event model for markup documents.** This model is sometimes referred to as the DOM o event model.
- 依据页码: 201
- 中文回答: **HTML 4.0 标准为标记文档提供了第一个事件模型规范。** 该模型有时被称为 DOM o 事件模型。

5.6

- 中文翻译: 在 DOM o 事件模型中, 事件处理程序可以通过哪两种方式与特定 HTML 元素生成的事件关联起来?
- 英文回答: One of these is by **assigning the event handler script to an event tag attribute.** The alternative is to **assigning the name of the handler to the associated event property** on the object,.
- 依据页码: 204, 205
- 中文回答: 第一种方式是**将事件处理程序脚本分配给事件标签属性**; 另一种方式是**将处理程序的名称分配给对象上关联的事件属性**。

5.7

- 中文翻译: 为什么不应该在事件处理程序中使用 document.write?
- 英文回答: The **write method of document should never be used in an event handler.** Events usually occur after the whole document is displayed. If write appears in an event handler, the **content produced by it might be placed over the top of the currently displayed document.**
- 依据页码: 202
- 中文回答: **document.write 方法不应在事件处理程序中使用。** 事件通常发生在整个文档显示之后。如果 `write` 出现在事件处理程序中, 它生成的内容可能会被放置在当前显示的文档之上。

5.8

- 中文翻译: HTML 表单元素可以通过哪些方式获取焦点?
- 英文回答: An HTML element is said to **get focus when the user puts the mouse cursor over it and clicks the left mouse button.** An element can also get focus **when the user tabs to the element.**
- 依据页码: 203
- 中文回答: 当用户将鼠标光标放在其上方并单击鼠标左键时, HTML 元素被认为**获得了焦点**。元素也可以在**用户使用 Tab 键切换到该元素时**获得焦点。

5.9

- 中文翻译: 描述使用 forms 和 elements 数组寻址 HTML 元素的方法。
- 英文回答: The original (DOM o) way is to use the **forms and elements arrays of the Document object**, which is referenced through the document property of the Window object.
- 依据页码: 199
- 中文回答: 最初 (DOM o) 的方法是使用 **Document 对象的 forms 和 elements 数组**, 这些数组通过 Window 对象的 document 属性引用。

5.10

- 中文翻译: 描述使用 name 属性寻址 HTML 元素的方法。
- 英文回答: Another approach to DOM addressing is to use element names. For this, the **element and its enclosing elements, up to but not including the body element, must include name attributes.**
- 依据页码: 200
- 中文回答: 另一种 DOM 寻址方法是使用元素名称。为此, **该元素及其包含元素, 直到 (但不包括) body 元素, 都必须包含 name 属性。**

5.11

- 中文翻译: 描述使用 getElementById 寻址 HTML 元素的方法。
- 英文回答: Yet another approach to element addressing is to use the **JavaScript method getElementById**, which is defined in DOM 1. Because an element's identifier (id) is **unique in the document**, this approach works, regardless of how deeply the element is nested in other elements in the document.
- 依据页码: 200
- 中文回答: 另一种元素寻址方法是使用 **JavaScript 方法 getElementById**。因为元素的标识符 (id) 在**文档中是唯一的**, 所以无论该元素在文档中的其他元素中嵌套多深, 此方法都有效。

5.12

- 中文翻译: 将事件处理程序分配给事件属性的缺点是什么?

- 英文回答: There is **no way to specify parameters on the handler function** when it is registered by assigning its name to the event property. Therefore, event handlers that are registered this way **cannot use parameters**.
- 依据页码: 212
- 中文回答: 通过将处理程序的名称分配给事件属性来注册处理程序时, **无法为处理程序函数指定参数**。因此, 以这种方式注册的事件处理程序**不能使用参数**。

5.13

- 中文翻译: 将事件处理程序分配给事件属性的优点是什么?
- 英文回答: First, it is good to **keep HTML and JavaScript separated in the document**. Second, having the handler function registered as the value of a property allows for the possibility of **changing the function during use**.
- 依据页码: 212
- 中文回答: 首先, **有助于将 HTML 和 JavaScript 在文档中分开**。其次, 将处理程序函数注册为属性值, 可以在**使用过程中更改该函数**。

5.14

- 中文翻译: 为什么最好在将表单数据发送到服务器之前使用 JavaScript 检查表单输入的有效性?
- 英文回答: It is obviously more **efficient to perform input data checks and carry on the user dialog concerning invalid input data entirely on the client**. This approach **shifts the task from the usually busy server to the client**... It also **results in less network traffic**... Furthermore, detecting incorrect form data on the client **produces quicker responses to users**.
- 依据页码: 215
- 中文回答: 显然, 在客户端执行输入数据检查和处理无效输入数据的用户对话**更有效率**。这种方法将**任务从通常繁忙的服务器转移到客户端**...它还可以**减少网络流量**...此外, 在客户端检测不正确的表单数据**可以为用户提供更快的响应**。

5.15

- 中文翻译: 当发现表单输入元素的数据格式不正确时, 应该做哪三件事?
- 英文回答: The function should **produce an alert message indicating the error to the user** and inform the user of the correct format for the input. Next, it would be good to **put the element in focus**. If an event handler returns **false, that tells the browser not to perform any default actions** of the event,.
- 依据页码: 215, 216
- 中文回答: 函数应**生成一个警告信息 (alert message) 向用户指出错误**并告知正确的输入格式。接下来, 最好**将该元素置于焦点 (in focus)**。此外, 如果事件处理程序返回 **false**, **则会通知浏览器不执行事件的任何默认操作**。

5.16

- 中文翻译: 当 onsubmit 事件的处理程序返回 false 时会发生什么?
- 英文回答: If an event handler returns false, that tells the browser **not to perform any default actions of the event**. For example, if the event is a click on the Submit button, the default action is to submit the form data to the server for processing. Returning false **prevents submission**.
- 依据页码: 216
- 中文回答: 如果事件处理程序返回 false, 它会通知浏览器**不执行事件的任何默认操作**。例如, 单击“提交”按钮的默认操作是提交表单数据, 因此返回 false **会阻止提交**。

5.17

- 中文翻译: 哪个事件用于触发事件处理程序, 以检查表单中文本按钮输入的有效性?
- 英文回答: The check is triggered by **a change event** (an event that occurs when a text field value is changed and the element loses focus)。
- 依据页码: 218
- 中文回答: 检查是由 **change 事件**触发的 (当文本字段值被更改且元素失去焦点时发生的事件)。

5.18

- 中文翻译: DOM o 事件模型中发生了什么事件传播?
- 英文回答: When the browser senses that an event has occurred, the **object associated with the element that caused the event is checked for event handlers**. If that object has a registered handler for the particular event that occurred, that **handler is executed**.
- 依据页码: 223
- 中文回答: 当浏览器感知到事件发生时, 它会**检查与引起事件的元素关联的对象中是否存在事件处理程序**。如果该对象针对特定事件注册了处理程序, 则**执行该处理程序**。

5.19

- 中文翻译: 解释 DOM 2 事件模型中事件处理的三个阶段。
- 英文回答: The first phase is the **capturing phase**, where the event propagates down the document tree to the target node. The second phase is the **target node phase**, where handlers registered at the target node are executed. The third phase is the **bubbling phase**, where the event bubbles back up the document tree to the document node.
- 依据页码: 223
- 中文回答: 第一阶段称为**捕获阶段 (capturing phase)**, 事件沿文档树向下传播到目标节点。第二阶段称为**目标节点阶段 (target node phase)**, 执行在目标节点注册的处理程序。第三阶段是**冒泡阶段 (bubbling phase)**, 事件沿文档树向上返回到文档节点。

5.20

- 中文翻译: 给出两个事件默认行为的例子。
- 英文回答: A mouse click on a link causes the document referenced in the link to **replace the current document**. Also, the default action of a Submit button click is to **submit the form data to the server for processing**.
- 依据页码: 224, 216
- 中文回答: 单击链接会导致浏览器用该链接引用的文档**替换当前文档**。此外,“提交”按钮的默认操作是**将表单数据提交到服务器进行处理**。

5.21

- 中文翻译: 解释 addEventListener 方法的前两个参数。
- 英文回答: The first parameter is the **name of the event as a string literal**. The second parameter is the **handler function**, which could be specified as the function code itself in the form of an anonymous function definition or as the name of a function that is defined elsewhere.
- 依据页码: 224
- 中文回答: 第一个参数是**事件的名称, 作为字符串字面值**。第二个参数是**处理程序函数**, 可以指定为匿名函数定义形式的函数代码本身, 或者指定为在其他地方定义的函数名称。

5.22

- 中文翻译: 如何注册事件处理程序, 使其在捕获阶段被调用?
- 英文回答: An event handler is registered to be called during the capturing phase by setting the **third parameter of addEventListener to true**.
- 依据页码: 224
- 中文回答: 通过将 **addEventListener 方法的第三个参数设置为 true**, 可以注册事件处理程序, 使其在捕获阶段被调用。

5.23

- 中文翻译: 如何取消注册事件处理程序?
- 英文回答: The **removeEventListener method deletes the registration of an event handler**. This method takes the same parameters as addEventListener.
- 依据页码: 224
- 中文回答: 使用 **removeEventListener 方法可以删除事件处理程序的注册**。该方法接受与 addEventListener 相同的参数。

5.24

- 中文翻译: clientX 和 clientY 属性到底存储了什么?
- 英文回答: They store the **x- and y-coordinates of the mouse cursor, relative to the upper-left corner of the client area of the browser window**.
- 依据页码: 225
- 中文回答: 它们存储**鼠标光标的 x 坐标和 y 坐标, 相对于浏览器窗口客户端区域的左上角**。

5.25

- 中文翻译: canvas 元素的目的是什么?
- 英文回答: A **canvas element creates a rectangle into which bit-mapped graphics can be drawn using JavaScript**.
- 依据页码: 228
- 中文回答: **canvas 元素创建了一个矩形, 可以使用 JavaScript 在其中绘制位图图形**。

5.26

- 中文翻译: moveTo 方法到底做了什么?
- 英文回答: The **initial point on the path is specified with a call to moveTo...** This method in effect **moves the drawing pen after raising from the canvas**.
- 依据页码: 229
- 中文回答: **路径的初始点是通过调用 moveTo 来指定的...该方法实际上是在离开画布后移动绘图笔**。

5.27

- 中文翻译: 解释 arc 方法的参数。
- 英文回答: The **arc method is used to draw arcs and circles**. It takes six parameters: the **first two provide the position of the center**; the **third is the radius**; the **fourth is the start angle**; the **fifth is the end angle** (in radians); and the **sixth is a Boolean value specifying whether the drawing is to be clockwise (false) or counterclockwise (true)**.
- 依据页码: 230
- 中文回答: **arc 方法用于绘制弧线和圆**。它接受六个参数: **前两个提供圆心位置**; **第三个是半径**; **第四个是起始角度**; **第五个是结束角度** (以弧度为单位); **第六个是布尔值, 指定绘图是顺时针 (false) 还是逆时针 (true)**。

5.28

- 中文翻译: navigator 对象的目的是什么?
- 英文回答: The **navigator object indicates which browser is being used to view the HTML document**. These properties allow the script to determine which browser is being used and to use processes appropriate to that browser.
- 依据页码: 232
- 中文回答: **navigator 对象指示正在使用哪个浏览器来查看 HTML 文档**。这些属性允许脚本确定正在使用的浏览器, 并使用适合该浏览器的进程。

以下是第6章 (Dynamic Documents with JavaScript, 使用 JavaScript 的动态文档) Review Questions 的回答, 所有内容均基于您提供的资料。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
6.1	定义一个动态 HTML 文档。	A dynamic HTML document is an HTML document whose tag attributes, tag contents, or element style properties can be changed by user interaction or the occurrence of a browser event after the document has been, and is still being, displayed.	240	动态 HTML 文档是指其标签属性、标签内容或元素样式属性可以在文档显示之后且仍在显示期间，通过用户交互或浏览器事件的发生而更改的 HTML 文档。
6.2	如果知道一个 HTML 元素的 ID，如何在 JavaScript 中获取该元素的 DOM 地址？	The approach is to use the JavaScript method getElementById .	200	使用 JavaScript 方法 getElementById。
6.3	如果您有一个存储 HTML 元素 ID 的变量，如何在 JavaScript 中获取该元素的 DOM 地址？	The parameter of getElementById can be any expression that evaluates to a string . In many cases, it is a variable.	200	getElementById 的参数可以是任何求值为字符串的表达式。在许多情况下，它是一个变量。
6.4	还有哪种额外的方式可以获取单个单选按钮和复选框的 DOM 地址？	An alternative... is provided by the implicit arrays associated with each checkbox and radio button group . Every such group has an array... that stores the DOM addresses of the individual buttons in the group .	201	另一种选择是通过与每个复选框和单选按钮组关联的隐式数组提供。每个这样的组都有一个数组，该数组存储了组中各个按钮的 DOM 地址。
6.5	什么是 CSS-P？	CSS-P is an abbreviation for Cascading Style Sheets–Positioning .	241	CSS-P 是层叠样式表-定位（Cascading Style Sheets–Positioning）的缩写。
6.6	描述 position 属性的三个可能值之间的所有区别。	The three possible values are absolute, relative, and static . The absolute value positions the element at a specific location in the display without regard to the positions of other elements . The relative value displaces the element by the specified amount from the position where it would have been placed . The default value static places the element as if it had the position value of relative but no top or left values were given; however, a statically placed element cannot have its top or left properties initially set or changed later .	242, 244, 245, 246	三个可能的值是 absolute、relative 和 static 。 absolute 值指定元素被放置在显示屏上的特定位置，而无需考虑其他元素的位置。 relative 值将元素从其原本应被放置的位置移动指定的量。默认值 static 的放置方式与 relative 相似但无初始位移；但静态定位的元素不能在初始时设置或在之后更改其 top 或 left 属性 。
6.7	visibility 属性的标准值是什么？	The two possible values for visibility are visible and hidden .	249	visibility 属性的两个可能值是 visible 和 hidden 。
6.8	哪些属性控制文档的前景和背景颜色？	The two properties are backgroundColor (for background color) and color (for foreground color).	251	这两个属性是 backgroundColor （用于背景颜色）和 color （用于前景颜色）。
6.9	当鼠标光标移入和移出元素时，可以使用哪些事件来更改字体？	The mouseover event is used when the mouse cursor is moved over an element, and the mouseout event is used when the cursor is moved away from the element.	252	mouseover 事件用于当鼠标光标移到元素上方时触发更改，而 mouseout 事件用于当光标移出元素时触发更改。
6.10	元素的“内容”存储在哪个属性中？	The content of an element is accessed through the value property of its associated JavaScript object.	254	元素的“内容”是通过其关联 JavaScript 对象的 value 属性 访问的。
6.11	与 z-index 属性关联的 JavaScript 变量是什么？	The JavaScript style property associated with the z-index attribute is zIndex .	257	与 z-index 属性关联的 JavaScript 样式属性是 zIndex 。
6.12	要将元素移动到显示屏顶部，是将其 z-index 属性设置为一个大数字还是一个小数字？	An element whose z-index is greater than that of an element in the same space will be displayed over the other element, effectively putting it at the top.	257	z-index 大于 同一空间内其他元素 z-index 的元素将被显示在其他元素之上，从而有效地将其置于顶部。
6.13	鼠标点击后，clientX 和 clientY 属性中精确存储了什么？	They store the coordinates of the element relative to the upper-left corner of the browser display window, in pixels .	261	它们存储了元素相对于浏览器显示窗口左上角的坐标，以像素为单位。
6.14	鼠标点击后，screenX 和 screenY 属性中精确存储了什么？	They store the coordinates of the element relative to the client’s computer screen .	261	它们存储了元素相对于客户端计算机屏幕的坐标。
6.15	描述 setTimeout 函数的参数和作用。	The setTimeout method takes two parameters: a string of JavaScript code to be executed and a number of milliseconds of delay before executing the given code.	265	setTimeout 方法接受两个参数：一个是要执行的 JavaScript 代码字符串，以及在执行给定代码之前延迟的毫秒数。

以下是对第9章复习题的回答，所有信息均来源于提供资料中第9章的内容。

题号	中文问题翻译	英文回答	依据页码	中文回答翻译
9.1	Web服务器如何判断所请求的文档中是否包含PHP代码？	The server determines that a document includes PHP script by the file-name extension . If it is .php, .php3, or .phtml, the document has embedded PHP.	358	服务器通过 文件名扩展名 来确定文档是否包含PHP脚本。如果扩展名是 .php、.php3 或 .phtml，则文档中嵌入了PHP。
9.2	PHP处理器有两种模式，是什么？	The PHP processor has two modes of operation: copy mode and interpret mode .	359	PHP处理器有两种操作模式： 复制模式 （copy mode）和 解释模式 （interpret mode）。
9.3	include构造的语法和语义是什么？	The include construct takes the file name as its string parameter. This construct causes the contents of the file to be copied into the document in which the include appears.	359	include构造接受文件名作为其字符串参数。该构造将文件内容复制到出现include的文档中。
9.4	PHP中哪些名称区分大小写，哪些不区分？	PHP variable names are case sensitive . Although variable names in PHP are case sensitive, neither reserved words nor function names are .	360	PHP变量名区分大小写 。尽管PHP中的变量名区分大小写，但 保留字和函数名则不区分大小写 。
9.5	PHP的四种标量类型是什么？	PHP has four scalar types— Boolean, integer, double, and string .	360	PHP有四种标量类型： 布尔型 （Boolean）、 整型 （integer）、 双精度浮点型 （double）和 字符串型 （string）。
9.6	如何测试变量是否被绑定？	A variable can be tested to determine whether it currently has a value. The test is carried out with the isset function, which takes the variable's name as its parameter and returns a Boolean value.	361	可以使用 isset 函数测试变量是否当前具有值，该函数接受变量名作为参数并返回布尔值。
9.7	如何指定PHP处理器报告未绑定变量的使用情况？	To be informed when an unbound variable is referenced, include a call to the error_reporting function with the parameter value E_ALL .	361	如果想在引用未绑定变量时得到通知，可以在脚本中包含对 error_reporting 函数的调用，参数值为 E_ALL 。
9.8	PHP中存储一个字符需要多少字节？	Characters in PHP are single bytes ; Unicode is not supported.	361	PHP中的字符是单字节 ；不支持Unicode。
9.9	单引号和双引号字符串有什么区别？	In single-quoted string literals , escape sequences, such as \n, are not recognized as anything special and the values of embedded variables are not substituted for their names. In double-quoted string literals , escape sequences are recognized and embedded variables are replaced by their current values.,	361, 362	在 单引号字符串字面值 中，转义序列（如 \n） 不被识别为特殊字符 ，嵌入变量的值也不会被替换。在 双引号字符串字面值 中，转义序列 会被识别 ，嵌入的 变量会被替换 为其当前值。
9.10	如果一个整数表达式出现在布尔上下文中，它的布尔值是如何确定的？	If an integer expression is used in a Boolean context, it evaluates to FALSE if it is zero ; otherwise, it is TRUE .	362	如果一个整数表达式在布尔上下文中使用，如果它的值是 零 ，则评估为 FALSE ；否则，评估为 TRUE 。
9.11	当一个整数算术运算结果的值不能表示为整数时，会发生什么？	Any operation on integers that results in integer overflow also produces a double .	363	任何导致 整数溢出 的整数运算也会产生一个 双精度浮点型 （double）值。
9.12	如果一个变量存储一个字符串，如何引用该字符串中特定位置的字符？	The position of a character in a string, relative to zero, can be specified in braces immediately after the variable's name .	363	字符串中字符的位置（从零开始）可以通过在 变量名后紧跟大括号 来指定。
9.13	chop函数的作用是什么？	The chop function returns the parameter with all white-space characters removed from its end .	364	chop函数返回 移除了其末尾所有空白字符 的参数。
9.14	什么是强制类型转换（coercion）？	Implicit type conversions are called coercions . The context can cause a coercion of the type of the value of the expression.	364	隐式类型转换被称为强制类型转换 （coercion）。上下文可以导致表达式值的类型发生强制转换。
9.15	将变量的值显式转换为特定类型有哪三种方式？	Explicit type conversions can be specified in three different ways: 1. C-style cast (type name in parentheses preceding the expression); 2. intval, doubleval, or strval functions ; 3. settype function.,	365	显式类型转换可以通过三种方式指定：1. C语言风格的类型转换 （在表达式前加上括号内的类型名）；2. 使用 intval、doubleval 或 strval 函数 ；3. 使用 settype 函数 。
9.16	如何确定变量的类型？	The type of the value of a variable can be determined in two different ways. The first is the gettype function ... The other way to determine the type of the value of a variable is to use one or more of the type-testing functions (e.g., is_int).	365	确定变量类型有两种方式。第一种是使用 gettype函数 ，它返回表示变量当前类型名称的字符串。另一种是使用一个或多个 类型测试函数 （例如 is_int），它们返回布尔值。
9.17	如果字符串与数字进行比较，会发生什么？	If a string is compared with a number and the string can be converted to a number, the string will be converted and a numeric comparison will be done . If the string cannot be converted to a number, the numeric operand will be converted to a string and a string comparison will be done .	368	如果 字符串与数字进行比较 ，并且该字符串可以转换为数字，则 字符串将被转换并进行数值比较 。如果该字符串不能转换为数字，则 数字操作数将被转换为字符串并进行字符串比较 。
9.18	与独有的关闭保留字（如 endwhile）相比，使用通用的关闭保留字（如 end）的优点是什么？	<i>The provided source material does not contain sufficient information to directly compare the advantages of generic vs. unique closing reserved words (like end vs. endwhile in alternative PHP syntax).</i>	N/A	提供的来源材料中没有足够的信息来直接回答该问题。
9.19	PHP中创建数组有哪两种方式？	There are two ways to create an array in PHP: 1. The assignment operation ... Assigning a value to a	371, 372	PHP中创建数组有两种方式：1. 赋值操作 ，即给先前不是数组的带下标变量赋值；2. 使用 array 构造 。

题号	中文问题翻译	英文回答	依据页码	中文回答翻译
		subscripted variable that previously was not an array creates the array. 2. With the array construct ,		
9.20	当创建数组时没有指定键时，会使用哪些键？	If empty brackets are used in an assignment to an array, the furnished subscript is 1 greater than the largest numeric key used so far in the array . If the array currently has no elements with numeric keys, the value 0 is used .	372	如果在对数组赋值时使用了空方括号，该下标是 迄今为止在数组中使用过的最大数字键值加1 。如果数组当前没有数字键元素，则使用值 0 。
9.21	数组中的所有值都必须是同一类型吗？	No. The elements of an array need not be of the same type .	372	不必。数组的元素 不必是相同的类型 。
9.22	数组中的所有键都必须是同一类型吗？	No. PHP arrays... can be mixtures of both [integer and string keys].	372	不必。PHP数组可以是整数键和字符串键的 混合形式 。
9.23	array_keys和array_values函数的作用分别是什么？	The array_keys function takes an array as its parameter and returns an array of the keys of the given array. The array_values function does for values what array_keys does for keys.	373	array_keys 函数接受一个数组作为参数，并返回该数组的 键数组 。 array_values 函数则返回 值数组 。
9.24	in_array函数的作用是什么？	The in_array function takes two parameters—an expression and an array—and returns TRUE if the value of the expression is a value in the array ; otherwise, it returns FALSE.	374	in_array 函数接受一个表达式和一个数组，如果 表达式的值是数组中的一个值 ，则返回 TRUE ；否则返回FALSE。
9.25	解释implode和explode函数的作用。	The explode function explodes a string into substrings and returns them in an array . The implode function... catenates the elements of the array together , placing the given separator string between the elements, and returns the result as a string.,	374	explode 函数将一个 字符串分解成子字符串并以数组形式返回 。 implode 函数将数组的元素 连接成一个字符串 ，并在元素之间放置给定的分隔符，并返回结果字符串。
9.26	描述next、reset和prev函数的作用。	The next function both moves the pointer to the next array element and returns the value... The reset function sets the current pointer to the first element of the array and returns the value... The prev function moves the current pointer backward (to the element before the current element).,	375, 376	next 函数移动指针到 下一个数组元素 并返回该元素的值。 reset 函数将当前指针设置到数组的 第一个元素 并返回该元素的值。 prev 函数将当前指针 向后 （移到当前元素之前）移动。
9.27	两种形式的foreach语句的语法和语义是什么？	Form 1: foreach (array as scalar_variable) loop body . One of the array's values is set to the scalar variable for each iteration. Form 2: foreach (array as key => value) loop body . Provides both the key and the value of each element.,	377	形式 1: foreach (array as scalar_variable) loop body 。在每次迭代中，数组的一个值被设置为标量变量。 形式 2: foreach (array as key => value) loop body 。提供数组中每个元素的键和值。
9.28	描述对一个同时包含字符串和数值的数组使用sort函数的结果。	The string values migrate to the beginning of the array in alphabetical order. The numeric values follow in ascending order.	377	字符串值 按字母顺序 迁移到数组的开头 。 数值值 紧随其后，按升序排列。
9.29	sort和asort函数有什么区别？	The sort function sorts the values in the array, replacing the keys with the numeric keys (0, 1, 2...). The asort function sorts the elements of a given array by their values, but keeps the original key-value associations .,	377	sort 函数对值进行排序， 并替换为数字键 。 asort 函数按值对元素进行排序，但 保留原始的键值关联 。
9.30	如果一个脚本多次定义同一个函数会发生什么？	If a second definition of a function appears in a script, it is reported as an error because function overloading is not allowed and functions cannot be redefined.	380	如果脚本中出现函数的第二次定义，这 将被报告为错误 ，因为不允许函数重载，且函数不能被重新定义。
9.31	函数名是否区分大小写？	Function names are not case sensitive .	380	函数名不区分大小写 。
9.32	如果一个函数在执行过程中没有执行return语句，它会返回什么值？	If no return statement was executed, no value is returned .	380	如果没有执行return语句，则 不返回值 。
9.33	指定引用传递参数的两种方式是什么？	Pass-by-reference parameters can be specified in two ways: 1. By adding an ampersand (&) to the beginning of the name of the formal parameter . 2. By adding an ampersand to the actual parameter in the function call .	381	指定引用传递参数有两种方式：1. 在 形参名称的开头添加一个和号(&) 。2. 在 函数调用中的实参处添加一个和号 。
9.34	如何让函数访问在函数外部定义的变量？	When a variable is listed in a global declaration in a function, that variable is expected to be defined outside the function.	382	当变量在函数中被列入 global 声明 时，该变量被认为是 在函数外部定义的 。
9.35	如何在一个函数中定义变量，使其生命周期超出该函数第一次执行的时间？	A local variable in a function can be specified to be static by declaring it with the reserved word static .	383	可以通过使用保留字 static 声明函数中的局部变量，从而使其生命周期得以延长。
9.36	PHP脚本如何访问表单元素的值？	The recommended approach is to use the implicit arrays \$_POST and \$_GET for form values . These arrays have keys that match the form element names and the values that were input by the client.	386	推荐的方法是使用隐式数组 \$_POST 和 \$_GET 来获取表单值。这些数组的键与表单元素名称匹配，值则是客户端输入的值。
9.37	如何在PHP脚本中创建cookie？	A cookie is set in PHP with the setcookie function . The first parameter, which is mandatory, is the cookie's name given as a string.	393	cookie是通过 setcookie 函数 设置的。第一个（强制性）参数是以字符串形式给出的cookie名称。

题号	中文问题翻译	英文回答	依据页码	中文回答翻译
9.38	脚本如何确定特定cookie是否存在？	All cookies that arrive with a request are placed in the implicit \$_COOKIES array . A PHP script can test whether a cookie came with a request by using the isset predicate function on the associated variable.	394	所有随请求到达的cookie都放在隐式数组 \$_COOKIES 中。PHP脚本可以通过对关联变量使用 isset 谓词函数来测试该cookie是否随请求而来。
9.39	如何将会话中（session）的变量保存下来？	Session variables are stored in the \$_SESSION array .	395	会话变量存储在 \$_SESSION 数组中。

好的，根据您提供的第13章内容，我将回答Review Questions中可见的问题（13.1至13.41）。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
13.1	关系数据库中表的主键的用途是什么？	Most tables have one column for special data values that uniquely identify the rows of the table .	560	大多数表都有一个用于存储特殊数据值的列，这些值可以 唯一标识表中的行 。
13.2	交叉引用表的作用是什么？	A cross-reference table contains no data; instead, it contains the primary keys of two data tables, providing a many-to-many relationship between the data in the two tables .	595	交叉引用表不包含数据；相反，它包含 两个数据表的主键，提供两个表之间数据的多对多关系 。
13.3	SQL 中字符串文字是如何定界的？	Apostrophes (') are used to delimit character strings .	563	撇号 (') 用于界定字符串 。
13.4	SQL CREATE TABLE 命令中的 NOT NULL 约束指定了什么？	NOT NULL means that every row in the table must have a value in a column that has this constraint.	567	NOT NULL 意味着表中的每一行在该受此约束的列中都必须有一个值。
13.5	当星号作为 SELECT 子句的值出现时，它指定了什么？	An asterisk (*) as the SELECT clause value indicates the selection of all the columns of the specified table of the rows that meet the condition specified in the WHERE clause.	563	作为 SELECT 子句值的 星号 (*) 表示选择指定表中符合 WHERE 子句中规定的条件的所有行中的 所有列 。
13.6	SELECT 命令的 WHERE 子句指定了什么？	The WHERE clause is used to specify constraints on the rows of the specified tables that are of interest.	563	WHERE 子句 用于 指定对 所关注的指定表的 行施加的约束 。
13.7	在 INSERT 命令中，列名是如何与值关联的？	The correspondence between the column names and the values is positional: The first value goes into the column that is named first, and so forth .	565	列名和值之间的对应关系是 按位置的：第一个值进入第一个命名的列中，依此类推 。
13.8	UPDATE 命令的目的是什么？	The UPDATE command is used to change one or more of the values of a row of a table .	565	UPDATE 命令 用于 更改表中一行的一个或多个值 。
13.9	表连接到底是什么？在 SQL 中如何指定表连接？	A join operation creates a new table by joining part of the data of one table with part of the data of another table . A join is specified with a SELECT command that has two tables named in the FROM clause and that uses a compound WHERE clause .	596, 564	连接 操作通过将 一个表的部分数据与另一个表的部分数据连接起来 ，创建一个新表。连接是用一个 SELECT 命令指定的，该命令的 FROM 子句中命名了两个表，并且使用了复合的 WHERE 子句 。
13.10	在用于 Web 访问数据库的客户端-服务器配置中，第三层的作用是什么？	The third tier in the system has the database server and the database itself .	568	系统的第三层有 数据库服务器和数据库本身 。
13.11	为什么两层客户端-服务器配置有时不足够？	Two-tier systems suffer problems such as the server capacity to deal with increasing application complexity and the difficulty of keeping the clients current with changes in the applications that use the database.	568	两层系统存在一些问题，例如服务器处理日益复杂的应用程序的能力，以及如果应用程序的某些部分移到客户端，则难以 使客户端与使用数据库的应用程序的更改保持同步 。
13.12	解释如何通过扩展编程语言来提供 SQL 数据库访问。	A method is to extend a general-purpose programming language so that it can specify SQL commands and interact with a database through those commands.	596	一种方法是 扩展通用编程语言，使其能够指定 SQL 命令 并通过这些命令与数据库进行交互。
13.13	在编程语言中嵌入 SQL 的缺点是什么？	The disadvantage of this approach is that such applications are not likely to be portable among the databases of different vendors .	596	这种方法的缺点是，此类应用程序 不太可能在不同供应商的数据库之间移植 。
13.14	什么是 ODBC，它有什么用？	ODBC specifies an API for a set of objects and methods that serves as an interface to different databases . By using ODBC, an application can include SQL statements that work with any database for which a driver has been installed.	568	ODBC 为一组对象和方法指定了 API，作为不同数据库的接口 。通过使用 ODBC，应用程序可以包含适用于已安装驱动程序的任何数据库的 SQL 语句。
13.15	ODBC 和 JDBC 之间有什么关系？	JDBC is similar to ODBC , at least in terms of purpose. Both have the X/OPEN SQL Call Level Interface (SQL CLI) in their heritages.	569, 596	JDBC 至少在目的上与 ODBC 相似 。两者都有 X/OPEN SQL Call Level Interface (SQL CLI) 的继承性。
13.16	什么是 MySQL？	MySQL is a free, efficient, widely used database system that implements SQL .	569	MySQL 是一个免费、高效、广泛使用的数据库系统，它实现了 SQL 。
13.17	MySQL 的 auto_increment 约束有什么作用？	AUTO_INCREMENT specifies that the values of this column need not be given in populating the table. The values 1, 2, 3, and so forth will be implicitly assigned.	571	AUTO_INCREMENT 指定在填充表时 不需要给出该列的值 。这些值（如 1、2、3 等）将被隐式分配。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
13.18	从 HTML 文档中的表单元 素获取的 SQL 命令中的引 号有什么问题？	If the PHP implicit function <code>magic_quotes_gpc</code> is enabled, values received from POST, GET, and COOKIE have backslashes implicitly inserted in front of all single quotes , double quotes, backslashes, and null characters. This resulting string is not a valid SQL command and will be rejected by MySQL.	572– 573	如果 PHP 隐式函数 <code>magic_quotes_gpc</code> 启用，则从 POST、GET 和 COOKIE 接收到的值会在 所有单引号、双引号、反斜杠和空字符前面隐式插入反斜杠 。得到的字符串 不是有效的 SQL 命令 ，将被 MySQL 拒绝。
13.19	PHP <code>mysqli_select_db</code> 函数的目的是什么？	The function <code>mysqli_select_db</code> is used to set the default database (if it was left out of the <code>mysqli_connect</code> call).	573	<code>mysqli_select_db</code> 函数用于 设置默认数据库 （如果 <code>mysqli_connect</code> 调用中省略了该名称）。
13.20	PHP 程序如何确定查询结 果中的行数？	The number of rows in the result is obtained with the <code>mysqli_num_rows</code> function , which is passed the result returned by <code>mysqli_query</code> .	574	结果中的行数是通过 <code>mysqli_num_rows</code> 函数 获得的，该函数会传递 <code>mysqli_query</code> 返回的结果。
13.21	PHP 函数 <code>mysql_fetch_array</code> 有什 么作用？	(<i>根据来源中介绍的</i> <code>mysqli_fetch_assoc</code> <i>回答</i>) <code>mysqli_fetch_assoc</code> returns an associative array of the next row of the result.	574	<code>mysqli_fetch_assoc</code> 返回结果的下一行的关联数组 。
13.22	解释 <code>mysql_fetch_array</code> 返回值的确切形式。	(<i>根据来源中介绍的</i> <code>mysqli_fetch_assoc</code> <i>回答</i>) The associative array has the column names as keys and the values are the values of those column fields.	574	关联数组以 列名作为键，值 则是这些列字段的值。
13.23	解释使用 JDBC 的两种方 式。	(<i>资料未明确给出两种方式，但提供了两种执行 SQL 的方式</i>) JDBC can be used to execute action commands (e.g., INSERT, UPDATE, DELETE) through the <code>executeUpdate</code> method or to execute a query command (SELECT) through the <code>executeQuery</code> method.	582	JDBC 可用于通过 <code>executeUpdate</code> 方法执行 操作命令 （例如 INSERT、UPDATE、DELETE）或通过 <code>executeQuery</code> 方法执行 查询命令 （SELECT）。
13.24	使用 JDBC 时，第三层计算 机提供了什么优势？	The third tier in the system has the database server and the database itself .	568	系统的第三层有 数据库服务器和数据库本身 。
13.25	使用 JDBC 时，使用哪个类 中的哪个方法连接到数据 库？	The connection is made by creating a <code>Connection</code> object with the <code>getConnection</code> method of the <code>DriverManager</code> class .	581	通过使用 <code>DriverManager</code> 类 中的 <code>getConnection</code> 方法 创建 <code>Connection</code> 对象来连接数据库。
13.26	解释注册 JDBC 驱动程序的 两种方法。	资料中未包含此信息。	N/A	资料中未包含此信息。
13.27	通过 JDBC 使用 SQL 时， Statement 对象有什么作 用？	The <code>Statement</code> object is created through which one of the Statement methods can actually issue the command .	581	<code>Statement</code> 对象 的创建是为了通过其方法之一 实际发出命令 。
13.28	使用哪个类中的哪个方法 执行 SQL 操作命令？	Action commands are executed through the <code>executeUpdate</code> method of the <code>Statement</code> object .	582	操作命令是通过 <code>Statement</code> 对象 的 <code>executeUpdate</code> 方法 执行的。
13.29	使用哪个类中的哪个方法 执行 SELECT 命令？	A SELECT SQL command is executed by passing it as the parameter to the <code>executeQuery</code> method of the <code>Statement</code> object .	582	SELECT SQL 命令是通过将其作为参数传递给 <code>Statement</code> 对象 的 <code>executeQuery</code> 方法 来执行的。
13.30	<code>executeQuery</code> 方法返回什 么类的对象？	The return value from the execution of a SELECT command is an object of <code>ResultSet</code> type .	596	执行 SELECT 命令的返回值是 <code>ResultSet</code> 类型的对象 。
13.31	程序如何迭代 <code>executeQuery</code> 返回的对 象？	The elements of a <code>ResultSet</code> object are accessed through an iterator method named <code>next</code> . The <code>next</code> method returns a Boolean value depending on whether there is another element in the <code>ResultSet</code> object.	582	<code>ResultSet</code> 对象是通过名为 <code>next</code> 的迭代器方法访问其元素的。 <code>next</code> 方法根据 <code>ResultSet</code> 对象中是否还有其他元素来返回布尔值。
13.32	什么是元数据？	Metadata is information that describes the database itself or some part of the database .	583	元数据 是 描述数据库本身或数据库某些部分的信息 。
13.33	如何从数据库中提取元数 据集合？	The <code>getMetaData</code> method of the <code>Connection</code> object creates an object of <code>DatabaseMetaData</code> type , which can be used to get information about a database.	583	<code>Connection</code> 对象 的 <code>getMetaData</code> 方法 会创建一个 <code>DatabaseMetaData</code> 类型 的对象，该对象可用于获取有关数据库的信息。
13.34	有哪两种方式可以从元数 据对象中获取列标签？	The name of the <i>i</i> th column is returned by <code>getColumnLabel(i)</code> . Alternatively, the data can be converted to strings with <code>getString</code> and then retrieved by column name.	584, 585	第 <i>i</i> 列的名称是通过调用 <code>getColumnLabel(i)</code> 返回的。另外，数据可以由 <code>getString</code> 转换成字符串，然后通过列名获取。
13.35	什么是 SQL Server？	资料中未包含此信息。	N/A	资料中未包含此信息。
13.36	ADO.NET 的设计理念是什 么？	The fundamental aim of ADO.NET is to provide a relationship between markup controls and some data source , in either the program or external to it.	589	ADO.NET 的基本目标是提供 标记控件与程序内部或外部某些数据源之间的关系 。
13.37	支持 ADO.NET 连接部分的 三个类是什么？	The three kinds of classes are those for connections , those for commands , and those for data readers .	589	这三种类是用于 连接 的类、用于 命令 的类以及用于 数据读取器 的类。
13.38	什么是 ASP.NET 数据绑定 控件？	Data-bound controls are server controls where ADO.NET fetches data from a data source and dynamically associates that data with the properties of the controls .	590	数据绑定控件 是服务器控件，ADO.NET 从数据源中获取数据，并将 该数据动态地与控件的属性关联起来 。
13.39	连接字符串中包含哪些信 息？	The connection string includes information about the driver to use, the server, the specific database, the user id, and, possibly, a database password .	591	连接字符串包括有关 要使用的驱动程序、服务器、特定数据库、用户 ID，以及可能的数据库密码 的信息。

序号	中文问题翻译	英文回答	依据页码	中文回答翻译
13.40	GridView 控件类的 DataBind 方法有什么作用？	The <code>DataBind</code> method moves data from the data source to populate the control .	590, 592	<code>DataBind</code> 方法 将数据从数据源移到控件中以 填充控件 。
13.41	哪个属性用于在 ADO.NET 中将数据源对象与数据绑定控件关联起来？	The <code>DataSource</code> property of a data-bound control specifies the data source object to which the control is bound.	590, 592	<code>DataSource</code> 属性 指定了与数据绑定控件关联的数据源对象。