1. 块元素一般都从新行开始，它可以容纳[内联元素](http://www.so.com/s?q=%E5%86%85%E8%81%94%E5%85%83%E7%B4%A0&ie=utf-8&src=se_lighten_f)和其他块元素,常见块元素是段落标签'P"。“form"这个块元素比较特殊，它只能用来容纳其他块元素。
2. 如果没有css的作用，块元素会顺序以每次另起一行的方式一直往下排。而有了css以后，我们可以改变这种html的默认布局模式，把块元素摆放到你想要 的位置上去。而不是每次都愚蠢的另起一行。需要指出的是，table标签也是块元素的一种，table based layout和css based layout从一般使用者（不包括视力障碍者、盲人等）的角度来看这两种布局，除了页面载入速度的差别外，没有其他的差别。但是如果普通使用者不经意点了 查看页面[源代码](http://www.so.com/s?q=%E6%BA%90%E4%BB%A3%E7%A0%81&ie=utf-8&src=se_lighten_f)按钮后，两者所表现出来的差异就非常大了。基于良好重构理念设计的css布局页面源码，至少也能让没有web开发经验的普通使用者把内容快 速的读懂。从这个角度来说，css layout code应该有更好的美学体验吧。
3. 你能够把块容器元素div想象成一个个box，或者如果你玩过剪贴文载的话，那就更加容易理解了。我们先把需要的文章从各种报纸、杂志总剪 下来。每块剪下来的内容就是一个block。然后我们把这些纸块按照自己的排版意图，用胶水重新贴到一张空白的新纸上。这样就形成了你自己独特的文摘快报 了。作为一种技术的延伸，网页布局设计也遵循了同样的模式。.
4. 内联元素(inline element)一般都是基于语义级(semantic)的基本元素。内联元素只能容纳文本或者其他内联元素，常见内联元素 “a”。
5. 块元素(block element)和内联元素(inline element)都是html规范中的概念。块元素和内联元素的基本差异是块元素一般都从新行开始。而当加入了css控制以后，块元素和内联元素的这种属 性差异就不成为差异了。比如，我们完全可以把内联元素cite加上display:block这样的属性，让他也有每次都从新行开始的属性。
6. 可变元素的基本概念就是他需要根据上下文关系确定该元素是块元素或者内联元素。可变元素还是属于上述两种元素类别，一旦上下文关系确定了他的类别，他就要遵循块元素或者内联元素的规则限制。大致的元素分类见全文。
7. 关于inline element的中文叫法，有多种内联元素、内嵌元素、行内元素、直进式元素。基本上没有统一的翻译，爱怎么叫怎么叫吧。另外提到内联元素，我们会想到有个display的属性是display:inline;这个属性能够修复著名的IE双倍浮动边界问题。
8. 块元素(block element)
9. \* address - 地址
10. \* blockquote - 块引用
11. \* center - 举中对齐块
12. \* dir - 目录列表
13. \* div - 常用块级容易，也是css layout的主要标签
14. \* dl - 定义列表
15. \* fieldset - form[控制组](http://www.so.com/s?q=%E6%8E%A7%E5%88%B6%E7%BB%84&ie=utf-8&src=se_lighten_f)
16. \* form - 交互表单
17. \* h1 - 大标题
18. \* h2 - 副标题
19. \* h3 - 3级标题
20. \* h4 - 4级标题
21. \* h5 - 5级标题
22. \* h6 - 6级标题
23. \* hr - 水平分隔线
24. \* isindex - input prompt
25. \* menu - 菜单列表
26. \* noframes - frames可选内容，（对于不支持frame的浏览器显示此区块内容
27. \* noscript - 可选脚本内容（对于不支持script的浏览器显示此内容）
28. \* ol - 排序表单
29. \* p - 段落
30. \* pre - 格式化文本
31. \* table - 表格
32. \* ul - 非排序列表
34. 内联元素(inline element)
36. \* a - 锚点
37. \* abbr - 缩写
38. \* acronym - 首字
39. \* b - 粗体(不推荐)
40. \* bdo - bidi override
41. \* big - 大字体
42. \* br - 换行
43. \* cite - 引用
44. \* code - 计算机代码(在引用源码的时候需要)
45. \* dfn - 定义字段
46. \* em - 强调
47. \* font - 字体设定(不推荐)
48. \* i - 斜体
49. \* img - 图片
50. \* input - 输入框
51. \* kbd - 定义键盘文本
52. \* label - 表格标签
53. \* q - 短引用
54. \* s - 中划线(不推荐)
55. \* samp - 定义范例计算机代码
56. \* select - 项目选择
57. \* small - 小字体文本
58. \* span - 常用内联容器，定义文本内区块
59. \* strike - 中划线
60. \* strong - 粗体强调
61. \* sub - 下标
62. \* sup - 上标
63. \* textarea - 多行文本输入框
64. \* tt - 电传文本
65. \* u - [下划线](http://www.so.com/s?q=%E4%B8%8B%E5%88%92%E7%BA%BF&ie=utf-8&src=se_lighten_f)
66. \* var - 定义变量
68. 可变元素
70. 可变元素为根据上下文语境决定该元素为块元素或者内联元素。
71. \* applet - java applet
72. \* button - 按钮
73. \* del - 删除文本
74. \* iframe - inline frame
75. \* ins - 插入的文本
76. \* map - 图片区块(map)
77. \* object - object对象
78. \* script - 客户端脚本