in test 2 - to Feb 11, 2009

1. rtyjertgj rthkty

uyklyui;luio;uio;iuo;rtth rwherghergh

Amount			
CLI : Gender	No- 13c.Poor practis..	Yes- 13c.Poor practi..	Grand Total
Female	1	2	3
Male	4		4
Grand Total	5	2	7

2. k,kkkkkkkkkkkk

The 802.11ac WiFi standard may sound like an alphabetical step backwards, but for high-bandwidth tasks like 1080p streaming it promises to wipe the face off 802.11n. Qualcomm Atheros wants its share of the billion unit pie and has just launched a series of products to flesh out its 802.11ac ecosystem. Top billing goes to the WCN3680 WiFi/BlueTooth/FM combo module, which plugs into the new Snapdragon S4 (MSM8960) and offers speeds of up to 433Mbps to complement that blistering CPU performance. Since the S4 already includes built-in b/g/n WiFi (not to mention its 3G/4G/LTE baseband), manufacturers who choose to add the 802.11ac component will achieve full WiFi cross-compatibility and make many consumers happy in the process. Meanwhile, you'll also find similar multilingual abilities in QA's other 5G WiFi modules for PCs, laptops, routers and enterprise, which are all detailed in the PR after the break. Rest assured that we'll bring you more hands-on impressions of the latest Snapdragon just as soon as things kick off at MWC -- and hopefully in the form of a finished, market-ready tablet or handset.

3. mtyjmtjtyjmyjm

currentTarget

As we've seen earlier, an event has a `target` or `srcElement` that contains a reference to the element the event happened on. In our example this is element2, since the user clicked on it.

It is very important to understand that during the capturing and bubbling phases (if any) this target does not change: it always remains a reference to element2.

But suppose we register these event handlers:

```
element1.onclick = doSomething;<br />
element2.onclick = doSomething;<br />
```

If the user clicks on element2 `doSomething()` is executed twice. But how do you know which HTML element is currently handling the event? `target/srcElement` don't give a clue, they always refer to element2 since it is the original source of the event.

To solve this problem W3C has added the `currentTarget` property. It contains a reference to the HTML element the event is currently being handled by: exactly what we need. Unfortunately the Microsoft model doesn't contain a similar property.

You can also use the `this` keyword. In the example above it refers to the HTML element the event is handled on, just like `currentTarget`.

4.