GET v POST

This was a question asked in the LeadsToYou interview, which I didn’t answer as well as I could have.

What I didn’t say clearly is that as the names imply GET is primarily for retrieving/requesting data and POST for sending/submitting. What I did say is that with POST the user data is in the body of the request and in GET is part of the URL. What I also said correctly is that whilst GET may be seen as less secure because parameter names are exposed, POST is not inherently any more secure than GET, assuming you are not so stupid as to pass sensitive data via the URL. In fact, GET should ‘not be used for operations that cause side-effects’ but only to ‘request a representation of the specified resource’.

Other differences:

* GET can be bookmarked
* GET limits what can be sent in ways that POST doesn’t, in both the amount of data (limit of query string length) and type of data (POST can send binary data i.e. files)
* browsers issue a warning on re-submission of POST requests (because by definition a POST request is going to be making some change, whereas GET is just a request for information with no side-effects)
* GET requests can be cached (which has implication when using them for AJAX calls)

(What’s the difference between POST and PUT? PUT is for altering an existing object).

Others:

HEAD is GET without a body…