Marcel Englmaier
CS 3500 Test

1.a Username of the person who's password is in question
1.b password field(but not actually containing a password, just an 'x' for security)
1.c userid as a number so user can be found by number not username
1.d group id as a number so group can be found by number not groupname
1.e full name of user
1.f user's home directory
1.g user's shell account so the user can access the bash shell

2. it is stored in the /etc/shadow file, and I'm certain it's encrypted

3. a home directory is made for the user

4. definitely something with "sudo" in it. We never needed it nor was it required for the website, so I don't actually know

5. the/etc/sudoers file

6. you can specify the commands a group can run in the sudoers file. So, you create a group, make that the group fred is in, and specify that that group can only run barny. I don't know the exact code or text for this(I'd need the internet for that), but that's the basics of doing it.

7. I don't really know, as I don't really need this. If I'd like to update apache, it would be "$sudo yum update httpd". That would take care of version issues. If I'd have to guess, I'd say "$ httpd --version"

8. sudo yum install httpd mod_ssl

9. It is the variable that contains the locations of all the places where executables reside so that's where the shell can look for the executable. $echo PATH

10. "$ls –l" will give you all the permissions for all the files in the directory. You can also specify a specific file by adding the filename to the end of that command

11.It means that root has read write execute functions, but the group and the user only have reading and executing permissions. "$chmod 755 <filename>" should do the trick

12. From what my group did, this file: "/etc/httpd/conf/httpd.conf" and the iptables file to make port 80 work

13./var/log/httpd

14.through the shell. I did not include specific commands on the web page because this was unneeded, but you generate a private key for your machine, and a public key on your server. Then you simply copy the key files from one machine to the other and vice versa

15. A public key is the key the server generates that is distributed to everyone. A private key is the key your local machine generates that is used. Everybody can get the public key, but you have to give your private key to somebody for them to have it.

16. I think it can, as you could send the key to the connecting machine, but I think that would be stupid because I wouldn't trust that service to be properly secure

17. no password logins

18.a. cleaner code
18.b. legacy support across browsers
18.c. audio/video support

19. via a <link> with an href to the .css file it needs. To get the actually specifications, there is a tag at the top of the html page that you can link to the specification that you want. Most I have seen have been to w3schools.com. I don't have the actual code because we don't need and therefore don't use it.

20.a. Formatting across multiple files by only changing one
20.b. In-depth formatting that html can't provide
20.c. layered formatting for tags within tags make much easier

21.Yes. That's why it's so nice to use(competition for Adobe Flash).