1. A backend server is essentially a program or device that provides functionality to websites and hosts websites (so they can be viewed on the internet). A backend server is a centralized mechanism that allows websites to have functionality by running said backend functionality on the server rather than on the website itself through the html and javascript. Backend servers provide functionality that is different from front-end because as front-end is loaded after being opened (so browser reads html, then javascript and other attached files) while back-end components are already “there” before the website is even read by the browser.

a) <http://www.codeconquest.com/website/client-side-vs-server-side/> was a good resource to use. I like it because it provided the basic information and has access to free tutorials on how to use the server side languages like PHP and Ruby on Rails.

b) <http://www.webopedia.com/TERM/D/database_server.html> was a good resource to use because it has a lot of basic definitions and explanations for many terms used in computing. It’s a convenient resource to use when one wants to find out information quickly about a term.

c) <https://www.simplilearn.com/cloud-computing-architecture-article> was a good resource as it not only offered basic information but also certification courses pertaining to computer programming that could be good for individuals looking for work in the computing industry.

2. An SQL database is a mechanism for storage/retrieval of information that is tabular based. NoSQL databases are not tabular based but rather document based and involved wide-column stores. SQL is seen as “vertically scalable” with predefined schemas for more organization while NoSQL is “horizontally scalable” with dynamic schema that is useful for unstructured data. One may be preferable over the other based on the needs of the client/business so this would involve availability of resources to increase the database capability of either type of system at the given time.

a) <https://www.w3schools.com/sql/sql_intro.asp> is a good resource because it walks through a lot of the basic information and has many categories to choose from that are related to SQL (w3 schools in general I have found to be a very good resource for information as the explanations are straight forward and many contain actual examples with code to work with and see run).

b) <http://www.thegeekstuff.com/2014/01/sql-vs-nosql-db/?utm_source=tuicool> was a good resource to look at because it explained in detail the difference between SQL and NoSQL database systems and went through the main points of each currently available type (MySQL, Oracle Express, etc.).

3. A promise is similar to an event listeners in javascript except that a promise can only succeed or fail once and the correct callback can be called in a later time (assuming a success/failure callback has been added in) even with the event taking place earlier on in time.

3 resources:

a) <https://developers.google.com/web/fundamentals/getting-started/primers/promises>

b) <https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/Promise>

c) <http://web-engineering.info/node/59>

4. <https://git-scm.com/about> and <https://help.github.com> are both good resources that can be used to understand git merges. Visual studio code is currently the editor I use to write all of my files and it works through a basic scheme of text, editing, saving, and exiting as with any other general program.