Section 2 Quiz

The core technology that your browser uses to fetch and display data to your phone or computer, is HTTP. In other words, HTTP allows for communication between a browser and a server. The communication itself usually takes place over TCP/IP, but any reliable transport can be used.

Every HTTP interaction includes a request and a response. Every request must have a URL address and a method (like POST or GET).

We’ve also seen that HTTP is stateless, meaning that each request is separate from each other. This means that each request that your browser makes must contain enough information on its own for the server to fulfill the request.

You've learnt a lot already (pat yourself on the back), and we're only just getting warmed up.

Grab a coffee, and try these few questions :)

# Question 1:

Do Browsers have to comply with a common set of W3C standards?

1. No
2. Yes

## Answer:

A - Correct. Browsers are not legally obliged to follow any set of standards. This was part of the problem in the early days of the web, where compatibility was limited to developers that could afford to continuously update and refactor their websites for each different Browser. This is partly why web standards were created - to help solve the problem of cross browser compatibility issues.

# Question 2:

Why do Browsers decide to follow the W3C standards?

1. Because they are legally obligated to do so.
2. Because it allows them to create faster web page loading times and gives them an 'edge' over their competitors.
3. Browsers decide to follow several W3C standards because it allows them to interpret the latest versions of HTML and CSS code.

## Answer:

C - When browsers conform to the W3C standards, it also helps web pages appear consistent across different browsers.

# Question 3:

What is HTTP?

1. HTTP stands for HyperText Transfer Protocol.
2. HTTP stands for HighText Transfer Protocol and is a programming language, allowing us to write very powerful web applications.
3. HTTP stands for HyperTransfer Text Programming and is a method of programming allowing us to develop single web page applications.

Answer:

A - This is the starting point for data communication between devices over the internet. The data communication starts with a request sent from a client (aka: your browser) and ends with the response received from a web server.

# Question 4:

What kind of information does a HTTP Request sent from a client contain?

1. It consists of only header information, such as accept-language, accept, authority, method etc.
2. It will consist of the following:
   1. A Request line to get a required resource, for example a request GET/content/example.html is requesting a resource called /content/example.html from the server.
   2. Headers (e.g. accept-language, content-type, accept, etc.).
   3. A Message Body which is optional.
3. It only consists of the method (such as GET, POST, DELETE, etc.), which tells the server what we want it to do with the information.

## Answer: B

# Question 5:

In client-server protocols, like HTTP, what does a typical session look like?

1. The server establishes a TCP connection.

The client sends its request, and waits for the answer.

The server processes the request, sending back its answer, providing a status code and appropriate data.

1. The client sends a request.

A connection is then opened (usually via TCP but it doesn't have to be).

The client then receives a response immediately.

1. The client establishes a connection (usually the transport layer uses TCP).

The client sends its request, and then chills out for a response.

The server processes the request, sending back its answer (the server will also send back other data like status codes).

## Answer:

C - In client-server protocols, it is the client which establishes the connection. Opening a connection in HTTP means initiating a connection in the underlying transport layer, usually this is TCP but it doesn't have to be.

# Question 6:

Why are browser requests so important to know about and master?

1. Because it forms the basis of pretty much everything we do. When you are browsing the web, you are actually calling up a list of requests to get content from various resource directories or servers on which the content for that page is stored. It can be compared to a recipe for a cookie - you have a shopping list of ingredients (requests for content) that when combined in the correct order, these ingredients bake a delicious cookie (the web page).
2. Because it allows us to write code more efficiently and ensure that our web page is displayed in the quickest possible time.
3. Because we can then use cookies to store the results of various requests.

## Answer:

A - Great stuff. Because a request forms such a large part of what we do, understanding it will make you a better programmer.

# Question 7:

What role do packets have when you want to visit a web page?

1. A packet is one large chunk of all the data that needs to be carried over a network.
2. Packets help you get the content of a web page you want to see and display it on your screen.

In order to achieve this, packets contain information like the sender's IP address, the intended receiver's IP address, something that tells the network how many packets have been created and the number of this particular packet (the sequence number).

Packets also carry the data in the protocols that the Internet uses: Transmission Control Protocol/Internet Protocol (TCP/IP).

1. A packets sole responsibility is to bundle data together.

## Answer:

B - The structure of a packet depends on the type of packet it is and on the protocol. A packet has a header and a payload.

# Question 8:

What is the general order of a data transmission you might make - such as when you send an email or visit a website?

1. Option A
   1. First, your computer creates a packet that the receiving computer can understand
   2. The packet has 3 main sections, and in the header section your computer adds the IP address of the computer that your data is supposed to get to
   3. During this process, your computer adds a sequence number to each packet
   4. Your computer then sends each packet over the Internet separately
   5. When the packet arrives on the receiving host, it travels through the packet protocol stack in reverse order
   6. Finally, the Application Layer on the receiving end receives the message and performs the operation requested
2. Option B
   1. First, your computer creates a packet that the receiving computer can understand
   2. The packet has 3 main sections, and in the header section your computer adds the IP address of the computer that your data is supposed to get to
   3. The packet does not always have to travel through each of the 5 TCP layers, and in many instances your computer will send the packet directly to the destination host.
   4. The destination host will receive the information, process it and always sends a success 200 status code back to your computer.

## Answer: A