

README.md

t1a1-workbook

Q1

HTML

Markup language is an essential tool in the web development world, they are used to define the website structure. HTML is the standard markup text language, and most of the websites in today's World are created using HTML.

Every HTML documents starts with **!DOCTYPE html** to describe document type and it helps the browser to read your document, followed by **html**, **head** and **body** elements.

In element **head** we store information about the website such as title, meta data, CSS link to import design and FavIcon. Some of these information helps with more visibility in the search engines. In element **body** we place everything that is visible to users using elements such as heading, paragraph, anchor links and images.

Semantic tags are a great tool to use through the process to allocate our content, keep it more clean and give our elements more meaning. It's great for screen-readers and accessibility. Some of the most common semantic elements are header, nav, main, section, footer.

Comments are another way to keep everything organised and well explained through the process. It's a powerful tool for web developers, as they can use comments to understand the process.

In HTML language, elements can have attributes, and attributes provides more information about elements. Some of the most common attributes are:

- *lang*: indicates the language, it is placed in the html tag, it also helps for visibility in search engines.
- *href*: indicates the URL that a link goes to
- *src*: path to image
- *width/height*: indicates dimensions of elements
- *style*: gives style such as colour to an element
- *alt*: alternative text used for screen-readers or if in case path/link to image is broken

HTML is a powerful tool that allows you to create lists, forms. Along with a programming language you can also receive and process users data.

References: W3Schools (2022). Introduction to HTML. [online] W3schools.com. Available at: https://www.w3schools.com/html/html_intro.asp

Q2

- Packets Packets are "containers" of data, packets are split into small "packages" of data and they are transferred over from one computer to another through computer network. Once the data is split in small pieces, it chooses the best route to its destination and packets have the ability to travel independently reducing the load on a single route. In case of an issue during the transmission, the route gets changed to ensure the recipient receives the complete data. Once the packets arrived in the destination they are re-organised through a sequence of numbers and they become complete again. Packets has contribute to the development of the internet allowing multiple computers to exchange data simultaneously, allowing computer users to reach each continuously. This technology also allows the reliability of sending and receiving data since routes are changed in case of congestion or failure.
- IP addresses (IPv4 and IPv6) IP stands for Internet protocol, every computer connected to the internet has an IP, which is an unique identifying number. This number is used to identify where the data goes, it's how devices connects one to another.

IPv4 it's the version 4 of internet protocol also know as 32-Bit system, computers switched to IP protocol back in 1983 and this version still in use. IPv4 contains 11 digit numbers separated by dots (decimal structure). It has provided over 4 billions IP, and over the years it ran out of addresses as our usage of devices connected to the internet surpassed the four billion addresses provided by IPv4. The solution for the problem arrived in late 90's, IPv6 (version 6).

- 13 header
- Requires less space for storing address information in the network
- IP addresses are divided by 5 classes A, B, C, D and E
- Checksum
- Manual and DHCP address configuration

IPv6 addresses uses the 128-bit format that means the capacity of unique IP are way more than the previous system and we won't have to worry about running out of IP addresses. Some of the new features in the new version is that the IP addresses contains both numbers and

letters separated by colons(Hexadecimal). With the abundance of available addresses provided by IPv6, every device can have their own IP, meaning that data transmission is smoother and faster since the data goes straight from one device to another.

- Larger header = more data each transfer
- Checks for accuracy in data transmission
- IPSEC standard security for data transfer
- Encryption and Authentication
- More efficient data routing
- Auto-configuration available
- Quality of Service (QoS)

IPv4 has contributed to the development of the internet allowing devices to connect between each other and transfer data efficiently. IPv6 has been improved and brought benefits in the updated version offering more IP addresses to keep up with today's world devices. IPSEC was also implemented to add an extra layer of security and QoS was implemented to prioritise data. Mobile phones, applications and multi-casting were also improved in the new version.

Router is a hardware device responsible for directing data packets through networks and controlling the data traffic between networks. Routers are also used to connect devices in the same network. Different types of routers are wired routers, wireless routers, core routers, etc.

Routing is the process that a data packet takes to reach their final destination. Once the router receives a data packet, it uses a router table to find out the best path to send the data, and the router table uses the content in the header of each packet to decide the final destination. Routing tables can be static, they don't change and they are set by the network administrator, and dynamic tables that get updated automatically to find the fastest/ shortest way to deliver the data, it's a more efficient way.

Routers contributed immensely to the development of the internet allowing multiple devices from the same network to share the internet in both home and business environments. Routers can exchange data and manage data traffic control allowing data to be transferred quickly and efficiently.

Domains and DNS

Domains are names that we use to access data, when you open a browser you don't need to remember the long number/letter address, instead you search for domain name. Domain is part of DNS (Domain Name system), this system converts domains names e.g <https://www.example.com> into IP addresses and then internet browsers are able to load and display content in the browser. The process is broken into few steps which includes DNS resolver gets the query to translate the domain to IP, DNS sends the query to TLD which contains the end of the website domain address, it creates a connection after finding the right domain. The resolver then sends a query to authoritative DNS which respond with the IP address, finally the resolver return the IP address back to the browser, and then the user can access the web page.

Domains are composed by:

- Top-Level Domain(TLD):e.g: *com, org* / country-specific TLD e.g: *com.au*
- Second-Level Domain(SLV): The main name, <http://www.example.com.au> in this case "*example*" is the SLD
- Subdomain: Option part <http://www.blog.example.com.au> in this case *.blog*.

References : Cloudflare (2022). What Is DNS? | How DNS Works | Cloudflare UK. Cloudflare. [online] Available at: <https://www.cloudflare.com/en-gb/learning/dns/what-is-dns/>

How Domain Name Servers Work. [online] HowStuffWorks. Available at: <https://computer.howstuffworks.com/dns.htm>

Q3

- TCP stands for transmission control protocol, it's bridge between the sender and the final destination. TCP is responsible for putting all packets back together, making sure the transmission occurs smoothly free of delays and errors. TCP uses a process called "Three way shake hand" that initialise the connection between the sender and the receiver, making sure there's a stable connection before data exchange. Three Way Handshake consist in tree steps *SYN* (this packet requests initialisation of transmission process, this is sent from source to target server), *SYN-ACK* (target server sends a packet back with agreement to source) and the last packet *ACK* (source sends target server a packet to confirm the connection is established). Another feature is Segment Number system, TCP packets are identified with numbers, in case any problem occurs to a packet in the transmission process, TCP can identify efficiently what packet has failed and retransmit that pack again.
- TCP has contributed to the development client-server making sure data transmission is secure and reliable, free of errors, control the flow of how much data server sends to client, allowing a reliable connection and less probability of corrupted files or congestion.
- HTTP stands for Hypertext Transfer Protocol, it's a transfer protocol that establish the connection between server and client. It's the bridge
- HTTPS is an considered more secure than HTTP. It uses encryption protocol (TLS) to ensure security of data. It's very useful when dealing with private information such as addresses, bank details, anything that contains your private information. Some web browsers such as Google Chrome are able to tell what web page doesn't use HTTPS, and are not secure.

- HTTPS has contributed to client and server to ensure security of data exchanges, it has an extra layer of protection and if it's leaked it won't be able to be understood.
- Web browsers are softwares that enables users access, find webpages and see it's content such as text, videos and images. We can use Web browsers in our computer, laptops, phones. Some of the most used web browsers are: Google Chrome, Safari, Microsoft Edge and plenty more. Web browsers translate HTML documents in web pages, web browser is the bridge between hyper text documents and what the user sees on the screen. Http, TCP and IP all work together.
- Request is the process of requesting a specific path from the client, request uses some import information such as URL.TCP is also established here.
- Rendering this is the process from translating HTML, CSS into webpages we see on the browser.
- Developer tools are tools built into the browsers to help developers fix any errors and optimise the webpages. Some of the developer tools include: elements panel (here you can see html, and css elements applied to webpage).

References: Cloudflare (n.d.). What is HTTPS? | Cloudflare UK. Cloudflare. [online] Available at: <https://www.cloudflare.com/en-gb/learning/ssl/what-is-https/>

GeeksforGeeks. (2024). What is HTTP ? [online] Available at: <https://www.geeksforgeeks.org/what-is-http/>

Q4

- Lines of code are translated one by one as the program runs.
- Interpreters language examples: Python, Ruby, JavaScript.
- Slower processing when compared to compiler languages
- More flexible, program display errors as you go.
- Independent platform
- Compiler languages translate the whole code in one go.
- Compilers languages examples:C, C++, Pascal.
- Processing is faster and more efficiently .
- Harder to debug since the error will only be displayed after processing the whole document.
- Interpreters languages are considered less secure, it requires less memory since no object is generated opposite to Compiler languages. Compiler are harder to debug since you need execute the whole code first, while Interpreter once the error occurs, you can see straight away.

References: Sassi, R.B. (2023). Compiler vs. Interpreter in Programming | Built in. [online] builtin.com. Available at: <https://builtin.com/software-engineering-perspectives/compiler-vs-interpreter>

GeeksforGeeks. (2018). Language Processors: Assembler, Compiler and Interpreter - GeeksforGeeks. [online] Available at: <https://www.geeksforgeeks.org/language-processors-assembler-compiler-and-interpreter/>

Q5

Python

Python is a high programming language and an interpreter language. Some of the benefits of being a high language is the code is easy to understand and code since the language is more similar to a human language. Another benefit is being an interpreter language which translates the code line by line so when an error occurs you can see it straight away and fix the error. Python also has an inbuilt " Standard Library " which offers plenty of functions and modules. Python offers REPL - Read, Eval, Print, Loop which is a function that allows us to test our code instantly. Another nice feature of Python is Object Oriented Language, you can create classes and objects. Python is a great option for people interested in learning a programming language because the syntax is considered easy. Some of the cons of Python include, slower than Compilers languages since the processing is done line by line. When it comes to mobile development Python doesn't rank high, it's not the preferred language for mobile, however is highly desirable for desktop. Consumes more memory than other programming languages.

JavaScript

Java Script is the most popular programming language, most of the websites currently use JavaScript. JavaScript works together with HTML and Css, syntax of JS can be implemented in HTML. It's responsible for creating dynamic content in web pages resulting in a better user experience. Another benefit is the syntax is considered relatively easy and being very popular there are plenty of resources to learn about JS. Different than Python, JS has more compatibility towards mobile and games. With JS you are able to create pop up windows, validate forms, autocomplete sentences, create animations, interactive forms, menus. JS can be used for front-end and back-end purposes, making the language very versatile. Some of the drawbacks is that since JS is a Client-Side language the security is more exposed. Another drawback is that JavaScript is considered harder to debug compared to other client-side languages. Another negative point is that JavaScript can be processed differently depending on the browser.

References:

GeeksforGeeks (2020). Advantages and Disadvantages of JavaScript. [online] GeeksforGeeks. Available at: <https://www.geeksforgeeks.org/advantages-and-disadvantages-of-javascript/>

GeeksForGeeks (2019). Disadvantages of Python - GeeksforGeeks. [online] GeeksforGeeks. Available at: <https://www.geeksforgeeks.org/disadvantages-of-python/>

Neville, M. (2023). The Advantages and Disadvantages of JavaScript - Softjournal. [online] Softjournal Inc. Available at: <https://softjournal.com/insights/the-advantages-and-disadvantages-of-javascript>

Q6

Dear Alex,

Congratulations on the Super Awesome Museum, I'm sure it will do great and our city really needs one. Thank you for your email, I'm beyond excited with the possibility of working together.

First, I would like to make sure I understand some key points for the project. You require a website that invites people to visit the museum, however there's no cost associated with entry and no intention of selling items. You would like have a gallery page containing photos of items and a small brief description or do you prefer to go more in details about those items?

We could have a page displaying all important information such as how to get here, opening/ closing times and possibly add a map to make it easier for customers to find the museum. We could also, have a map displaying all sections inside of the museum that helps customers locate what items/ area they want to see? I believe this is can be very beneficial.

Last we could have a page dedicated to contact, which could include a contact form so customers could message the museum straight from the form, however we could also include email and phone number for those who prefer those methods.

This sounds like a very exciting project and I can definitely work on your website. For my TechStack(tools) I will be using Html, Css and Javascript, if you have any questions related to the TechStake please let me know.

HTML: molds the website structure.
CSS: gives the website styling.
JavaScript: adds animations and cool effects.

Let me know if my understanding aligns with what you want and we could set up a meeting to discuss timeline and prices. If you have any questions regards the process, please let me know. I'm excited to hear back from you. Thank you Alex, best regards.

Jessica Vaz, Fullstack Developer 0000 000 000 Jessvaz@developer.com

Q7

Gibbs Reflective cycle:

- Description:
I have created a website from scratch using HTML and CSS. When creating the submission folder, I have copied the files and placed them in the 2 required folders, however I didn't copy the folder itself, only it's content, so CSS didn't link to HTML.
- Feelings:
I felt very happy to deliver the assignment, however I was very upset for my mistake in the submission folder.
- Evaluation:
The good was being able to deliver the task, learn new skills and be able to create a website and the bad was the mistake I have made in the submission folder.
- Analysis:
My analysis is that every outcome makes me learn something new, even the bad situation. Now I know I need to double check the submission folder before I send it to make sure everything works fine.
- Conclusion:
I could paid more attention, I could asked in the discord chanel if the way I was doing was right instead of guessing it would be ok.
- Action plan:
For the next time I will pay more attention and be more careful.

Q8

- IT conferences and MeetUps: Participate in IT conferences and MeetUps in Melbourne. I expect conferences and MeetUps to be a great way to understand and be involved in the IT field, get to know the latest in Tech, connect with IT professionals and potential employers.

- IT Facebook/ WhatsApp groups: Be part of groups where people are constantly exchanging information, conference and meetups events, job opportunities and resources to learn about specific IT topics. Here I expect to get information on events, study resources and also help with anything I can. Exchange experiences and connect with IT professional all over the world.
- IT StandUps: Here I could talk about my first step into IT with Coder Academy, my first website project and future projects. I expect to connect with others who are in the same area to share experience, share feedbacks, future steps and tips on how to navigate problems.
- LinkedIn: With LinkedIn I expect to show my projects and GitHub profile. I want to connect with other IT professionals and hopefully a potential employer. I want to share all my steps and be active on the platform and take advantage of their library online courses.
- CoderAcademy Placement: Here I could put everything I learned in the course into action, I expect to have a real experience of how is to work as a web developer and gain experience needed to enter the workforce. Here I could excel both hard and soft skills.
- Volunteer jobs: Volunteer jobs is another way to gain experience, I could look for a volunteer job after Coder Academy to gain more experience and confidence to execute the job and help me find my first IT job.

Q9

Language-learning systems are able to provide improved content every time it generates content, because of it's ability of learning. It can provide reports, extract key points of reports making the reading process faster, it can translate documentation. It can proofread and modify reports making it more casual or more formal. When it comes to technical, it's also able to give you instructions to solve problems, steps-by-step to conclude a task. For software development it can be used to help developers to choose the TechStack for the project, to create Boilerplate, to generate codes, generate code documentation, test code and it can provide suggestions to make the code more efficient and clean. It can spot bugs, provide solutions to make the debugging process faster. ChatGPT also adds comments when spilling out the code making it easier for the developer to understand the process, all of the features above allows the developer to save time and prioritise more important tasks. It can be used to explain work-frame documentation, breaking down step by step. It can be used to translate code from one language to another.

References for Question 9 and 10: Khurram, M. (2023). Ethical Considerations of Using ChatGPT in Software Development. [online] Tech Lead Hub. Available at: <https://medium.com/tech-lead-hub/ethical-considerations-of-using-chatgpt-in-software-development-cff577427867>

Q10

Although Language-learning platforms has a lot of advantages and it is a power tool if we use it in a smart and cautions way, it also comes with risks. Some of the legal and ethical issues related to written and technical works are copyright, some of the content provided by language learning platforms could contain copyrighted material making people vulnerable to copyright infringements. Another common issue is misleading information, since the platform uses existing data to generate responses, it can contain mistakes, so it's always important to double check data before using it, using a protocol to check misleading info is also important. Some of the ethical impacts are job replacements, e.g in software projects, companies could prefer using these system in roles that involves writing to cut company cost. Another issue is data privacy and data handling, users needs to informed about data collection, how it is used and what protocols are in place to secure their information.

Q11

- Soft skills:
 - Effective communication: Ability to communicate with team to solve issues, come up with ideas, understand the requirements to deliver common goal.
 - Team work: Be able to work together with the team to deliver the project, always helping each other and supporting.
 - Problem-solving: Be able to solve issues, trace back to the root of the problem and come up with an alternative path to solve the issue.
 - Time management: Be able to prioritise more import tasks. Be aware of the time to deliver project in time, never late.
 - Willingness to learn and help: Be open to help a coworker when needed and be open to ask for help when needed.
- Hard skills
 - Skills/ knowledge gained through education, previous jobs or any experience such as placement.
 - Programming language: You need to understand how they work, what language will be better for each project, be able to code efficiently.
 - Software Dev Tools: Understanding of version control systems.

I believe the balance is to have a good mix of both Soft and Hard Skills. For software development I believe all skills above are equally important, soft skills such as team-work, problem solving and willing to help will make sure the team works together towards one goal. Hard skills will make the work achievable, since the team has the knowledge needed to develop and deliver the application and the ability and knowledge to solve problems as they occur.

References: Indeed Career Guide. (n.d.). Hard Skills vs. Soft Skills. [online] Available at: <https://au.indeed.com/career-advice/resumes-cover-letters/hard-skills-vs-soft-skills>

Q12

Some of the jobs in a development company are:

- Front-End Developer: Developers that work on the user interface side, their work-stack include HTML, CSS and JavaScript. They use work frames tools, draw.io to help them with design, they are responsible for responsiveness designs: designs that works and functions in any screen size.
- Back-End Developer: Developers that work on the server-side, they ensure the server runs properly and smoothly. Their work-stack include programming language such as Python, Ruby or Java.
- Fullstack Developer: Developers that work on both sides, front and back end combination of both skills.
- UI/UX Designers: They design the user interface, their job is to create an interface that it's pleasant to use.
- Graphic designer: They create logos, icons and any other visual content for the application.
- Project Manager: Manages the team, ensure delivery, oversees the project.
- Product Manager: Defines the roadwork to achieve result, and prioritise what needs to be done first.
- QA Engineer: Test the application to make sure everything runs smoothly, report bugs and makes application meets it's standards.
- DevOps Engineer: Automates deployment process and ensure system reliability.
- System Administrator: Administrate servers, networks and systems, they provide technical support for software and operating systems.
- Data Scientist/ Data Analyst: Analyses large amount of data and provides reports on how to improve service/ application.

References: Kanoika, A. (2023). How to Structure Your Software Development Team. [online] SumatoSoft. Available at: <https://sumatosoft.com/blog/software-development-team-structure#:~:text=Medium-sized Business Software Development Teams%3A&text=You might find roles like>