

The internet and the world wide web

Read online at pastopic.com/cambridge/computer-science-0478/notes/internet-and-world-wide-web

Last checked 30 September 2026

Read first: [Network hardware](#), [Encryption](#)

© 2026 Pastopic. Free for personal study. Please share the link, not the file.

What you need to know

By the end of this topic you should be able to:

1. **Tell the internet and the world wide web apart.** The internet is the infrastructure: the networks, cables and devices that carry data around the world. The world wide web is the collection of websites and web pages that you reach by using the internet.
2. **Explain what a URL (uniform resource locator) is:** a text-based address for a web page, made of up to three parts: the **protocol**, the **domain name** and the **web page or file name**.
3. **Describe what HTTP and HTTPS are for and how they work:** HTTP is the set of rules for requesting and sending web pages; HTTPS does the same job but encrypts the data.
4. **Explain what a web browser is for and what else it does.** Its main job is to render HTML and display web pages. It also stores bookmarks and favourites, records your history, lets you use several tabs, stores cookies, gives you navigation tools and provides an address bar.
5. **Describe how a web page is found, fetched and shown** after you type its URL, including the parts played by the web browser, IP addresses, the domain name server (DNS), the web server and HTML.
6. **Explain what cookies are and how websites use them**, and the difference between **session** and **persistent** cookies. Uses include saving personal details, tracking the user's preferences, holding items in an online shopping basket and storing login details.

How HTTPS checks a server with a digital certificate is listed under cyber security (SSL) in the syllabus, but exam questions on HTTPS in this topic ask for it too, so it is explained here. How to lay out a page in HTML (structure and presentation), which some 2022 papers asked about, is **no longer in the syllabus**: you only need to know that web pages are written in HTML and the browser renders it.

Understand it

The internet and the world wide web

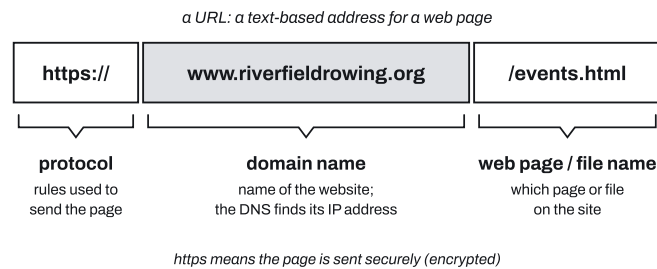
People use the two names as if they mean the same thing, but they don't.

- The **internet** is the hardware and connections: a huge network of networks, joined by cables, fibre, radio links and routers, that lets any connected device send data to any other. It is the **infrastructure**.
- The **world wide web** (WWW) is one thing that runs **on** the internet: the collection of all the **websites and web pages**, linked to each other, that you open in a browser.

An analogy: the internet is the road network; the web pages are the shops you drive to. Roads exist without shops, and a shop is useless without roads. A router or a fibre-optic cable is part of the internet but not part of the web; a page of train times is part of the web. In an answer, never say the web "is" the infrastructure, and never say the internet "is" a collection of web pages.

URLs

Every device on the internet is found by its **IP address**, a number such as 203 . 0 . 113 . 25 (see the network hardware note). Numbers are hard to remember, so each web page also has a **URL**, a **text-based address** that people can read and type.



A URL has up to three parts:

- the **protocol**, such as `https://`: the set of rules the browser and server use to send the page;
- the **domain name**, such as `www.riverfieldrowing.org`: the name of the website (it identifies the web server);
- the **web page or file name**, such as `/events.html`: which page on that site. It can include folder names, for example `/club/events.html`.

If you type just the domain name, the browser fills in the protocol and the server sends its home page.

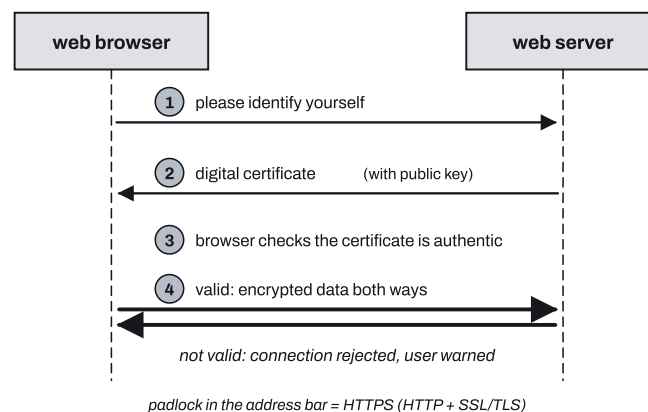
HTTP and HTTPS

A **protocol** is a set of rules for sending data. **HTTP (hypertext transfer protocol)** is the protocol a browser uses to **request** a web page from a web server and that the server uses to **send** the page's data back. With plain HTTP the data travels **unencrypted**, so anyone who intercepts it can read it, passwords and card numbers included.

HTTPS (hypertext transfer protocol secure) is HTTP combined with the **SSL** (secure socket layer) protocol or its newer version **TLS** (transport layer security). It does two extra jobs:

1. It **checks the web server is genuine**, using the server's **digital certificate**.
2. It **encrypts all the data** sent between browser and server, so intercepted data is meaningless.

The connection is set up like this:



- It is the **browser** (the software, not the user) that asks the **web server** (not the web page) to identify itself.
- The encryption is **asymmetric**: data sent to the server is encrypted with the server's **public key** from the certificate, and only the server's **private key** can decrypt it (the encryption note explains the two keys).

A user can tell a site uses HTTPS because the URL begins `https://` and the browser shows a **padlock** next to the address; clicking it shows the digital certificate.

The web browser

A **web browser** is a piece of **software** (an application). Its **main purpose** is to **render HTML**, turning the page's HTML code into the text, images and layout you see, and **display the web page**. HTML (hypertext markup language) is the language web pages are written in, using tags such as `<head>` and `<body>`.

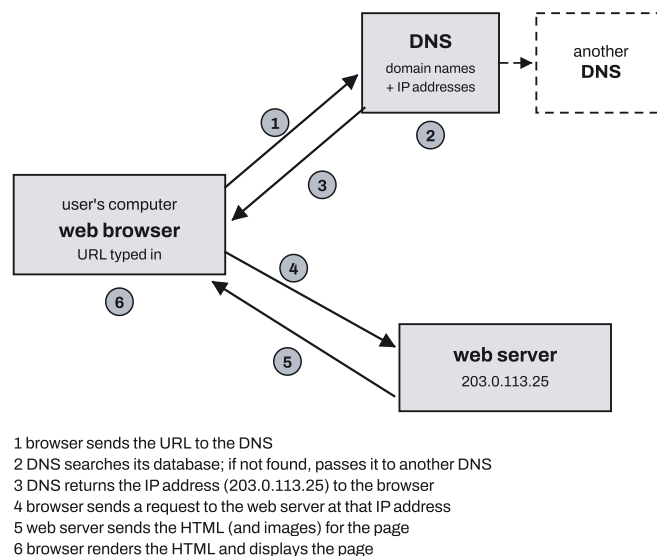
Its other **functions**:

Function	What it does for the user
Address bar	Somewhere to type or see a URL
Bookmarks and favourites	Saves a page's URL so you can go straight back to it
History	Records the pages you have visited, and when
Multiple tabs	Several pages open in one window
Storing cookies	Keeps websites' cookies so sites can remember you
Navigation tools	Back, forward, refresh and home buttons; clickable hyperlinks

Behind the scenes it also sends the URL to the DNS, sends the request to the web server, checks digital certificates for HTTPS, and lets you download files.

How a web page is found, fetched and shown

A URL is for people; the internet delivers data to **IP addresses**. So the first job is to turn the domain name into an IP address, and that is what a **domain name server (DNS)** is for. A DNS is a server that holds a **database of domain names (URLs) and their matching IP addresses**.



1. The user types the URL into the browser's **address bar** (or clicks a link or bookmark).
2. The **browser** sends the URL (its domain name) to a **DNS**.
3. The DNS **searches its database** for the matching IP address.
4. If this DNS doesn't hold that domain name, it **passes the request on to another DNS**, and so on until one finds it.
5. The DNS **sends the IP address back to the browser**.
6. The browser sends a **request** (using HTTP or HTTPS) for the page to the **web server** at that IP address.
7. The **web server** is a computer that stores the website's pages. It finds the page and **sends its HTML** (and any images and other files) back to the browser.
8. The browser **renders the HTML** and **displays** the page.

Because the browser asks the DNS each time, a website can move to a new web server with a new IP address: the DNS record is updated, and users still type the same URL.

Cookies

HTTP on its own has no memory: each request is separate, so a website can't tell that the person asking for page 2 is the one who just logged in on page 1. **Cookies** fix this.

A **cookie** is a **small text file**. It is **created by the website's web server**, sent to the browser, and **stored and managed by the web browser** on the user's device. It holds data about the user, such as an ID, their preferences or login details. Each time the user returns to that website, the browser sends the cookie back, so the site can recognise them. A cookie is **data, not a program**: it can't run or do anything by itself.

There are two types, which differ in **where they are kept** and **how long they last**:

	Session cookie	Persistent cookie
Stored in	Memory (RAM)	Secondary storage (such as the hard drive)
How long it lasts	Temporary: deleted when the browser is closed	Permanent: kept after the browser closes, until it expires or the user deletes it
Typical use	Remembering what you do during one visit, such as a basket before checkout	Remembering you between visits: login details, preferences, payment details

Both are small text files, both are stored and managed by the browser, both are downloaded to the user's device, and both hold data about the user.

What websites use cookies for:

- **saving personal details**, such as name and address, so forms fill in automatically;
- **storing login details**, so the user stays logged in or doesn't retype them;
- **tracking preferences**, such as language, font size or currency, so the page appears the way the user chose;
- **holding items in an online shopping basket**;
- **storing payment details** so they can be filled in at the next purchase;
- **tracking browsing habits**, for example to show **targeted adverts**.

Why users may be concerned: cookies let a site build up a record of what someone looks at, which can be used for targeted adverts or passed on without the user fully realising; and a persistent cookie holding personal or payment details on the device could be read by a hacker or by someone else who uses the same computer.

Key facts and formulas

There are no formulas in this topic. These are the facts to learn; nothing here is given in the exam.

Fact	Status
Internet: the infrastructure (networks, cables, routers, devices); WWW: the collection of websites and web pages accessed using the internet	Learn it
URL: a text-based address for a web page; parts: protocol, domain name, web page/file name	Learn it
HTTP: protocol for requesting and sending web pages; data unencrypted	Learn it
HTTPS: HTTP with SSL/TLS; encrypts data; server checked by its digital certificate; padlock	Learn it
HTTPS steps: browser asks server to identify itself → server sends digital certificate → browser authenticates it; valid: encrypted connection; not valid: rejected, user warned	Learn it
Web browser: software whose main purpose is to render HTML and display web pages	Learn it
Browser functions: address bar, bookmarks/favourites, history, tabs, storing cookies, navigation tools	Learn it
DNS: server holding a database of domain names/URLs and matching IP addresses	Learn it
Retrieval: URL → DNS (next DNS if not found) → IP address to browser → request to web server → HTML back → browser renders	Learn it
Web server: stores web pages; receives requests and sends the requested page	Learn it
HTML: the language used to write web pages	Learn it

Fact	Status
Cookie: small text file created by a web server, stored and managed by the browser	Learn it
Session cookie: RAM, temporary, deleted when the browser closes; persistent cookie: secondary storage, permanent until it expires or is deleted	Learn it

How to do it

In the Paper 1 questions we have tagged for this topic (24 questions, 2021–2025), the types came up this often. 10 of the 24 questions mix more than one type, so the counts add up to more than 24.

Question type	Questions
Cookies: what they are, session and persistent, uses	8
Locating and retrieving a web page (DNS, diagram)	7
HTTP, HTTPS and the secure connection	7
URLs: what they are and their parts	6
Tables and tick grids of internet terms	6
The web browser: purpose and functions	5
The internet and the world wide web	2

1. Cookies: what they are, session and persistent, uses (8 questions)

1. **"What is a cookie?"**: say it is a **small text file**, created by a website's **web server** and **stored by the web browser** on the user's device, holding **data about the user**. Examiners note that many answers say what a cookie stores but never say it is a file.
2. **"Explain the difference between session and persistent cookies"**: give matched pairs, each joined by "whereas":
 - a session cookie is stored in RAM, **whereas** a persistent cookie is stored in secondary storage;
 - a session cookie is deleted when the browser is closed, **whereas** a persistent cookie stays until it expires or the user deletes it.
 "Temporary" and "permanent" alone earn little; add where each is stored and when each is removed.
3. **"State one similarity"**: something true of **both**: small text files; stored and managed by the browser; hold data about the user; downloaded to the user's device.
4. **"Give three uses of cookies"**: pick three different ones from the list in "Understand it". **Read the stem**: if it already mentions the shopping basket, that use can't be given again.

Variations you will meet:

- **Gap-fill paragraphs** with a word list: the answers come from *text*, *web browser*, *web server*, *session*, *persistent*, *temporary*, *permanent*, *expire*, *RAM* or *secondary storage*. The list always contains distractors such as *static*, *dynamic*, *binary* or *image*; cookies are never "static" or "dynamic".
- **Pick the type for a situation**: something kept **after the browser is closed** needs a **persistent** cookie; something needed only **during this visit** suits a **session** cookie.
- **"Describe how cookies store and fill in payment details"**: don't just repeat the question. Say the details are saved in a **persistent** cookie the first time they are entered; the browser keeps it on the device; on the next visit the browser **sends the cookie to the web server**, which **reads** the details and **fills in** the payment form, so the user doesn't retype them.
- **"Why might a user be concerned?"**: browsing habits are tracked (targeted adverts, data passed to others); personal or payment details stored on the device could be stolen by a hacker or seen by another user of the same computer.

2. Locating and retrieving a web page (7 questions)

1. **"Describe the role of the DNS"**: it **stores a database** of domain names (URLs) and matching IP addresses; it **searches** it for the IP address matching the URL it receives; if not found it **sends the URL on to another DNS**; it **returns the IP address to the browser**.

2. **"Describe how a web page is located, retrieved and displayed" or "draw and annotate a diagram"**: use the eight steps in "Understand it". Name the **right actor** at each step: the **browser** sends the URL and the request; the **DNS** finds and returns the IP address; the **web server** sends the HTML; the **browser** renders it.
3. For a diagram, draw four boxes (user's computer with web browser, DNS, another DNS, web server) and numbered arrows. **Each annotation is a mark**, so write one short labelled step per arrow; there are usually 5 or 6 marks.

Variations you will meet:

- **"Identify the system that stores URLs and matching IP addresses"**: the **DNS** (domain name server). **"Identify the software that sends a request to the IP address"**: the **web browser**.
- **"Explain how the URL is converted into an IP address"** (2 marks): the URL is sent to a DNS, which looks up the matching IP address in its database.
- **Half-drawn diagrams** where you complete the arrows and labels: the same steps, one per mark.

3. HTTP, HTTPS and the secure connection (7 questions)

1. **"Describe HTTPS"**: it is a **secure protocol** (a set of rules for transmitting data); it is **HTTP combined with SSL/TLS**; it **encrypts** the data before it is sent.
2. **"Explain how HTTPS makes data secure"**: it uses SSL/TLS; the data is **encrypted**, so if intercepted it can't be understood; the encryption is **asymmetric**: data is encrypted with the web server's **public key** and can only be decrypted with the server's **private key**; the server's identity is checked with its **digital certificate**.
3. **"Explain how a secure connection is created (using SSL)"**: the steps in order:
 1. the **web browser** asks the **web server** to identify itself;
 2. the web server sends its **digital certificate**;
 3. the browser **checks (authenticates)** the certificate;
 4. if it is authentic, an **encrypted** connection is set up and data transmission begins;
 5. if it is not, the connection is **rejected** and the user is warned.

Variations you will meet:

- **Gap-fill** of the steps: the same sentences with blanks for *web browser*, *web server* and *encrypted*. Check who does what: the browser asks and authenticates; the server sends the certificate.
- **"How can a user check a site uses HTTPS, apart from the URL?"**: look for the **padlock**, or view the **digital certificate**.
- **"How does the protocol change the data?"** (1 mark): it **encrypts** it.
- **"Name the protocol"** used to send web pages securely: **HTTPS**; the protocol for sending web pages in general: **HTTP**.

4. URLs: what they are and their parts (6 questions)

1. **"State what is meant by a URL"**: a **text-based address** for a website or web page.
2. **"Identify the parts"**: **protocol, domain name, web page name / file name**. Examiners report that most students remember the last two and forget the protocol.
3. **Labelled URL**: match each part to its label. In `https://www.shop.com/index.html`, `https` is the protocol, `www.shop.com` the domain name and `index.html` the file name.

Variations you will meet:

- **"The protocol is one part; give the other two"**: domain name and web page/file name. Don't repeat what the stem gives.
- **Features of a URL compared with an IP or MAC address**: a URL is text-based and readable by people; it contains a domain name; it has to be converted into an IP address by a DNS; an IP address is numerical and identifies a device on a network.

5. Tables and tick grids of internet terms (6 questions)

1. For each row, decide whether you are giving the **term** or the **description**.
2. **Terms** must be exact: *world wide web*, *IP address*, *DNS*, *web browser*, *web server*, *URL*, *cookie*, *HTML*, *HTTP/HTTPS*, *SSL/TLS*.
3. **Descriptions** must say what the thing **is** and what it **does**: "software that renders HTML to display web pages", not "where you go on the internet"; "a small text file stored by the browser", not "it stores your details".

The descriptions examiners expect:

Term	What to write
World wide web	The collection of all web pages/websites, accessed using the internet
URL	A text-based address for a web page (protocol, domain name, file name)
IP address	A numerical address given to a device when it connects, used to find it (for example the web server)
DNS	Stores URLs and matching IP addresses; gives the browser the IP address for a URL
Web server	Stores web pages; receives requests and sends back the requested page
Web browser	Software that renders HTML and displays web pages
HTML	The language used to write web pages
HTTP	Protocol used to request and send web pages
HTTPS / SSL / TLS	Secure protocol that encrypts data sent to and from a web server
Cookie	Small text file, stored by the browser, holding data about the user

Variations you will meet:

- **Tick grids** (which statements apply to *browser*, *IP address*, *URL*): a URL **contains the domain name**; a browser **is software**, **renders HTML** and **stores cookies**; a URL **and** an IP address are both **types of address**. Tick every column a statement fits.
- Older tables (2021–2022) asked for ISP, which is no longer in the syllabus.
- A row about security (proxy server, hacking) belongs to the cyber security topic.

6. The web browser: purpose and functions (5 questions)

1. **"Describe the main purpose"** (2 marks): it **displays web pages** by **rendering HTML**. Examiners note that many answers list functions here instead of the purpose.
2. **"Give functions"**: any from address bar, bookmarks/favourites, history, multiple tabs, storing cookies, navigation tools (back, forward, refresh), setting a home page. The stem often names one; give **others**.
3. **"Identify the function for this purpose"**: returning quickly to often-used sites → **bookmarks/favourites**; payment or login details filled in automatically → **storing cookies**; finding a page seen last week → **history**.
4. **"Name the software"** that displays the page or sends the request: **web browser**.

7. The internet and the world wide web (2 questions)

1. **"Describe the difference"** (2 marks): the internet is the **infrastructure** (the network of networks); the world wide web is the **collection of web pages/websites** accessed using the internet.
2. **"Define the world wide web"** (1 mark): a collection of web pages (accessed using the internet). Don't add "the infrastructure that...": a contradictory answer loses the mark.

Worked examples

The bold codes show where marks are earned, as in Cambridge mark schemes. The number is how many marks.

- **B1** mark for a correct result on its own, with no method needed.

Example 1

A town library puts its catalogue online at <https://www.eastbridgelibrary.org/catalogue.html>.

- (a) One librarian says the catalogue is "on the internet". Another says it is "on the world wide web". Explain how each description fits. **[2]**
- (b) The library renames the page `search.html`. State which part of the URL changes, and name the two parts that stay the same. **[2]**

(c) The library moves its website to a new web server with a different IP address, but keeps the same domain name. Explain why readers who type the usual URL still reach the catalogue. [3]

(a) The catalogue page is part of the world wide web, the collection of websites and web pages. **B1** It reaches readers through the internet, the infrastructure of connected networks, cables and routers that carries the page's data. **B1**

(b) The web page / file name changes (from `catalogue.html` to `search.html`). **B1** The protocol (`https`) and the domain name (`www.eastbridgelibrary.org`) stay the same. **B1**

(c) Any three:

- the browser sends the URL's domain name to a DNS each time the page is requested **B1**;
- the DNS's database is updated so the domain name now matches the new IP address **B1**;
- the DNS returns the **new** IP address to the browser **B1**;
- so the browser's request goes to the new web server, which sends back the page.

Example 2

Kai wrote these steps to describe what happens after he types a URL into his browser.

1. The browser sends the domain name from the URL to a domain name server (DNS).
2. The DNS searches its database for the matching IP address.
3. If the DNS cannot find the domain name, the browser shows an error message straight away.
4. The DNS sends the web page to the browser.
5. The browser sends a request to the web server at the IP address.
6. The web server sends the HTML back, and the web server displays the page.

Three of the steps are wrong. Identify each incorrect step and rewrite it correctly. [6]

Step 3. B1 If the DNS cannot find the domain name, it passes the request on to another DNS, and so on until the IP address is found. **B1**

Step 4. B1 The DNS sends the matching **IP address** (not the page) back to the browser. **B1**

Step 6. B1 The web server sends the HTML back, and the **web browser** renders the HTML and displays the page. **B1**

Steps 1, 2 and 5 are correct as written.

Example 3

A recipe website uses cookies.

(a) The website's help page says: "Cookies are small programs that run on your computer to make it faster." Explain why this description is wrong. [2]

(b) The website remembers each reader's choice of metric or imperial units, even weeks later. It also remembers which recipes a reader has opened, but only until the browser is closed. For each, identify the type of cookie used and explain where it is kept and when it is removed. [4]

(c) Give one reason a reader may be uneasy about the website's cookies. [1]

(a) A cookie is a small **text file** holding data, not a program, so it cannot run. **B1** Its purpose is to store data about the user (such as preferences or login details) so the website can recognise them, not to speed up the computer. **B1**

(b) Units choice: a **persistent** cookie **B1**, kept in secondary storage after the browser closes, until it expires or the reader deletes it. **B1**

Recipes opened this visit: a **session** cookie **B1**, kept in RAM and deleted when the browser is closed. **B1**

(c) Any one: the site can track which recipes they view, for example to show targeted adverts **B1**; the data could be shared with others; if their device is hacked, data stored in cookies could be stolen.

Example 4

An online concert-ticket seller uses HTTPS for its checkout pages. Before this, the pages used HTTP.

(a) A criminal intercepts the card details a customer sends at checkout. Explain what the criminal would see with HTTP and with HTTPS. [2]

(b) One day, when a customer opens the checkout, her browser shows a security warning instead of the padlock. Describe the steps that took place before the warning appeared, and state what should happen next. [5]

(a) With HTTP the data is sent unencrypted, so the criminal could read the card details. **B1** With HTTPS the data is encrypted using SSL/TLS, so intercepted data is meaningless without the server's private key. **B1**

(b) Any five:

- the customer's **web browser** asked the **web server** to identify itself **B1**;
- the web server sent its **digital certificate** to the browser **B1**;
- the browser checked whether the certificate was authentic / valid **B1**;
- the certificate was not authentic (for example out of date), so the browser could not trust the server **B1**;
- so a secure connection is not made: the connection is rejected and the customer is warned **B1**;
- the customer should not enter card details (had the certificate been valid, an encrypted connection would have been set up).

Common mistakes

- **Mixing up the internet and the web**, or saying both are "the infrastructure and the web pages". Examiners give no credit to answers that contradict themselves. Internet = infrastructure; web = pages.
- **Forgetting the protocol** as a part of a URL. Most students give domain name and file name only.
- **Describing a cookie without saying it is a file**. Many answers say what a cookie stores but not what it is: "a small text file stored by the browser".
- **"Session is temporary, persistent is permanent" and nothing more**. Add where each is stored (RAM or secondary storage) and when each is removed (browser closed, or expiry/deletion).
- **Choosing "static" or "dynamic"** for a type of cookie. Those words describe IP addresses; cookies are session or persistent.
- **Repeating the question**. If the stem says cookies hold a shopping basket or payment details, giving that back as your answer earns nothing; explain **how**, or give a different use.
- **Vague DNS answers**. "It converts a URL" is weaker than "it searches its database of URLs and matching IP addresses and returns the IP address to the browser". Examiners say few students can describe the DNS's role.
- **The wrong actor at a step**. The web server sends the HTML; the browser renders it. The DNS sends an IP address, never the web page. The browser, not the user, asks the server to identify itself, and it is the web server, not the website, that sends the certificate.
- **Listing functions when asked for the purpose** of a browser. The purpose is to render HTML and display web pages.
- **Elaborate drawings**. Diagram marks are for the annotated steps, not for artwork; a few labelled boxes and numbered arrows are enough.

How to get the marks

- **"State", "identify", "give"** (1 mark each): the exact term, such as *DNS*, *web browser*, *persistent*, *protocol*. For a tick question, tick exactly one box unless told otherwise.
- **"Describe"**: one fact per mark, in a sensible order. For a process (retrieving a page, a secure connection), one step per mark, each naming who does it.
- **"Explain"**: give the point **and** why or how. "HTTPS is secure **because** the data is encrypted, **so** it can't be understood if intercepted."
- **"Explain the difference"**: matched pairs joined by "whereas"; each side can earn a mark.
- **"Draw and annotate"**: every arrow gets a short label that states the step. Count the marks and make sure you have at least that many annotated steps.
- **Tables of terms**: spell the term out or give the standard abbreviation (DNS, URL, HTML, HTTPS). A description must say what the thing is or does.
- **Use the context**. If the question is about a clothing shop or a hospital, answers such as "stores items in the shopping basket" should fit that site, and don't repeat what the stem already told you.

Quick check

1. A zoo's page is at `http://www.greenvalleyzoo.net/penguins.html`. (a) Which part changes if the zoo switches the site to a secure protocol, and what does it become? What difference does this make to data sent to the site? (b) Which part changes if the page is renamed `birds.html`?
2. Explain why a router and a fibre-optic cable are part of the internet but not part of the world wide web.
3. A user has already typed the full URL. Explain why the browser still needs to contact a DNS before it can fetch the page.
4. A games website stores (i) a player's chosen sound volume, so it is the same when they come back next week, and (ii) a code that must disappear as soon as the browser is closed. State the type of cookie that suits each and where each is kept.
5. A student wants to get back to a page he read yesterday but can't remember its URL. Name two different web browser functions that could help him, and say how each helps.

Answers

1. (a) The protocol changes from `http` to `https`. The data is then encrypted (using SSL/TLS), so it can't be understood if intercepted, and the server's identity is checked with its digital certificate. (b) The web page / file name changes, from `penguins.html` to `birds.html`; the domain name `www.greenvalleyzoo.net` stays the same.
2. The internet is the infrastructure: the connected networks and the hardware and links (routers, cables) that carry data. The world wide web is the collection of websites and web pages that are accessed using that infrastructure. A router and a cable carry data; they are not web pages.
3. Data on the internet is sent to IP addresses, not text names. The URL's domain name has to be looked up in the DNS's database, which returns the web server's IP address, and only then can the browser send its request to that server.
4. (i) A persistent cookie, stored in secondary storage and kept until it expires or is deleted. (ii) A session cookie, stored in RAM and deleted when the browser is closed.
5. History: it records the pages visited and when, so he can find yesterday's page in the list. Bookmarks/favourites: if he saved the page, he can open it straight from the bookmark. (Navigation tools such as the back button also help if the tab is still open.)

Past questions to try

- **0478/12/M/J/24 Q11**: the internet and the web, the parts of a URL, and browser functions.
- **0478/12/F/M/24 Q4**: draw and annotate how a web page is fetched after a URL is entered.
- **0478/11/M/J/23 Q10**: a browser's purpose and functions, then session and persistent cookies (parts (c) to (e)).
- **0478/13/M/J/22 Q4**: the parts of a URL, how it becomes an IP address, and HTTPS.