

Solutions

Each answer shows the steps, then the **result**.

2.1

- an agreed **set of rules** governing how devices communicate (format, order, meaning, timing, errors)

2.2

- any one: **modularity** (change one layer without the others); standardisation/interoperability; abstraction (hide details)

2.3

- **Application** → **Transport** → **Internet** → **Link**

2.4

- **SMTP**

3.1

- any two: TCP is **connection-oriented**, UDP **connectionless**; TCP **guarantees delivery and order**, UDP does not; TCP **retransmits** lost packets, UDP does not; TCP has higher overhead

3.2

- **HTTP** — request/transfer **web pages**
- **FTP** — transfer **files**
- **SMTP** — **send** email

3.3

- without an agreed protocol the two computers have no shared rules for **format/order/meaning**
- so each side cannot interpret the other's data and communication **fails**

3.4

- **POP3** usually **downloads and removes** mail from the server
- **IMAP keeps** mail on the server and syncs across devices

3.5

- **application layer** — provides network services to the user's programs (e.g. HTTP, SMTP) and formats the data
- **transport layer** — splits the data into packets and provides reliable, in-order delivery (TCP) using port numbers

3.6

- (a) a protocol is an agreed **set of rules** for the exchange

- covering the format of the data, the order of the messages, the speed, and how errors are handled
- without agreement on all of that, two systems from different manufacturers could not interpret each other's bits at all

- (b) retrieving and deleting email — **POP3**

- sending email — **SMTP**
- transferring a large file — **FTP**
- a secure web page — **HTTPS**

- (c) any two, developed: each packet carries a **sequence number**, so the receiver can reassemble them in the right order however they arrive

- each packet carries a **checksum or error check**, and a packet that fails it is **re-requested** from the sender

- (d) an IP address — the **internet (network)** layer

- a port number — the **transport** layer
- HTTP — the **application** layer

- (e) each layer only has to work with the layer directly above and below it through a defined interface

- so a new application protocol can be added at the top without changing the transport, internet or link layers, or any of the hardware