Web Services Exam Flashcards (Ch. 1, 2, 3, 6)

Chapter 1: Intro to IPC

Q: What is the difference between a program and a process?

A: Program: passive code/data. Process: active, executing instance of a program.

Q: Define an independent process.

A: An independent process cannot affect or be affected by other processes and does not share data.

Q: What makes a process 'cooperating'?

A: It can affect and be affected by others and shares data.

Q: Name and explain the two IPC communication models.

A: Shared memory: common region. Message passing: send/receive via OS.

Q: List the two types of IPC message formats.

A: Binary-based and text-based.

Q: What are the two basic operations in message passing?

A: Send (sender transmits), Receive (receiver handles receipt).

Q: Difference between synchronous and asynchronous communication?

A: Synchronous = blocking. Asynchronous = non-blocking or callback.

Q: List the five components of message passing communication.

A: Protocol, Sender, Receiver, Medium, Message.

Q: What is a protocol in IPC?

A: A set of rules for formatting, sending, and receiving data.

Q: What is a socket and what does it identify?

A: An endpoint for communication between two processes (IP + port).

Chapter 2: Foundation of the Web

Q: What is a web resource?

A: Any object/info that can be accessed on the web via URI.

Q: List the three architectural principles of the Web.

A: Identification (URI), Representation (format), Interaction (HTTP).

Q: What is an Internet media type, and how is it structured?

A: A format type: type/subtype (e.g., application/json).

Q: Write the general syntax of a URI and name each part.

Web Services Exam Flashcards (Ch. 1, 2, 3, 6)

A: scheme://host:port/path?query#fragment

Q: What is HTTP and what does it do?
A: A protocol used to send/receive data between client and server.

Q: What are the two types of HTTP messages?
A: Request and Response.

Q: Describe the structure of an HTTP message.
A: Start line, headers, blank line, optional body.

Q: Purpose of the request vs response body?
A: Request body = sent to server, Response body = returned to client.

Q: List four HTTP methods and their use.
A: GET (retrieve), POST (create), PUT (replace), DELETE (remove).

Q: What is the target of an HTTP request called and why?
A: A resource - it's the object being acted on.

Chapter 3: RESTful Web Services

Q: What is REST and why is it used in web services?
A: An architecture style enabling scalable, flexible web services using HTTP.

Q: List the 5 core REST constraints.
A: Client-Server, Stateless, Cacheable, Uniform Interface, Layered System.

Q: What is the optional 6th constraint in REST?
A: Code on Demand.

Q: Purpose of client-server model in REST?
A: Separation of frontend/backend, improves flexibility/scalability.

Q: What does statelessness mean in REST?
A: Each request must carry all needed info, server stores no state.

Q: 4 sub-constraints of the uniform interface?
A: URI, Representation, Self-descriptive messages, Hypermedia.

Q: What is a representation in REST?
A: The data format of a resource (e.g., JSON, XML).

Q: Role of self-descriptive messages?

Web Services Exam Flashcards (Ch. 1, 2, 3, 6)

A: Messages include headers to describe content/context.

Q: What is HATEOAS?
A: Hypermedia links in response that guide client actions.

Q: How does caching improve REST performance?
A: Stores responses to reduce repeated server requests.

Q: What is a layered system in REST?
A: Allows proxies, gateways - client doesn't know internal layers.

Q: What is code-on-demand?
A: Server sends executable code (optional constraint).

Q: Why does REST use nouns not verbs for resources?
A: Resources = nouns (URIs); actions = verbs (HTTP methods).

Q: 4 common HTTP methods in REST?
A: GET, POST, PUT, DELETE - used to interact with resources.

Q: What makes a web service RESTful?
A: It follows all 5 REST constraints (+ optional code on demand).

Q: What is ROA in REST?
A: Resource-Oriented Architecture using REST principles.

Q: 2 pros and 1 con of RESTful services?
A: Pros: simple, format-flexible. Con: lacks built-in state/security.

Q: Why is REST not a protocol?
A: It's a flexible architecture style, not strict rules like SOAP.

Q: Difference: collection, sub-collection, singleton?
A: Collection = group, Sub = nested under one, Singleton = unique.

Q: How does a REST client transition state?
A: Via representations with hyperlinks (HATEOAS).

Chapter 6: Content Negotiation

Q: Difference between resource and representation?
A: Resource = abstract data; Representation = specific format (e.g., JSON).

Q: Define content negotiation.

Web Services Exam Flashcards (Ch. 1, 2, 3, 6)

A: Choosing the best representation when multiple formats exist.

Q: What is proactive content negotiation?
A: Client sends headers; server selects best match.

Q: What is reactive content negotiation?
A: Server sends options; client picks preferred one.

Q: What does a q-factor do?
A: Sets priority of formats in Accept header.

Q: Default value of a q-factor if missing?
A: 1.0

Q: What header is used to request specific formats?
A: Accept

Q: Status code if no acceptable format found?
A: 406 Not Acceptable

Q: What does the Content-Type header tell the client?
A: The format of the returned resource.

Q: Why list multiple types in Accept header?
A: To offer fallbacks in case preferred format isn't available.