Câu hỏi bài tập chương 2

Câu 1: Khái Niệm XML

XML là ngôn ngữ đánh dấu mở rộng.

XML được tổ hợp web toàn cầu (World Wide Web Consortium) W3C đề nghị tạo ra với mục đích phát triển và mở rộng các ngôn ngữ đánh dấu khác ngoài định dạng .txt, ASCII,…

Cách lưu trữ file XML này khá đơn giản, nhưng nó có thể mô tả được nhiều dữ liệu khác nhau. XML có nhiệm vụ truyền dữ liệu và mô tả những loại dữ liệu khác nhau. XML (Extensible Markup Language) là ngôn ngữ đánh dấu mở rộng.

XML được sử dụng với mục đích lưu trữ, cấu trúc khi trao đổi dữ liệu giữa các ứng dụng và để lưu trữ dữ liệu. Vì vậy việc xây dựng file XML tương đối hữu ích trong việc chia sẻ dữ liệu giữa các hệ thống, chương trình.

Câu 2: Khái Niệm DOM

DOM là tên viết tắt của (Document Object Model – có thể hiểu là Mô hình Đối tượng Tài liệu), là một giao diện lập trình độc lập nền tảng và độc lập ngôn ngữ. Nó cho phép các chương trình, các mã lập trình truy xuất động và cập nhật nội dung, cấu trúc cũng như định dạng của tài liệu HTML hay XML được định nghĩa bởi W3C. [W3C.ORG]

DOM là một chuẩn dùng để truy xuất và thao tác trên các tài liệu có cấu trúc dạng HTML hay XML bằng các ngôn ngữ lập trình thông dịch (scripting language) như Javascript, PHP, Python…

DOM giúp thao tác dữ liệu theo mô hình hướng đối tượng. Các phần tử bên trong 1 tài liệu có cấu trúc được định nghĩa thành các đối tượng, phương thức và thuộc tính để có thể truy xuất dễ dàng mà vẫn đảm bảo tính cấu trúc: mỗi phần tử là một đối tượng, sở hữu các thuộc tính và các phương thức để làm việc với các thuộc tính đó như thêm, xóa, sửa, cập nhật, …

Câu 3: Cú pháp tài Liệu XML

<nodename>nội dung</nodename>

Trong đó:

<nodename>là thẻ mở, tên của thẻ này do người dùng tự định nghĩa. •

</nodename>là thẻ đóng, tên của thẻ này phải trùng với tên của thẻ mở.

content là nội dung của thẻ này.

Câu 4: Khái Niệm JSON

JSON là chữ viết tắt của JavaScript Object Notation, đây là một dạng dữ liệu tuân theo một quy luật nhất định mà hầu hết các ngôn ngữ lập trình hiện nay đều có thể đọc được, các lập trình viên có thể sử dụng lưu nó vào một file, một record trong CSDL rất dễ dàng.

JSON có nguồn gốc từ JavaScript nhưng là định dạng dữ liệu độc lập với ngôn ngữ. Việc tạo và phân tích cú pháp JSON được hỗ trợ bởi nhiều ngôn ngữ lập trình hiện đại. JSON có định dạng đơn giản, dễ dàng sử dụng và truy vấn hơn XML rất nhiều nên tính ứng dụng của nó hiện nay rất là phổ biến, trong các ứng dụng sắp tới việc sử dụng JSON có thể sẽ trở thành đa số.

Định dạng JSON sử dụng các cặp key – value để lưu trữ dữ liệu. Nó hỗ trợ các cấu trúc dữ liệu như đối tượng và mảng

Câu 5: Hạn chế của việc sử dụng JSON trong python

Triển khai JSON trong Python có một số hạn chế như sau:

Hiệu suất: Việc xử lý JSON có thể chậm hơn so với các định dạng nhị phân như Protocol Buffers hay MessagePack, đặc biệt khi làm việc với dữ liệu lớn.

Loại dữ liệu: JSON chỉ hỗ trợ một số kiểu dữ liệu cơ bản như chuỗi, số, mảng, đối tượng, boolean và null. Nó không hỗ trợ trực tiếp các kiểu dữ liệu phức tạp như tuple, set, hoặc các kiểu dữ liệu tùy chỉnh.

Độ chính xác của số: JSON không phân biệt giữa số nguyên và số thực, có thể dẫn đến mất độ chính xác khi chuyển đổi giữa các kiểu số.

Chuyển đổi không hoàn hảo: Việc chuyển đổi giữa các kiểu dữ liệu Python và JSON không luôn diễn ra một cách mượt mà. Ví dụ, một đối tượng datetime không thể được lưu trữ trực tiếp trong JSON mà cần phải chuyển đổi thành chuỗi.

Không hỗ trợ nhận dạng: JSON không có cơ chế để quản lý phiên bản hay nhận diện kiểu dữ liệu, điều này có thể gây khó khăn khi làm việc với dữ liệu phức tạp.

Kích thước tệp: Dữ liệu JSON thường lớn hơn so với các định dạng nhị phân khác do việc sử dụng chuỗi ký tự cho tên thuộc tính và cấu trúc.

Bảo mật: Khi nhận dữ liệu JSON từ nguồn không đáng tin cậy, có nguy cơ bị tấn công XSS hoặc các lỗ hổng khác nếu không xử lý cẩn thận.