

# Prova d'Esame—Tecnologie Web L-A

A.A. 2008/2009

**Tempo Disponibile: 3 ore**

*Ai candidati è consentita la consultazione di manuali e materiale didattico*

## Esercizio 1. Modellazione

Si progetti il DTD per documenti XML che rappresentano la lista dei voli di una compagnia aerea. La lista è costituita da 0 o più aeroporti. Ogni aeroporto ha un nome, un codice identificativo ed una lista composta da uno o più voli. Ogni volo è caratterizzato dalla data, dall'orario di partenza, dalla durata del volo e dalla destinazione. Ogni destinazione è caratterizzata dal codice dell'aeroporto di destinazione, dalla città, e dal paese.

## Esercizio 2. Programmazione Client-Side

Si richiede di implementare la sola parte client-side di un applicativo Web in tecnologia AJAX per il reperimento di informazioni sui voli disponibili fra due aeroporti specificati. La pagina è costituita da una form contenente due campi di testo editabili dall'utente. I campi di testo devono consentire all'utente di inserire il codice identificativo degli aeroporti di partenza e di destinazione di suo interesse. I codici degli aeroporti sono costituiti da stringhe di tre caratteri alfabetici. La form è dotata di un bottone che consente di inviare la request al server. Le informazioni devono essere inviate al server in modo asincrono tramite una chiamata GET e passando i codici identificativi degli aeroporti di partenza e di destinazione. A seguito della pressione del bottone la pagina deve:

- verificare che i codici identificativi degli aeroporti di partenza e destinazione siano costituiti da tre caratteri alfabetici. In caso contrario deve essere segnalato il problema inserendo sotto la form un messaggio che informa l'utente dell'errore;
- nel caso in cui entrambe i codici identificativi degli aeroporti siano stati inseriti correttamente viene inviata la richiesta al server. Si richiede che i risultati contenuti nella risposta del server vengano presentati in una tabella che deve essere inserita sotto la form. In particolare, la prima riga riporta il nome degli aeroporti di partenza e di destinazione. Le righe successive riportano i dati sui voli disponibili: per ogni volo disponibile vengono riportati il nome della compagnia aerea ed prezzo del volo (vedi Tabella1);

<i>Nome dell'Aeroporto di Partenza</i>	<i>Nome dell'Aeroporto di Destinazione</i>
<i>Nome della Compagnia Aerea</i>	<i>Prezzo del Volo</i>
<i>Nome della Compagnia Aerea</i>	<i>Prezzo del Volo</i>
.....	
<i>Nome della Compagnia Aerea</i>	<i>Prezzo del Volo</i>

**Tabella 1.** *tabella presentata all'utente nel caso in cui siano presenti voli fra gli aeroporti selezionati*

- Per semplicità si assume che siano disponibili voli fra gli aeroporti specificati.

Si assume che il server restituisca i dati sui voli in formato XML come segue:

```
<?xml version='1.0' encoding='UTF-16'?>
<informazioni>
  <aeroporti>
    <partenza> nome dell'aeroporto di partenza </partenza>
    <destinazione> nome dell'aeroporto di destinazione </destinazione>
  </aeroporti>
</listavoli>
<volo>
  <compagnia> nome della compagnia </compagnia >
  <prezzo> prezzo del volo </prezzo>
</volo>
<volo>
  <compagnia> nome della compagnia </compagnia >
  <prezzo> prezzo del volo </prezzo>
</volo>
          .....
<volo>
  <compagnia> nome della compagnia </compagnia >
  <prezzo> prezzo del volo </prezzo>
</volo>
</listavoli>
</informazioni>
```

dove *nome dell'aeroporto di partenza* e *nome dell'aeroporto di destinazione* rappresentano rispettivamente il nome dell'aeroporto di partenza e di destinazione; *nome della compagnia* rappresenta il nome della compagnia aerea; *prezzo del volo* il prezzo del biglietto aereo.

### Esercizio 3. Programmazione Server-Side

Si richiede di implementare una sezione dell'applicazione Web per la gestione dei voli di una compagnia aerea. In particolare, si richiede che il candidato implementi due distinte Servlet che consentano di inserire nuovi voli (InserisciNuovoVolo) e di leggere i dati sui voli che questa effettua (LeggiVoli). Le Servlet hanno accesso ad un comune data-base costituito da una sola tabella strutturata come segue :

<i>CodiceVolo</i>	<i>CodiceAeroportoPartenza</i>	<i>CodiceAeroportoDestinazione</i>
<i>CodiceVolo</i>	<i>CodiceAeroportoPartenza</i>	<i>CodiceAeroportoDestinazione</i>
<i>CodiceVolo</i>	<i>CodiceAeroportoPartenza</i>	<i>CodiceAeroportoDestinazione</i>
.....		
<i>CodiceVolo</i>	<i>CodiceAeroportoPartenza</i>	<i>CodiceAeroportoDestinazione</i>

dove *CodiceVolo* è un intero che rappresenta il codice del volo, mentre *CodiceAeroportoPartenza* e *CodiceAeroportoDestinazione* sono stringhe che rappresentano rispettivamente il codice dell'aeroporto di partenza e di destinazione.

La Servlet *InserisciNuovoVolo* consente di inserire un nuovo volo nel data-base. La servlet richiede il passaggio di *CodiceVolo*, *CodiceAeroportoPartenza* e *CodiceAeroportoDestinazione* via HTTP GET. La Servlet, una volta ricevuti i parametri, verifica che non esista nel data-base alcun volo con il *CodiceVolo* specificato. La servlet verifica inoltre che il valore dei parametri *CodiceAeroportoPartenza* e *CodiceAeroportoDestinazione* siano differenti. Nel caso in cui i controlli vadano a buon fine la Servlet inserisce il nuovo volo nel data-base e restituisce all'utente una pagina di conferma dell'avvenuto inserimento. In caso contrario viene presentata all'utente una pagina in cui si segnala che si sono presentati errori nell'inserimento.

La Servlet *LeggiVoli* consente di presentare all'utente una pagina che visualizza tutti i voli in partenza da un aeroporto specificato. A questo fine, la Servlet richiede il passaggio di una richiesta HTTP POST che specifica il codice di un aeroporto. Una volta ricevuto il codice di un aeroporto la Servlet controlla che esista almeno un volo in partenza dell'aeroporto specificato. In questo caso, viene presentata all'utente una pagina in cui è inclusa la lista dei codici di tutti i voli in partenza dall'aeroporto specificato (vedi esempio sotto).

- *CodiceVolo*
- *CodiceVolo*
- *CodiceVolo*
- ...
- *CodiceVolo*

Nel caso in cui non esista nel data-base alcun volo in partenza dall'aeroporto specificato viene mostrata all'utente una pagina che segnala l'errore.

Si richiede che il candidato implementi l'accesso al database utilizzando il pattern DAO. Si richiede inoltre che il data-base sia implementato utilizzando un DBMS relazionale (es. hsqldb) oppure, in alternativa, utilizzando semplici file di testo.