

Esame di Tecnologie Web

CdL in Informatica per il Management - a.a. 2020/2021

Data Appello: 03/06/2021 - turno I

Tempo a disposizione: 2h

INFORMAZIONI PRELIMINARI

L'esame consiste nella scrittura di codice HTML, CSS e Javascript ed esercizi su API ed altri argomenti del corso. Scrivere il codice usando un editor di testo (jEdit o Gedit). Per testare il risultato usare il browser Firefox o Chromium.

La scelta dei nomi dei file, se non espressamente indicati, non è rilevante. I file possono essere salvati in directory diverse. E' possibile comprimere file/directory e decomprimere.

Consegnare tutti i file su IOL.

NON è consentito l'uso di testi e appunti cartacei.


E' consentito il solo accesso a Virtuale. NON è consentito l'accesso ad altre risorse on-line.

NOTE TECNICHE

Risorse utili allo svolgimento della prova:

- Librerie jQuery, Bootstrap: <http://diiorio.nws.cs.unibo.it/twe/lib/>
- Sorgenti HTML degli esercizi, immagini: <http://diiorio.nws.cs.unibo.it/twe/resources/>

Combinazioni di tasti e applicazioni utili:

- Tasto "Windows" : per aprire menù con le applicazioni disponibili
 - *Nautilus*: esplora file
 - *Firefox/Chromium*: browser
 - *jEdit, GEdit*: editor di testo
- ALT+TAB: per passare da un'applicazione all'altra
- CTRL+C, CTRL+V: copia&incolla
- CTRL+R: ricarica pagina nel browser
- CTRL+S: salva file
- Per cambiare le impostazioni del testo in jEdit: menù "Preferences"->Global Options->jEdit->Textarea

NOTA LOGISTICA

Agli studenti affetti da disturbi quali dislessia, è richiesto di completare tutti gli esercizi in un tempo maggiorato del 30%.

Nota per lo studente: Per sicurezza, indicare "**studente con DSA**" al momento della consegna.

TESTO DEL COMPITO

Esercizio 1

Progettare un API REST (parziale) per la gestione di una scuola elementare e descriverla in Swagger/OpenAPI. La scuola è formata da classi, ognuna caratterizzata da una sezione ("A", "B", "C", ...), da un anno (da 1 a 5) e da un giudizio complessivo in formato testuale; ogni classe include diversi alunni, ognuno caratterizzato da un ID e un nome. Tutti gli attributi sono obbligatori.

Scrivere un file in formato JSON o YAML.

L'API permette di:

- ottenere l'elenco di tutte le classi di un determinato anno
- aggiungere un alunno in una classe
- aggiornare il giudizio complessivo di una classe

Specificare: URL di accesso, metodi HTTP, parametri e risposte con esempi.

L'API restituisce un errore (e codice 400) se i parametri in input non sono corretti.

Note:

- Non è richiesto includere le sezioni *host*, *schemes*, *servers*, *tags*
- Non è richiesto gestire autenticazione

Esercizio 2

Dato il codice HTML mostrato di seguito:

1. Scrivere il codice CSS, in un file esterno, per ottenere la visualizzazione mostrata in Figura 1 se il documento è visualizzato in un browser.
Oltre alle caratteristiche tipografiche già evidenti in figura, si tenga presente che:
 - L'area con le informazioni relative ai resort è centrata, occupa 50% del viewport ed è divisa in due aree, una per ogni resort
 - al passaggio del mouse sull'area relativa ad un resort lo sfondo cambia colore e diventa #FFDDAA;
 - misure esatte di margini, bordi e padding non sono rilevanti
 - colori esatti non rilevanti
2. Aggiungere anche le regole per visualizzare solo le aree relative ai resort, senza gli elementi del form, se il documento viene stampato.
 - Il documento in stampa usa un font "Times New Roman"
 - I bordi delle aree non sono arrotondati
 - le altre caratteristiche tipografiche della stampa non sono rilevanti

Vincoli:

- non è possibile modificare il codice della pagina HTML;
- non è possibile usare layout *grid* e *flexbox*
- non è possibile usare Javascript (eventuali comportamenti dinamici vanno nell'esercizio successivo).

Sorgente HTML:

- Disponibile su: <http://diiorio.nws.cs.unibo.it/twe/resources/0306/>

Elemento BODY:

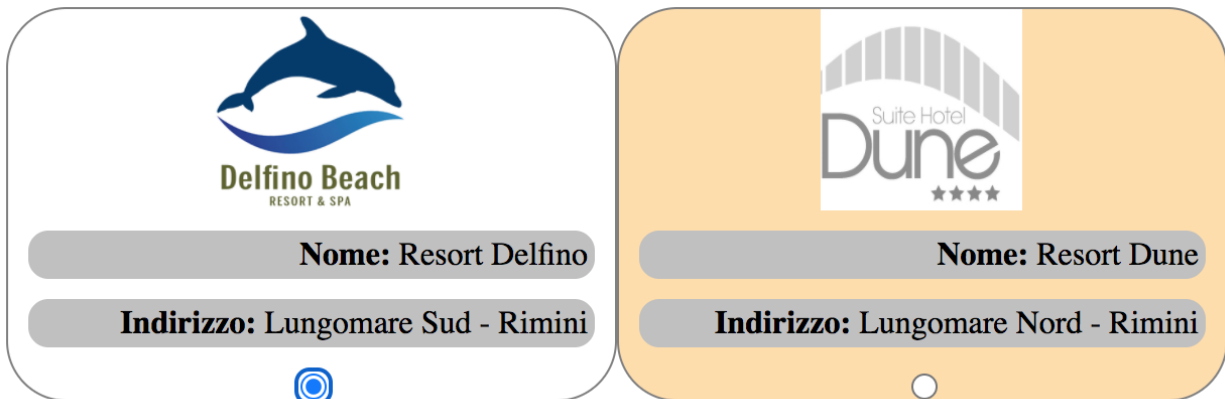
```
<body>
    <form action="richiesta.php">
    <div>
        <h3>Aggiorna descrizione:</h3>
        <textarea rows="4" cols="40" name="descrizione">Un
resort meraviglioso con vista delfinario...</textarea>
        <p>
            <input type="submit" value="Invia" name="submit"
/>
            <input type="reset" value="Annulla"
name="annulla" />
        </p>
    </div>
    <div role="resorts">
        <div>
            
```

```

        <p>Resort Dune</p>
        <p>Lungomare Nord - Rimini</p>
        <input type="radio" name="hotel" value="dune" id="dune"
checked="checked"/>
    </div>
    <div>
        
        <p>Resort Delfino</p>
        <p>Lungomare Sud - Rimini</p>
        <input type="radio" name="hotel" value="delfino"
id="dune"/>
    </div>
</div>
</form>
</body>

```

Figura 1 - Resa della pagina in un browser:



Aggiorna descrizione:

Un resort meraviglioso con vista
delfinario...

Invia

Annulla

Esercizio 3

Data la pagina HTML dell'esercizio precedente, in cui l'area con le informazioni relative ai resort è inizialmente vuota, si realizzino alcuni comportamenti dinamici utilizzando Javascript.

Vincoli:

- è ammesso l'uso di jQuery

In particolare si realizzi:

1. Al caricamento, la pagina accede asincronamente in GET al servizio web <http://diiorio.nws.cs.unibo.it/twe/api/resort/resort.php>, ottenendo un JSON con i dati dei resort, e li visualizza in modo appropriato.

```
[
  {
    "id" : "delfino",
    "nome": "Resort Delfino",
    "indirizzo" : "Lungomare Sud - Rimini",
    "descrizione" : "Un resort meraviglioso con vista delfinario..."
  },
  {
    "id" : "dune",
    "nome": "Resort Dune",
    "indirizzo" : "Lungomare Nord - Rimini",
    "descrizione" : "Le dune...che meraviglia..."
  }
]
```

Note sulla visualizzazione:

- La visualizzazione include nome, indirizzo e logo di ogni resort.
 - Gli URL delle immagini sono calcolati dall'ID di ogni resort e possono essere caricate da: <http://diiorio.nws.cs.unibo.it/twe/api/resort/images/>
 - La descrizione del resort non è visibile (ma usata al punto 2.)
 - Le aree relative ad ogni resort sono affiancate.
2. Alla selezione di un resort viene visualizzata nella textarea la descrizione del resort selezionato.

Suggerimento: si può sfruttare l'evento "change" sugli elementi radio-button

Esercizio 4

Dato il frammento seguente HTML:

```
<section id="main">
  <section id="main-content">
    A month is a unit of time, used with calendars, that is approximately as
    long as a natural orbital period of the Moon; the words month and Moon are
    cognates.
  </section>
  <nav>
    <div id="jan" class="nav-item">Jan</div>
    <div id="feb" class="nav-item">Feb</div>
    <div id="mar" class="nav-item">Mar</div>
  </nav>
</section>
```

Scrivere il codice CSS per ottenere in un browser la visualizzazione mostrata sotto.

Vincoli:

- **Usare il layout *flexbox***, non è ammesso usare altri meccanismi di layout
- La larghezza del contenitore *#main* è arbitraria
- La sezione *#main-content* occupa metà dello spazio a disposizione in larghezza, la restante parte è occupata dal menù di navigazione
- Misure esatte di margini, bordi e padding non sono rilevanti
- Colori esatti non rilevanti

Resa nel browser:

A month is a unit of time, used with calendars, that is approximately as long as a natural orbital period of the Moon; the words month and Moon are cognates.

Jan

Feb

Mar