

# Insegnamento di Tecnologie Web

## CdS In Informatica

(A.A. 2020-21)

Esame scritto del del 20/07/2021

*Prova online, III turno*

**Nome:**

**Cognome:**

**Matricola:**

**Corso di Studi**

*Ad esempio: Informatica, Informatica per il Management, etc.*

**Anno di frequenza:**

*Come specificato nel piano di studi: o "2020-21" oppure "precedente".*

**Attenzione:**

- *Rispondete solo negli spazi delimitati dai blocchi ```` qui la risposta ```` , senza modificarli o eliminarli.*
- *Consegnate solo questo file. Copiate ed incollate dentro agli appositi spazi la vostra risposta per intero.*
- *Potete decidere se inserire il CSS inline nel file HTML o metterlo in un file esterno. Nel secondo caso inserite l'elemento nella posizione corretta e mettete il CSS in un blocco separato.*
- *You can use either English or Italian for your answers.*
- *Per favore, per favore, per favore: nessun errore di ortografia. Questa è un università e non la scuola elementare.*

## Domanda #1 - Domande di base

### a) HTTP

Spiegare brevemente a cosa corrisponde il codice di stato *301 Moved Permanently*.

### b) HTML

Fare un esempio d'uso appropriato e non banale degli elementi `dl`, `dt`, `dd`.

### c) CSS

Scrivere una regola CSS a piacere con un selettore che contenga il carattere " " (spazio) e almeno tre statement appropriati e non banali, e commentarla.

### d) Codifica caratteri

Indicare quali tra i seguenti non sono uno schema di codifica caratteri:

- ASCII
- ASCII esteso
- UTF-8
- UTF-16
- CHAR3
- ISO Latin1

## Domanda #2 - HTML + CSS

### Parte I

Scrivere il codice HTML e CSS (bootstrap è ammesso solo se importato correttamente nella pagina) della seguente pagina web. Le immagini sono fornite in uno zip scaricabile da EOL. Il codice deve funzionare su Firefox. Può essere usato come base il documento `base.html`. Non è importante essere totalmente precisi con colori e misure, ma essere ragionevolmente attenti alle differenze tra elemento ed elemento. Se il file non si apre correttamente nel browser, il punteggio è 0.

Si noti che il rettangolo in primo piano intitolato “Purchase tickets” va considerato come una finestra modale che nasconde temporaneamente il resto della pagina, e che il contenuto sottostante va comunque marcato in maniera corretta (ci si può inventare il contenuto non visibile).

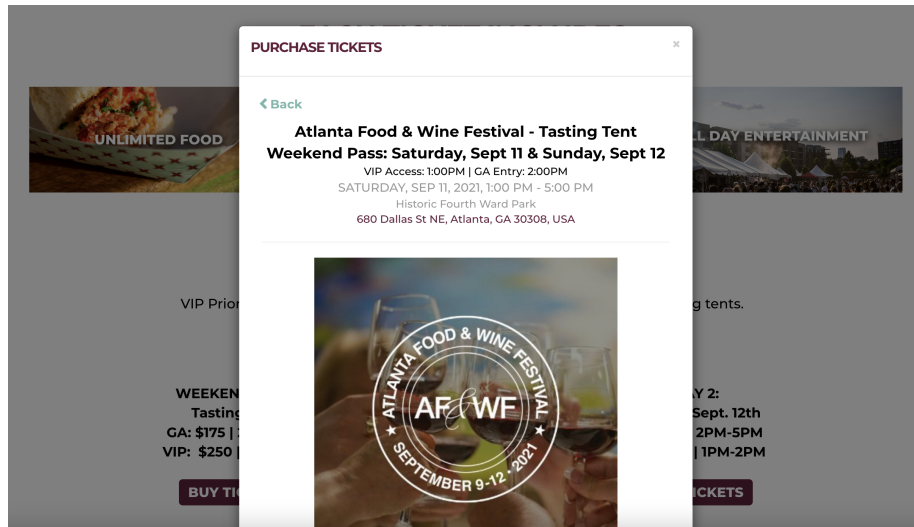


Figure 1: domanda HTML e CSS

### Parte II

Scrivere il codice Javascript aggiuntivo che, quando l'utente fa click sul pulsante di chiusura in alto a destra, crea un bordo rosso intorno alla finestra modale e chiede all'utente “Sei sicuro di voler chiudere senza prenotare?”.

**Codice HTML** (ed eventualmente CSS interno)

**Codice CSS** (solo se esterno)



## Domanda #3 - Javascript

Si sta progettando un jukebox musicale in rete che permette di prenotare l'esecuzione di un brano tra quelli disponibili. I brani scelti vengono accodati in una delle playlist esistenti, e al momento giusto riprodotti in streaming a tutti gli abbonati alla playlist. La playlist accetta al massimo un brano da ogni utente e può durare al massimo 3 ore, e non accetta nuovi accodamenti se è già piena (ma ogni brano suonato riapre del necessario la coda, ovviamente). Ogni playlist e ogni brano sono identificati da un id. Si ipotizzi poi a vostra discrezione un'API REST con tre servizi:

- `/brani`, restituisce l'elenco dei brani disponibili e compatibili con un certo pattern testuale di ricerca (ad esempio "Queen" restituirà tutti i brani dei Queen e tutti i brani che contengono la parola "Queen" nel titolo). Ogni brano ha un titolo, un artista e una durata.
- `/programma`, restituisce l'elenco dei brani in coda per una data playlist. Ogni playlist ha un titolo e una breve descrizione e l'elenco dei brani attualmente in coda.
- `/aggiungi`, che permette di accodare una specifica canzone ad una specifica playlist.

Potete usare i servizi <http://site202100.tw.cs.unibo.it/info> oppure <http://www.fabiovitali.it/TW/test/2021/doResponse.php> per effettuare prove e debugging. Non dovete descrivere l'API, solo ipotizzarla con chiarezza e usarla in maniera corretta.

Basandosi, dove si ritiene, su uno o più framework Javascript a piacere tra quelli illustrati a lezione, si descrivano:

**Parte I** Il codice HTML che contiene la struttura di una pagina e una testata del sito con un menu a piacere (questa struttura non cambia mai). Oltre agli elementi fissi, la pagina contiene:

- un modo per visualizzare l'elenco di tutte le canzoni disponibili
- un modo per filtrarle secondo un criterio a scelta dell'utente (titolo, artista).
- un modo per scegliere una playlist tra quelle esistenti sulla base del nome.
- un modo per visualizzare nome, descrizione e coda della playlist selezionata
- un modo per accodare un brano alla playlist selezionata

**Parte II** Uno o più script Javascript che, appena prima di visualizzare la pagina dell'applicazione, richiama il servizio `/brani` e il servizio `/programma` (con i parametri correttamente inizializzati sulla base delle ultime richieste effettuate, oppure una richiesta `top10` di default) e popolano correttamente gli elementi della pagina. Fornire tutti i meccanismi di chiamata (ricordiamo: ci

sono due liste diverse, playlist e brani), modificare la pagina quando i dati vengono ricevuti e gestire eventuali errori.

**Parte III** Uno o più script Javascript che vengano attivati cliccando sugli elementi necessari per fare una nuova ricerca di brani o di playlist, e uno o più script per accodare un brano alla playlist attualmente visualizzata invocando il servizio `/aggiungi` (con gli eventuali parametri correttamente inizializzati). Si fornisca tutto il meccanismo di invocazione, di gestione della risposta sia in caso positivo sia in caso di eventuali errori, in tutti i tipi possibili di errore. Nel caso di errore, si cerchi una soluzione più soddisfacente che mostrare un alert di errore. Si ricordi che ogni playlist può al massimo durare tre ore, ma via via si svuota dai brani già suonati, e che ogni utente può al massimo suggerire un brano in ogni playlist.

#### **Domanda #4 - Semantic Web**

Scrivi in Turtle il grafo RDF della seguente frase, poi specifica quante e quali entità descrive: «Thomas Atkinson (1729 – 1798) è stato un architetto inglese noto soprattutto per aver rimodellato Bishopthorpe Palace nello stile Neogotico».

## Domanda #5 - Domanda di accessibilità

La tabella di seguito riportata contiene le informazioni sulla presenza di alcuni specialisti all'interno di un poliambulatorio medico. Scrivere il codice HTML necessario per rappresentare queste informazioni sotto forma di tabella rispettando tutti i criteri di accessibilità.

Giorno	Orario	Medico	Specializzazione
Lunedì	09:00-13:00	dott. Tiradenti	dentista
mercoledì	15:00-19:00	dott.ssa Curabimbi	pediatra
venerdì	09:00-13:00	dott.ssa Leggitesta	psicologa

Il menu deve rispettare tutti i criteri di accessibilità; non è possibile utilizzare il codice JavaScript di Bootstrap.



### **Domanda #6 - Domanda di teoria**

Differenze e similarità tra il modello a prototipi di Javascript e il modello a classi di altri linguaggi di programmazione come C++, Java o C#.