

# Tecnologie Web

## C.d.L. in Informatica e Informatica per il Management

### Compito del 22 settembre 2014

**Nome:**

**Cognome:**

**Matricola:**

**C.d.L.:**

**Team:**

**Corso seguito nell'A.A.:**

Non è la prima volta. Data indicativa dell'ultimo appello provato: \_\_\_\_\_

#### **Importante:**

- Indicare ciascun esercizio risolto con una croce sul numero relativo. Saranno corretti solo gli esercizi così segnati.
- Se si consegna un foglio protocollo, scrivere in maniera chiara il numero di ogni esercizio presso la sua soluzione e, in intestazione, indicare nome, cognome e numero di matricola.

<b>Esercizio</b>		<b>Punti</b>	<b>Voto</b>
1	Domande di base	12	
2	HTML	6	
3	Javascript	6	
4	Semantic Web	6	
5	Teoria	4	
<b>Totale</b>		<b>34</b>	

## 1. Domande di base (12 punti)

Rispondere correttamente a tre delle seguenti domande:

A. Quali dei seguenti non si riferisce al Semantic Web:

- Linked Data
- Web Ontology Language
- Transmission Control Protocol
- Resource Description Framework
- Internet Protocol

B. Scrivere una regola CSS che associ testo bianco su sfondo nero a tutti gli elementi *span* di classe *invert*.

C. Quali tra le seguenti stringhe occupano un numero di caratteri dispari, se codificate in UTF-8 ?

- Città
- città!
- In città?
- Dove?
- Lì!
- CITTA'

D. Fornire una breve definizione per *AJAX*.

## 2. HTML (6 punti)

Scrivere il codice HTML 5 (e CSS) di un documento che rappresenti il contenuto di questa immagine:



# Accursio

Ristorante

Accursio è un porto su un fiume di storia. Nel cuore del cuore di Modica, nei bassi di un vecchio palazzo del Corso, ora c'è una piccola cucina per raccogliersi e godere di secoli di cultura. In questo luogo ci si ferma per attraversare le Sicilie di mare e di terra. Tanto del fascino della Sicilia è come custodito in uno scrigno; da Accursio vi si offre una chiave.

## Prenotazioni

Cliente

Nome e cognome:

Orario:

Numero di persone:

Scelta Menu

Menu di mare:

Menu di terra:

Menu di stagione:

## Menu di mare

### Antipasto

Un tuffo nel pomodoro

*Succo di pomodoro pachino con gamberi rossi, ricci di mare freschi di giornata, mozzarella di pecora, mentuccia e basilico. E anche bla bla bla bla bla*



---

### Primo



Spremuta di Sicilia

*Linguine con acciuga, scorza di arancia candita, peperoncino, cipollotto e scaglie di pane tostato. In abbinamento un calice di Marsala. Omaggio della cantina Rallo*

---

### Secondo

Cotoletta di pesce San Pietro

*Code di rana pescatrice con insalata di ortaggi, uva candita e mentuccia. Inoltre bla bla bla, bla bla bla, e soprattutto bla bla bla. Bla bla bla. Bla bla bla.*



---

### Dessert



L'uovo a la coque

*Gelatina di latte di mandorla con passion fruit e biscotto di mandorla con cedro, zuccata e arancia. Bla bla bla. Bla bla bla. Bla bla bla. Bla bla bla. Bla bla bla.*

Nello scrivere il codice si tengano in considerazione questi vincoli:

- i contenuti della pagina sono 'statici' (eventuali comportamenti dinamici fanno parte dell'esercizio Javascript)
- non è possibile usare tabelle di layout
- non è possibile usare elementi `i`, `b`, `strong`, `em`
- nessun elemento deve contenere l'attributo `style`
- non è possibile usare l'elemento `LINK` (usare un CSS "interno")
- le dimensioni esatte di margini e padding non sono rilevanti
- usare URL a piacere per le immagini
- nella sezione "prenotazioni", il campo "orario" prevede intervalli di 30 minuti dalle ore 19:00 alle 22:00, e il campo "numero di persone" valori da 1 a 15
- non è possibile usare l'attributo `id`, nè di conseguenza farvi riferimento nel foglio di stile
- non è possibile usare l'attributo `class` per posizionare le quattro immagini relative alla descrizione del menu di mare, nè di conseguenza farvi riferimento nel foglio di stile
- i dati del form sono spediti allo script `http://menu.example.com/prenotazione.php`.

### 3. Javascript (6 punti)

Si consideri la pagina descritta nell'esercizio precedente. Scrivere, utilizzando un framework Ajax a scelta tra ExtJs e jQuery, il codice Javascript che permetta di visualizzare i menu disponibili ed effettuare una prenotazione inserendo i propri dati, l'orario, il numero di coperti e le preferenze relative ai menu proposti. In particolare si tenga presente che:

- Il codice Javascript deve aggiornare le opzioni selezionabili nella sezione "prenotazioni" relative ad ogni menu in funzione del numero totale di coperti (campo "numero di persone" nella sezione cliente) e delle scelte fatte dall'utente per gli altri tipi di menu. Inoltre si deve impedire di inviare una prenotazione qualora la somma dei menu scelti non sia coerente con il numero di coperti (disabilitando il pulsante di prenotazione, o mostrando un messaggio di errore, ecc.)
- Il codice Javascript deve far apparire un riquadro con il menu disponibile ogni volta che si modifica il numero di coperti relativi ad uno dei menu disponibili. Il riquadro deve scomparire nel momento in cui ci si sposta nella casella di testo usata per lasciare i propri dati.
- La lista dei menu disponibili può essere reperita all'indirizzo <http://menu.example.com/listamenu>. Tale servizio remoto fornisce tali informazioni in formato JSON tramite documenti in questo formato:

```
[
  {
    "nome": "Menu di mare",
    "antipasto": {
      "nome": "Tuorlo d'uovo fritto",
      "descrizione": "Bla bla bla."
    },
    "primo": {
      "nome": "Crema di cannellini profumata allo scalogno",
      "descrizione": "Bla bla bla."
    },
    "secondo": {
      "nome": "Brasato di barolo",
      "descrizione": "Bla bla bla."
    },
    "dessert": {
      "nome": "Cupcakes salati al parmigiano",
      "descrizione": "Bla bla bla."
    }
  },
  ...
]
```

- Le immagini relative ad ogni menu possono essere reperite interrogando il servizio [http://menu.example.com/getimage/\[nomemenu\]](http://menu.example.com/getimage/[nomemenu]), dove nomemenu è uno fra quelli elencati nel documento JSON contenente la lista dei menu.

## 4. Semantic Web (6 punti)

Considerate le seguenti risorse e proprietà definite in Turtle mediante statement RDF (prefissi non dichiarati):

```
# Describe qualsiasi documento
foaf:Document a owl:Class .

# Describe atti citazionali tra documenti
cito:CitationAct a owl:Class .

# Indica il tipo di citazione che esiste tra documenti
cito:hasCitationCharacterization a owl:ObjectProperty ;
    rdfs:domain cito:CitationAct ;
    rdfs:range cito:CitationType .

# Identifica il documento citato
cito:hasCitedEntity a owl:ObjectProperty ;
    rdfs:domain cito:CitationAct ;
    rdfs:range foaf:Document .

# Identifica il documento citante
cito:hasCitingEntity a owl:ObjectProperty ;
    rdfs:domain cito:CitationAct ;
    rdfs:range foaf:Document .

# Permette di associare un titolo ad un documento
dcterms:title owl:DatatypeProperty ;
    rdfs:domain foaf:Document ;
    rdfs:range xsd:string .
```

Rispondere alle seguenti domande:

1. In un linguaggio a scelta tra Turtle e RDF/XML, aggiungere i seguenti individui come istanze della classe *cito:CitationType*: *cito:agreesWith*, *cito:disagreesWith*, *cito:extends*, *cito:usesMethodIn*. Inoltre descrivere anche la seguente frase in linguaggio naturale:

*Il documento (citante) intitolato "Evaluating citation functions in CiTO: cognitive issues" estende il documento (citato) intitolato "Characterising citations in scholarly articles: an experiment".*

2. Supponendo di aver realizzato molti atti citazionali simili a quello definito al punto precedente, scrivere una query SPARQL che restituisca un **booleano**: *true* nel caso in cui un articolo intitolato "Tracking changes through EARMARK: a theoretical perspective and an implementation" abbia almeno tre atti citazionali di tipo "cito:agreesWith", *false* altrimenti.
3. Usando un linguaggio a scelta tra PHP e Python, caricare il file "triples.rdf", contenente le triple definite precedentemente, e salvarlo in formato Turtle nel file "triples.ttl".

## 5. Teoria (4 punti)

Rispondere ai seguenti quesiti su CSS, eventualmente commentando esempi di fogli di stile:

- Cosa sono le regole *at rules* (precedute dal simbolo '@') ?
- A cosa servono le *media query*?