

Progettazione di Applicazioni Web T
Prova d'Esame – 12 luglio 2021

Tempo a disposizione: 120 minuti

La soluzione comprende la consegna elettronica dei seguenti file:

Foto.zip	file zip contenente il sorgente java/class, file XML e txt per il punto 1
Acquisto.zip	file zip contenente il sorgente java/class e txt per il punto 2
WebRatio.zip	file zip contenente il testo di risposta (txt/doc/pdf) al punto 3

Ogni file .zip consegnato DEVE CONTENERE TUTTI e SOLI i file creati/modificati e/o ritenuti importanti in generale ai fini della valutazione (ad esempio, codice java e relativi .class, descrittori XML, file txt/doc/pdf, ecc.) e NON dell'intero progetto.

N.B. La prova si intende superata se il punteggio finale non è inferiore a 18 punti (su un totale di 33). In particolare, è necessario (ma non sufficiente) ottenere la sufficienza in tutti gli esercizi.

ESERCIZIO 1 - 14 punti (sufficienza: 7 punti)

Si progetti una grammatica **XML Schema**, e un **documento XML** di esempio, per la modellazione delle informazioni relative ad **Album Fotografici**, nel rispetto delle seguenti specifiche:

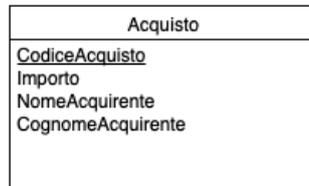
- Gli album fotografici possono essere di due tipi: album digitali in formato book (“DB”) o album digitali in formato calendario (“DC”);
- Ogni album fotografico, sia di tipo “DB” che di tipo “DC”, è descritto da informazioni obbligatorie quali l’autore, il titolo, la data di inizio creazione album, la data di fine creazione album e costituito da una o più fotografie in formato JPG;
- L’autore è costituito da un nome e da un cognome;
- Ogni fotografia è descritta dalla data di scatto, dal nome del file JPG sorgente, da uno o più tag testuali che ne riassumono il contenuto (campo opzionale) e dal luogo in cui la foto è stata scattata;
- Il luogo è costituito dal nome del luogo (ad esempio “Bologna” o “Italia”) e dalle sue coordinate geografiche GPS (in termini di latitudine e longitudine).

Si realizzi quindi l'**applicazione Java “Foto”** che, facendo uso del **parser SAX** e del **documento XML di esempio** realizzato al punto precedente, esponga il metodo `getFotoBookBologna()`, unitamente a un suo `main` di prova in grado di restituire: *“Autore e titolo degli album fotografici digitali in formato book (“DB”) contenenti fotografie scattate a Bologna”*. Stampare il risultato prodotto dal metodo `getFotoBookBologna()` opportunamente formattato sul file **Foto.txt**.

Progettazione di Applicazioni Web T
Prova d'Esame – 12 luglio 2021

ESERCIZIO 2 - 14 punti (sufficienza: 7 punti)

Partendo dalla realtà illustrata nel diagramma UML di seguito riportato, si fornisca una soluzione alla gestione della persistenza basata su **JDBC “Forza Bruta”** in grado di “mappare” efficientemente e con uso di ID surrogati il modello di dominio rappresentato dal **JavaBean “Acquisto”** del **diagramma UML** con la corrispondente **tabella relazionale derivata dalla progettazione logica del diagramma** stesso.



Nel dettaglio, dopo aver creato da applicazione Java le tabelle all'interno del proprio **schema** nel database **TW_STUD** di **DB2** (esplicitando tutti i **vincoli** opportuni), implementato **JavaBean** e metodi necessari per la realizzazione delle **operazioni CRUD**, si richiede la **implementazione di un metodo** in grado di restituire: *“L'elenco dei codici di acquisto con importo superiore ad una certa soglia”* (il valore di soglia è fornito come parametro di input al metodo).

Si richiede infine la realizzazione di una **classe main di prova** in grado di:

- **istanziare** almeno tre JavaBean “Acquisto” rendendoli persistenti rispetto alla base di dati associata al diagramma UML;
- **modificare** il costo di uno dei tre acquisti inseriti;
- **utilizzare** correttamente il metodo realizzato al punto sopra per **selezionare** i codici degli acquisti di importo superiore a 1000 euro; **produrre una stampa opportunamente formattata del risultato ottenuto** sul file **Acquisto.txt**;
- **eliminare** gli acquisti restituiti al punto precedente;

il tutto mediante opportuna demarcazione delle transazioni.

ESERCIZIO 3 - 5 punti (sufficienza: 3 punti)

Considerando l'ipertesto progettato nello **schema IFML di WebRatio** di seguito riportato, il candidato descriva le relazioni esistenti tra i componenti pagina **pA**, **pB** e **pC** e l'area **aA** presenti all'interno della site view **SV**.

