

Progettazione di Applicazioni Web T
Prova d'Esame – 18 giugno 2021

Tempo a disposizione: 120 minuti

La soluzione comprende la consegna elettronica dei seguenti file:

Asporto.zip	file zip contenente il sorgente java/class, file XML e txt per il punto 1
Aglio.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 suo **documento XML** di esempio, per la modellazione delle informazioni relative al servizio on-line “**Asporto**”, nel rispetto delle seguenti specifiche:

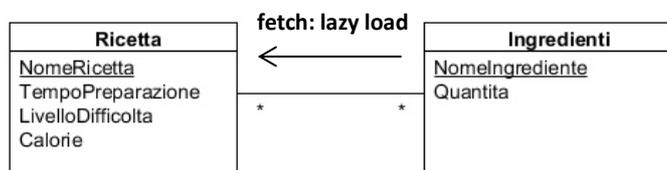
- Il servizio comprende i migliori ristoranti di una data località presso i quali l’utente può effettuare un’ordinazione di portate pronte al consumo.
- Ogni ristorante è caratterizzato da un nome (obbligatorio), da una località (obbligatoria), da un indirizzo completo (obbligatorio) - composto da via, numero civico e CAP - da un numero di telefono (obbligatorio) - costituito da prefisso e numero - e da un link a sito web dedicato se presente (opzionale).
- Inoltre, ogni ristorante espone le proprie offerte dividendo i piatti in cinque diverse categorie; nello specifico, antipasti, primi, secondi, contorni e dessert. Ogni categoria riporta l’elenco dei nomi dei piatti offerti e il relativo prezzo.

Si realizzi quindi l’applicazione Java “**Asporto**” che, facendo uso del **parser DOM** e del **documento XML di esempio** realizzato al punto precedente, esponga il metodo `getNumeroTotaleOfferte ()`, unitamente a suo un `main` di prova in grado di calcolare il **numero totale di portate offerte da ogni ristorante** (sommando su tutte le categorie). Stampare il risultato prodotto dal metodo `getNumeroTotaleOfferte` sul file **Asporto.txt**.

Progettazione di Applicazioni Web T Prova d'Esame – 18 giugno 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 **Pattern DAO** in grado di “mappare” efficientemente e con uso di ID surrogati il modello di dominio rappresentato dai **JavaBean Ricetta e Ingredienti del diagramma UML** con le corrispondenti **tabelle relazionali derivate dalla progettazione logica del diagramma** stesso.



Nel dettaglio, dopo aver creato da applicazione Java gli **scemi delle tabelle** all'interno del proprio schema nel database **TW_STUD** di **DB2** (esplicitando tutti i **vincoli** opportuni), **implementato i JavaBean e realizzato le classi** relative al **Pattern DAO** per l'accesso **CRUD** alle tabelle, si richiede l'**implementazione di un metodo in grado di restituire i nomi delle ricette che contengono l'ingrediente “aglio”**.

Si crei poi un **main di prova** in grado di:

- inserire due o più tuple nelle tabelle di interesse;
- fare uso corretto del metodo realizzato al punto precedente al fine di produrre la stampa del risultato, opportunamente formattata, sul file **Aglio.txt**.

N.B. L'implementazione del **Pattern DAO** deve **limitarsi** al solo **DBMS DB2**. La soluzione deve **sfruttare il mapping N-M** specificato nello schema UML e **propendere per il caricamento dei dati indicato**. Ogni ulteriore scelta da parte dello studente deve essere opportunamente giustificata con commenti nel codice.

ESERCIZIO 3 - 5 punti (sufficienza: 3 punti)

Considerando l'ipertesto progettato negli **scemi IFML di WebRatio a), b) e c)** di seguito riportati, il candidato descriva le relazioni esistenti tra i componenti pagina **pA**, **pB** e **pC**.

