

Cognome e Nome (stampatello): \_\_\_\_\_ Matricola: \_\_\_\_\_



I progetti funzionanti, con file Readme.txt che illustri le modifiche, vanno sottomessi su Piattaforma e-learning, in un unico archivio COGNOMENOME.zip entro le ore **15:00** del **25/11/2024**.  
Le ammissioni verranno comunicate sulla piattaforma di dipartimento e sul Teams della vostra classe.

**Riservato alla Commissione per la Correzione**

a	b	c	d	TOT	Commenti
15	6	3	6		

**TRACCIA**

Scrivere un insieme di Enterprise JavaBeans e client che rappresentino un archivio di nominativi di vini commercializzati dalla Enoteca "PerBaccocheVino". Per ogni tipo di vino commercializzato l'archivio mantiene le informazioni su ID (int), nominativo, vitigno, azienda, numero bottiglie disponibili, provenienza, prezzo, tipologia, acquisto (SI/NO).

a) Tramite JPA, si deve gestire l'archivio persistente dei vini su DB (EsameDB), dove la chiave primaria è la ID.

- Devono essere previste query per ID, prezzo, per vitigno, ed una query che restituisce tutte le entry del database
- Deve essere previsto un bean Singleton che inizializzi l'archivio
- Scrivere un client basato su invocazione di un bean stateless che prevede la stampa di:
  - Tutte le entry i cui corrispondenti prezzi siano inferiori di un importo dato in input
  - Tutte le entry di una Provenienza data in input
  - Tutte le entry del database.

b) Scrivere un interceptor che provvede a mantenere il numero di volte che ogni singolo metodo è stato chiamato stampando a video il nome del metodo chiamato, il numero di volte che è stato chiamato, il nome del metodo che ha il massimo numero di chiamate finora, e il numero massimo di chiamate. Esempio: "Invocato metodo findPROVENIENZA(), numero invocazioni 3. Metodo con massimo numero di invocazioni findALL(), numero invocazioni massime 7"

c) Scrivere un client basato su messaggi e la corrispondente parte lato server che invia un messaggio per aggiornare il numero di bottiglie di un determinato vino dato in input. Quando viene effettuato questo aggiornamento, tramite un evento viene stampato sulla console del server un messaggio di avviso "Aggiornamento effettuato". Inoltre, se il numero di bottiglie è inferiore a 10, viene stampato anche il messaggio "Il vino X deve essere riassortito", dove X rappresenta il nominativo della bottiglia di vino. Contestualmente verrà settato anche il campo acquisto a SI

c) Rendendo i metodi del bean invocabili come Web services, scrivere un client che stampi la lista di tutte le entry relative ai vini la cui tipologia sia "Champagne".

d) Descrivere i differenti tipi di query di JPQL.

[6 punti]

**Dati di esempio del DB**

ID	Nominativo	Vitigno	Azienda	Bottiglie	Provenienza	Prezzo	Tipologia	Acquisto
1	Montevetrano	Cabernet	Colli di Salerno	11	Italia	60	Rosso	NO
2	Terra di Lavoro	Aglianico	Galardi	15	Italia	50	Rosso	NO
3	Influence Miniere	Meunier	Rodolphe Miniere	5	Francia	SI	Champagne	NO

**NOTE:**

- Il DataSource deve chiamarsi: jdbc/EsameDS
- PersistentUnit e DB devono chiamarsi rispettivamente EsamePU ed EsameDB
- Il DatabasePopular deve prevedere la database definition
- La ConnectionFactory deve chiamarsi: jms/javaee7/ConnectionFactory
- Il topic deve chiamarsi: jms/javaee7/Topic

**ATTENZIONE:** si richiede di:

- Indicare a quali parti della prova (a, b, c, d) si risponde barrando le caselle apposite.
- Scrivere il codice tenendo presente le fondamentali richieste di ordine e buona strutturazione delle classi secondo i principi della programmazione e oggetti. Inoltre è necessario scrivere il codice seguendo le regole e le convenzioni di scrittura di programmi Java, non ultima la necessità di indentare correttamente e di commentare il codice.
- Procedere nella scrittura secondo il seguente ordine:
  - Entità, Interfacce, Database Producer, Database Populator, EJB, Client EJB

Cognome e Nome (stampatello): \_\_\_\_\_ Matricola: \_\_\_\_\_

- JMS: MDB e Client
- Web Services: Client

#### PER L'INVIO DEL PROGETTO, LA CORREZIONE E GLI ORALI

- Il progetti funzionanti, entro le ore **15:00 del 17/01/2023**, vanno sottomessi come da istruzioni fornite sulla Piattaforma di dipartimento al seguente link: **PD\_Esame\_12Gennaio2023**
- L'invio dei progetti entro la data stabilita è da considerarsi come richiesta di correzione (nel senso che se non viene inviato, il compito non viene corretto!).
- Regole per la consegna
  - I progetti da consegnare (progetti NetBeans) si devono chiamare "COGNOMENOMEXXX" nei rispettivi files COGNOMENOMEXXX.zip (devono essere esportati).
  - L'archivio che li contiene TUTTI insieme deve chiamarsi COGNOMENOME\_EE.zip
  - Deve esserci un UNICO file **readme.txt** (non **Readme**, non **leggi**, ...) all'interno di COGNOMENOME\_EE.zip che illustri le modifiche che si sono rese necessarie (per ogni file di ogni progetto) rispetto a quanto consegnato, secondo lo schema che viene pubblicato sulla piattaforma.
  - Le modifiche devono essere commentate nel codice dei progetti.
- Le ammissioni verranno fornite sul canale di "Annunci e Avvisi" del corso sulla piattaforma di dipartimento
- Gli orali si terranno a partire dalle date indicate sulla piattaforma del corso dopo la fase di correzione dei compiti del pre-appello.

**TERMINE PER L'INVIO DEL PROGETTO: ore 15 del 17/01/2023**

**AMMISSIONI ed ORALI pubblicati sulla piattaforma/Teams**