

Cognome e Nome (stampatello): _____ Matricola: _____

Scrivere in stampatello Cognome e Nome su ogni foglio	I progetti funzionanti dal nome "COGNOMENOMEXXX" in un file COGNOMENOMEXXX.zip con file Readme.txt che illustri le modifiche) vanno sottomessi su Piattaforma e-learning, in un unico archivio COGNOMENOME_EE.zip entro le ore 11:00 del 08/02/2022 Gli orali inizieranno orientativamente il 10/02/2022 e termineranno probabilmente nella settimana
---	--

Riservato alla Commissione per la Correzione

a	b	c	d	TOT	Commenti
10	7	5	8		

TRACCIA

Scrivere un insieme di Enterprise JavaBeans e client che rappresentino un archivio dati relativo al Festival di Sanremo, per la gestione dei cantanti in gara e dei voti ricevuti. Per ogni Canzone in gara, l'archivio contiene le informazioni su ID (int), Titolo, Genere (rock, pop, rap, altro), Cantanti (stringa che contiene i nomi dei cantanti), SerataEsibizione (intero da 1 a 4), Argomento (può essere "Amore", "Solitudine", "Disperazione", "Lagna", "Allegria", "Altro"), Voti popolari ricevuti (intero), Voti della Giuria Demoscopica (intero), Voti della Giuria di Qualità (intero).

- a) Tramite JPA, si deve gestire l'archivio persistente dei dati delle canzoni di Sanremo su DB (EsameDB), dove la chiave primaria è la ID.
- Devono essere previste query per Genere, per Cantanti, per Serata Esibizione ed una query che restituisce tutte le canzoni 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 canzoni che hanno ricevuto più di un certo numero (passato in input) di voti popolari
 - Tutti le canzoni che trattano di un argomento passato in input [10 punti].
- b) Scrivere un client basato su messaggi e la corrispondente parte lato server che invia un messaggio che serve per aggiornare i voti popolari, i voti della Giuria Demoscopica e i voti della Giuria di Qualità. I voti passati rappresentano l'aggiornamento al voto presente nell'archivio, e vanno quindi sommati a quelli precedentemente memorizzati. Si deve controllare che i voti siano aggiornati sempre con interi positivi e in caso di errore si stampa, tramite un evento, sulla console del server un messaggio di avviso "Errore" [7 punti].
- c) Rendendo i metodi del bean invocabili come Web services, scrivere un client che stampi tutti i dati della canzone che ha ottenuto il massimo numero di voti totali sommando quindi sia Voti popolari che della Giuria Demoscopica che della Giuria di Qualità. Si deve tenere presente la efficienza della soluzione dal punto di vista della banda di rete utilizzata. [5 punti].
- d) Descrivere gli Interceptor (3pt), Decorators (3pt) e la Iniezione di Dipendenze (2pt) [8 punti]

Dati di esempio del DB

ID	Titolo	Genere	Cantanti	SerataEsibizione	Argomento	Voti pop ricevuti	Voti Giuria ricevut	Voti Giuria di qualità
1	Brividi	Pop	Mahmood e Blanco	1	Amore	12000	10000	12323
2	O forse sei tu	Pop	Elisa	2	Solitudine	9000	21000	13100
3	Ogni volta è così	Pop	Emma	1	Disperazione	1100	19876	21000

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) 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, EJB, Client EJB
 - JMS: MDB e Client
 - Web Services: Client

Cognome e Nome (stampatello): _____ Matricola: _____

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

- Il progetti funzionanti, entro le ore **11:00 del 08/02/2022**, vanno sottomessi tramite la piattaforma Teams, seguendo le iscrizioni e il link, dove verranno anche fornite le ammissioni
- 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 **leggimi**, ...) 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.
- Il calendario degli orali verrà comunicato sul Teams del corso dell'Anno Accademico in corso. Gli orali si terranno a partire dalle date indicate sulla piattaforma Teams, il link vi verrà comunicato con le ammissioni. Si specifica che è possibile che i risultati siano disponibili solo poco prima di questa data, e che gli orali (salvo diversa comunicazione su piattaforma) inizieranno comunque in questa data.

TERMINE PER L'INVIO DEL PROGETTO: ore 11 del 08/02/2021

AMMISSIONI ed ORALI pubblicati sulla piattaforma/Teams