

Cognome e Nome (stampatello): _____ Matricola: _____

EMAIL DI CONSEGNA: pd01malandrino@gmail.com

Scrivere in stampatello Cognome e Nome su ogni foglio	I progetti funzionanti esportati in un file COGNOMENOME.zip con file Readme.txt che illustri le modifiche vanno sottomessi su Piattaforma e-learning entro le ore 15:00 del 31/01/2022 Gli orali inizieranno orientativamente il 4/2/2022, ma attendete informazioni sulla piattaforma e sul Teams dell'esame/del corso.
-------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Riservato alla Commissione per la Correzione

a	b	c	d	TOT	Commenti
12	8	4	6		

TRACCIA

Scrivere un insieme di Enterprise JavaBeans e client che rappresentino un archivio dati relativo ai candidati alle elezioni del Presidente della Repubblica Italiana per il periodo 2022-2028. Per ogni candidato l'archivio contiene le informazioni su ID (int), Cognome, Nome, Genere, Professione, Età, Numero Voti (intero), Numero Scrutinio (intero), Eletto ("SI", "NO").

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

- Devono essere previste query per ID, per età, per genere, 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:
 - Numero candidati di un genere fornito in input
 - Tutti i candidati che hanno meno di 60 anni

[12 punti]

b) Scrivere un client basato su messaggi e la corrispondente parte lato server che invia un messaggio per aggiornare, per un dato candidato, il numero di voti ricevuti ed eventualmente l'avvenuta elezione (campo Eletto="SI"). Il messaggio contiene: ID, il numero dello scrutinio ed il nuovo valore del campo voti che va sommato al valore precedente. Quando viene effettuato questo aggiornamento, tramite un evento viene stampato sulla console del server un messaggio di avviso "Dati aggiornati". Se il numero scrutinio è <=3 e si è raggiunti una maggioranza pari ai 2/3 dell'assemblea (673 voti su 1009), aggiornare opportunamente il valore del campo Eletto; tramite un evento verrà stampato il messaggio: "Il candidato NOME_COGNOME" è stato eletto nuovo Presidente della Repubblica". Se il numero scrutinio è > 3 e si è raggiunti la maggioranza assoluta (metà +1, 505 voti) aggiornare opportunamente il valore del campo Eletto; tramite un evento verrà stampato il messaggio: "Il candidato NOME_COGNOME" è stato eletto a maggioranza assoluta nuovo Presidente della Repubblica".

[8 punti]

c) Rendendo i metodi del bean invocabili come Web services, scrivere un client che stampi la lista di tutte le entry del database

[4 punti]

d) Dopo aver descritto il ciclo di vita di un MDB, descrivere perché sono stati introdotti i diversi livelli di affidabilità di JMS, descrivendo in particolare il recapito persistente ed il concetto dei Durable Consumers

[6 punti]

Dati di esempio del DB

ID	Cognome	Nome	Genere	Professione	Età	Numero voti	NumeroScrutinio	Eletto
1	Mattarella	Sergio	Uomo	Giurista	80	400	1	NO
2	Draghi	Mario	Uomo	Economista	74	100	1	NO
3	Cartabia	Marta	Donna	Costituzionalista	59	73	1	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.

Cognome e Nome (stampatello): _____ Matricola: _____

EMAIL DI CONSEGNA: *pd01malandrino@gmail.com*

- 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, DB management, Client EJB
 - 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 31/01/2022**, vanno sottomessi tramite la piattaforma di dipartimento (PD_28Gennaio2022), 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) devono essere esportati in un archivio che li contiene TUTTI e che deve chiamarsi COGNOMENOME.zip
 - Deve esserci un UNICO file **readme.txt** (non **Readme**, non **leggimi**, ...) all'interno di COGNOMENOME.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 sulla canale di “Annunci e Avvisi” 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 15 del 31 Gennaio 2022

AMMISSIONI ed ORALI pubblicati sulla piattaforma e sui Teams (Corso/esame)