

Cognome e Nome (stampatello): _____ Matricola: _____

Mail INVIO Compito: 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 17/02/2022 Gli orali inizieranno orientativamente il 21/02/2022 ma attendete conferme sulla piattaforma di dipartimento e sul Teams della vostra classe.
---	---

Riservato alla Commissione per la Correzione

a	b	c	d	e	TOT	Commenti
12	4	4	4	6		

TRACCIA

Scrivere un insieme di Enterprise JavaBeans e client che rappresentino un archivio dati relativo ai cani in adozione del Canile "AdottatoloSubito". Per ogni cane, l'archivio mantiene le informazioni su ID (int), razza (stringa), nome del cane (stringa), taglia (piccola, media o grande), sesso (maschio o femmina), età (intero), microchip (codice numerico - 0 se ancora non dotato di microchip), status di adozione (boolean: true se è stato già adottato).

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

- Devono essere previste query per ID, Razza, Status di adozione, 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:
 - Tutti i cani che non hanno ancora un microchip.
 - Tutti i cani che sono stati adottati.

[12 punti]

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 *findADOZIONE()*, numero invocazioni 3. Metodo con massimo numero di invocazioni *findALL()*, numero invocazioni massime 7".

[4 punti]

c) Scrivere un client basato su messaggi e la corrispondente parte lato server che invia un messaggio aggiungere una nuova entry nel database. Quando viene effettuato questo aggiornamento, tramite un evento viene stampato sulla console del server un messaggio di avviso "Nuovo inserimento effettuato, benvenuto NOME", dove NOME è il campo Nome del nuovo ospite del Canile.

[4 punti]

d) Rendendo i metodi del bean invocabili come Web services, scrivere un client che stampi la lista di tutti i cani di una taglia data in input.

[4 punti]

e) Dopo aver definito cosa è l'Entity Manager e come si ottiene, descrivere il concetto di Persistence Context.

[6 punti]

Dati di esempio del DB

ID	Razza	Nome	Taglia	Età	Sesso	Microchip	Status
1	Boston terrier	Leo	Media	1	M	0	false
2	Meticcio	ISI	Grande	3	F	941000001581506	true
3	Pinscher	Michelle	Piccola	7	F	981100000297746	true

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

Cognome e Nome (stampatello): _____ Matricola: _____

Mail INVIO Compito: pd01malandrino@gmail.com

ATTENZIONE: si richiede di:

- Indicare a quali parti della prova (a, b, c, d, e) 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 (se presenti):
 - Entità, Interfacce, EJB, Client EJB
 - JMS: MDB e Client
 - Interceptors
 - Web Services: Client

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

- Il progetti funzionanti, entro le ore **15:00 del 17/02/2022**, vanno sottomessi come da istruzioni fornite sulla Piattaforma Teams, 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 sulla canale dell'esame. 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 15:00 del 17/02/2022

AMMISSIONI ed ORALI pubblicati sulla piattaforma/Teams