

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

PREFERENZE PER ORALE (scrivere i soli giorni in cui non si è disponibili): \_\_\_\_\_

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 inviati sulla Piattaforma e-learning, in un unico archivio COGNOMENOME_EE.zip entro le <b>ore 23:00 del 9/2/2023</b> Gli orali inizieranno orientativamente dopo il <b>12/2/2023</b> , ma attendete conferme sulla piattaforma di dipartimento o da ESSETRE.
---	---

*Riservato alla Commissione per la Correzione*

a	b	c	d	TOT	Commenti
12	6	4	8		

## TRACCIA

Scrivere un insieme di Enterprise JavaBeans e client che rappresentino un archivio delle **case in affitto di un'agenzia immobiliare**. Per ogni immobile, l'archivio mantiene le informazioni su ID (int), tipologia (stringa), indirizzo (stringa), CAP (stringa), il prezzo di affitto (float), e lo stato di locazione (ovvero: locato, libero o non disponibile).

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

- Devono essere previste query per tipologia, CAP, ed una query che restituisca 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 gli immobili con prezzo di affitto inferiore ad una soglia (ad esempio: prezzo < 800 euro/mese)
  - Tutti gli immobili con prezzo compreso in un intervallo e in una determinata zona definita dal CAP (ad esempio: prezzo compreso tra i 400 e 600/mese in zona 84121)

[12 punti]

b) Scrivere un client basato su messaggi e la corrispondente parte lato server che invii un messaggio per aggiornare il prezzo di affitto e lo stato di locazione. Quando viene effettuato questo aggiornamento, viene stampato sulla console del server un messaggio di avviso "Aggiornamento immobile".

Se un immobile cambia lo stato, tramite un evento verrà stampato il messaggio "Immobile TIPO in INDIRIZZO ha cambiato lo stato in STATO".

[6 punti]

c) Rendendo i metodi del bean invocabili come Web services, scrivere un client che stampi tutti gli immobili liberi.

[4 punti]

d) Descrivere brevemente i seguenti concetti: architettura di JMS, ciclo di vita dei session bean, gli standard dietro ai Web Services, cinque tipi di query di JPA.

[8 punti]

## Dati di esempio del DB

ID	Tipologia	Indirizzo	CAP	Prezzo	Stato
1	Bilocale	Piazza Portanova	84121	1000	Libero
2	Trilocale	Via Posidonia	84129	800	Libero
3	Bilocale	Lungomare Marconi	84128	1000	Locato
4	Monolocale	Via Guercio	84134	600	Locato

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

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

PREFERENZE PER ORALE (scrivere i soli giorni in cui non si è disponibili): \_\_\_\_\_

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

- Il progetti funzionanti, entro le **ore 23:00 del 9/2/2023**, vanno sottomessi come da istruzioni fornite sulla sito di e-learning del corso, 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 sito di e-learning e/o inviate da email tramite ESSETRE.
- Gli orali si terranno a partire dalle date indicate. Si specifica che è possibile che i risultati siano disponibili solo poco prima di questa data, e che gli orali (salvo diversa comunicazione) inizieranno comunque in questa data.

**TERMINE PER L'INVIO DEL PROGETTO: ore 23:00 del 9/2/2023**

**AMMISSIONI ed ORALI pubblicati sulla piattaforma e-learning o inviate da ESSETRE**