

Esempio di Prova di Livello Base-Intermedio

Cognome e Nome: _____ Matricola: _____

Tempo a disposizione: 3 ore

È necessario scrivere un'applicazione per gestire un'anagrafica di aziende e dei relativi dipendenti. Ogni azienda è descritta da (a) partita iva (b) denominazione (c) la città della sede sociale. Ogni azienda ha inoltre un insieme di dipendenti, descritti da (a) codice fiscale (per semplicità di 6 cifre), (b) nome (c) cognome (d) data di assunzione, (e) regione di residenza.

L'applicazione deve consentire di svolgere i seguenti casi d'uso:

"Utente carica un archivio"

- l'utente carica l'archivio *NOTA: per velocizzare le operazioni, si suggerisce di utilizzare un "mock object" per simulare il funzionamento del DAO che carica i dati dal disco*

"Utente cerca azienda per città"

- l'utente inserisce il nome di una città
- il sistema mostra la lista delle aziende che hanno sede sociale in quella città, ordinate per denominazione crescente

"Utente inserisce nuovo dipendente"

- l'utente seleziona una azienda dall'elenco dei risultati
- il sistema mostra, in un nuovo pannello, i dati della società e una tabella con i dati dei suoi dipendenti. Per ogni dipendente bisogna mostrare codice fiscale, nome, cognome e data di assunzione.
- Se ci sono più utenti con lo stesso nome, cognome e data di assunzione, il sistema mostra un messaggio di avvertimento all'utente: "Attenzione: nell'azienda potrebbero esserci duplicati"
- l'utente può decidere di inserire i dati di un nuovo dipendente, specificando tutti i dati.

Scenario alternativo

- dati scorretti: il sistema mostra un messaggio di errore all'utente
- se esiste già un utente con quel codice fiscale nell'azienda, viene mantenuto tra i due il dipendente con la data di assunzione più recente

"Utente verifica archivio"

- utilizzando una voce di menu, l'utente può chiedere al sistema di verificare se nell'archivio ci sono grosse disparità regionali. In particolare, il sistema calcola, per ogni regione, il numero di dipendenti residenti in quella regione.
- Il sistema verifica se la regione più frequente ha più del triplo dei dipendenti della regione meno frequente. In questo caso il sistema mostra un messaggio del tipo "La regione Lombardia è la più frequente e ha più di tre volte il numero di dipendenti della Sardegna".

"Utente ricerca anni frequenti" – Per la prova di livello intermedio

- il sistema calcola, per ogni regione, l'anno più frequente in cui sono state effettuate le assunzioni
- i dati saranno mostrati in una tabella ordinata per anno crescente. Un esempio di tabella è la seguente
Nota: è necessario sviluppare i relativi test di regressione

Regione	Anno Frequente	Assunzioni Anno Frequente
Basilicata	1999	32 assunzioni nel 1999
Lombardia	2023	60 assunzioni nel 2023

Sviluppare l'applicazione che implementa i casi d'uso elencati, seguendo il processo di sviluppo descritto a lezione, e in particolare le seguenti operazioni:

- costruire e documentare il modello concettuale dell'applicazione
- sviluppare il frammento di applicazione che implementa i casi d'uso descritti
- sviluppare l'interfaccia grafica in **Java Swing** utilizzando l'**architettura MVC** presentata a lezione, rispettando i requisiti per l'accessibilità e l'usabilità.
- sviluppare i test di regressione utilizzando la libreria JUnit
- per effettuare operazioni di logging utilizzare le librerie SLF4J e LogBack
- utilizzare **Gradle** come sistema di costruzione del codice

Per le librerie, se necessario, utilizzare le seguenti dipendenze

- org.slf4j:slf4j-api:2.0.1
- ch.qos.logback:logback-classic:1.3.1
- ch.qos.logback:logback-core:1.3.1
- org.junit.jupiter:junit-jupiter:5.8.2

Per utilizzare Lombok, applicare il seguente plugin gradle

- id "io.freefair.lombok" version "6.5.1"

E' possibile trovare il materiale necessario nelle seguenti cartelle

- materiale didattico: **c:\documentazione\POO1 e POO2**
- file di esempio di logback.xml: **c:\documentazione\POO1 - materiale software aggiuntivo\java**