

Esercizio 1: Calcolatrice Semplice

Consegna: Realizza una calcolatrice con interfaccia grafica che permetta di eseguire le quattro operazioni aritmetiche fondamentali: addizione, sottrazione, moltiplicazione e divisione.

Requisiti:

- Crea una finestra con un display per visualizzare numeri e risultati
- Implementa pulsanti per le cifre da 0 a 9
- Implementa pulsanti per le quattro operazioni (+, -, *, /)
- Aggiungi un pulsante per il calcolo del risultato (=)
- Aggiungi un pulsante per cancellare il display (C)
- Gestisci il caso di divisione per zero mostrando un messaggio di errore

Suggerimenti:

- Utilizza un JTextField per il display
- Organizza i pulsanti in un pannello con GridLayout
- Usa gli ActionListener per gestire gli eventi dei pulsanti

Competenze da sviluppare:

- Uso di componenti base di Swing
- Gestione eventi
- Operazioni matematiche
- Gestione errori

Esercizio 2: Form di Registrazione

Consegna: Crea un form di registrazione che raccolga informazioni personali e verifichi la validità dei dati inseriti prima di confermare la registrazione.

Requisiti:

- Il form deve raccogliere: nome, cognome, email, password, città e genere
- Implementa una casella di controllo per l'accettazione dei termini e condizioni
- Verifica che tutti i campi siano compilati correttamente prima di accettare la registrazione
- L'email deve contenere @ e .
- La password deve essere di almeno 6 caratteri
- Deve essere selezionato un genere
- L'utente deve accettare i termini e condizioni

- Mostra messaggi di errore specifici per ciascun problema riscontrato

Suggerimenti:

- Utilizza diversi tipi di componenti: JTextField, JPasswordField, JComboBox, JRadioButton, JCheckBox
- Usa ButtonGroup per raggruppare i radio button
- Crea un metodo dedicato alla validazione del form

Competenze da sviluppare:

- Utilizzo di vari componenti Swing
- Validazione dati
- Gestione messaggi di errore
- Organizzazione layout

Esercizio 3: Lista della Spesa

Consegna: Realizza un'applicazione che permetta di gestire una lista della spesa con funzionalità di aggiunta, rimozione e salvataggio degli elementi.

Requisiti:

- Crea un'interfaccia con un campo di testo per inserire nuovi prodotti
- Implementa un pulsante per aggiungere prodotti alla lista
- Visualizza la lista dei prodotti in un componente scorrevole
- Aggiungi la possibilità di rimuovere prodotti selezionati
- Implementa un pulsante per pulire l'intera lista
- Aggiungi un pulsante per visualizzare un riepilogo della lista

Suggerimenti:

- Utilizza JList e DefaultListModel per gestire la lista
- Aggiungi un JScrollPane per rendere la lista scorrevole
- Implementa un KeyListener per permettere l'aggiunta di prodotti premendo Invio

Competenze da sviluppare:

- Gestione di collezioni di dati
- Interazione con liste grafiche
- Eventi da tastiera
- Manipolazione di stringhe

Esercizio 4: Editor di Testo Semplice

Consegna: Sviluppa un semplice editor di testo che permetta di creare, aprire, modificare e salvare file di testo.

Requisiti:

- Crea una finestra principale con un'area di testo modificabile
- Implementa una barra dei menu con le seguenti voci:
 - Menu File: Nuovo, Apri, Salva, Salva con nome, Esci
 - Menu Modifica: Taglia, Copia, Incolla, Seleziona tutto
 - Menu Formato: A capo automatico
- Prima di creare un nuovo documento o uscire, chiedi all'utente se vuole salvare le modifiche
- Salva i file in formato .txt

Suggerimenti:

- Utilizza JTextArea per l'area di testo
- Aggiungi un JScrollPane per rendere scorrevole l'area di testo
- Usa JFileChooser per la selezione dei file
- Implementa le operazioni di I/O con Reader e Writer

Competenze da sviluppare:

- Creazione di menu e barre degli strumenti
- Operazioni di I/O su file
- Gestione eventi di menu
- Dialog di conferma

Esercizio 5: Convertitore di Valute

Consegna: Crea un'applicazione che permetta di convertire importi tra diverse valute utilizzando tassi di cambio predefiniti.

Requisiti:

- Crea un'interfaccia con un campo per inserire l'importo da convertire
- Implementa due menu a tendina per selezionare la valuta di origine e destinazione
- Aggiungi un pulsante per eseguire la conversione
- Implementa un pulsante per scambiare le valute selezionate
- Mostra il risultato della conversione formattato correttamente
- Gestisci gli errori di input (es. inserimento di testo invece di numeri)

Suggerimenti:

- Utilizza tassi di cambio rispetto a una valuta base (es. Euro)

- Formatta i numeri utilizzando DecimalFormat
- Aggiungi un KeyListener per permettere la conversione premendo Invio

Competenze da sviluppare:

- Calcoli matematici
- Formattazione dei numeri
- Gestione di input utente
- Conversione di dati

Esercizio 6: Gioco del Snake Semplificato

Consegna: Implementa una versione semplificata del classico gioco Snake, in cui il giocatore controlla un serpente che si muove su un piano raccogliendo cibo e crescendo di lunghezza.

Requisiti:

- Crea un'area di gioco con sfondo nero
- Implementa un serpente rappresentato da blocchi verdi
- Genera cibo (rappresentato da un cerchio rosso) in posizioni casuali
- Permetti il controllo del serpente usando i tasti freccia
- Implementa il rilevamento delle collisioni (con i bordi e con se stesso)
- Mostra il punteggio durante il gioco
- Visualizza un messaggio di "Game Over" quando il gioco termina
- Aggiungi un pulsante per iniziare una nuova partita

Suggerimenti:

- Usa un pannello personalizzato con override del metodo paintComponent
- Implementa un timer per gestire il movimento del serpente
- Usa array per memorizzare le coordinate del serpente
- Aggiungi un KeyListener per rilevare gli input da tastiera

Competenze da sviluppare:

- Animazione grafica
- Gestione di timer
- Rilevamento collisioni
- Logica di gioco

Esercizio 7: Gestione Studenti

Consegna: Sviluppa un'applicazione per la gestione degli studenti di una classe, con possibilità di aggiungere, modificare ed eliminare dati e calcolare la media dei voti.

Requisiti:

- Crea un'interfaccia divisa in due sezioni:
 - A sinistra: form per l'inserimento/modifica dei dati (matricola, nome, cognome, voto)
 - A destra: tabella per visualizzare gli studenti registrati
- Implementa pulsanti per:
 - Inserire un nuovo studente
 - Modificare uno studente selezionato
 - Eliminare uno studente selezionato
 - Calcolare la media dei voti
- Verifica la validità dei dati inseriti (campi obbligatori, voto tra 0 e 30)
- Gestisci il caso di matricola duplicata chiedendo conferma per l'aggiornamento
- Chiedi conferma prima di eliminare uno studente

Suggerimenti:

- Utilizza un JSplitPane per dividere l'interfaccia
- Implementa JTable con DefaultTableModel per visualizzare i dati
- Crea una classe Studente per rappresentare i dati
- Usa ArrayList per memorizzare gli studenti

Competenze da sviluppare:

- Gestione di tabelle di dati
- Organizzazione avanzata del layout
- Manipolazione di oggetti
- Calcoli statistici semplici

Sfide Aggiuntive per Tutti gli Esercizi

Per ciascun esercizio, considera di implementare una o più delle seguenti sfide aggiuntive:

1. **Persistenza dei dati:** Salva i dati su file per mantenerli tra le diverse sessioni dell'applicazione.
2. **Personalizzazione grafica:** Aggiungi la possibilità di cambiare colori, font o temi dell'interfaccia.
3. **Internazionalizzazione:** Implementa il supporto per più lingue.
4. **Scorciatoie da tastiera:** Aggiungi combinazioni di tasti per le operazioni più comuni.
5. **Validazione avanzata:** Implementa controlli più sofisticati sui dati inseriti.
6. **Responsive design:** Assicurati che l'interfaccia si adatti correttamente a diverse dimensioni della finestra.
7. **Documentazione:** Aggiungi JavaDoc completo alle classi e ai metodi.

8. **Testing:** Crea unit test per verificare il corretto funzionamento dell'applicazione.