

Prova Esame di Laboratorio

12 Settembre 2025

La prova prevede lo sviluppo di un portale per la gestione di itinerari. Dovrà essere svolta a partire dal progetto Flask di base contenuto nel file

- Una route `/index` con un template HTML di base.
- Un file `places.csv` nella cartella data, contenente i luoghi visitabili.
- Un file `itineraries.csv` nella cartella data, contenente i dati relativi agli itinerari disponibili.
- Un file `style.css` per lo stile.

Il file `places.csv` contiene le seguenti colonne: `place_id`, `name`, `category`, `municipality`, `image_link`, `price`.

Il file `itineraries.csv` contiene le colonne: `itinerary_id`, `name`, `duration`, `place_list`. L'attributo `place_list` ha come valore una stringa contenente una lista python con tutti `place_id` dei luoghi compresi nell'itinerario.

Il codice aggiunto deve essere commentato con **ESERCIZIO #<numero esercizio>**.

A partire da questo progetto si chiede:

1. (3pt) Scrivere due funzioni per leggere e restituire:
 - a. Tutti le opere presenti nel file `places.csv`.
 - b. Tutte i prestiti inseriti nel file `itineraries.csv`.

Le funzioni devono essere richiamate, dove necessario, in `app.py` per ottenere l'elenco dei luoghi e degli itinerari. Il caricamento dei file deve essere implementato in un'unica parte del codice; in caso contrario, verranno sottratti punti, anche se il caricamento è corretto in tutte le parti sviluppate.

2. (2pt) Modificare la route `/index` per renderizzare un template `index.html`, che estenda il template base. Il template deve mostrare una tabella di tutti i luoghi visitabili con le seguenti colonne: nome, comune e categoria.
3. (6pt) Rendere cliccabili le righe della tabella. Quando un utente clicca su un'opera, reindirizzarlo a una pagina dedicata `/place/<place_id>` che mostri l'immagine del luogo e un riquadro con i seguenti dettagli: nome, comune, categoria, prezzo di visita e numero di itinerari in cui è incluso.
4. (5pt) Nella pagina index, sopra la tabella deve essere presente un bottone "Aggiungi luogo", che reindirizza a una pagina con un form per l'aggiunta di un nuovo luogo. Deve essere possibile inserire: nome, categoria, comune, il link di una immagine e il prezzo di visita. Il `place_id` deve essere generato automaticamente lato backend.

5. **(6 pt)** Creare due API:

- Una per restituire tutti gli itinerari in formato JSON.
- Una per restituire tutti luoghi compresi in un itinerario specifico.

Non è richiesto che le API restituiscano anche le colonne del file, ma solo un dizionario contenente i dati delle valutazioni.

6. **(2 pt)** Creare una route `/react` lato Flask per renderizzare il template `index_react.html`, dove costruire una Single Page Application (SPA) con React. Tutte le route impostate con React devono seguire il prefisso `/react`, ad esempio `/react/place`, `/react/itinerary`, ecc.
7. **(4 pt)** Creare un componente React `ItineraryList` per visualizzare l'elenco degli itinerari in una tabella che mostri il nome dell'itinerario, la durata e il prezzo, dato dalla somma dei prezzi dei luoghi compresi nell'itinerario. La componente deve contenere un bottone o un link "Crea Itinerario". Associare questo componente alla route `/react/itineraries`.
8. **(5 pt)** Il bottone "Crea itinerario" deve portare alla visualizzazione di un componente `NewItinerary`, contenente il form per aggiungere un nuovo itinerario. Il form deve permettere di selezionare tra i luoghi presenti nel file `places.csv`. Il codice deve essere generato automaticamente lato backend all'interno di un'API Flask creata appositamente per aggiungere l'itinerario al file `itineraries.csv`.

Per consegnare la prova di laboratorio

- Creare un'unica cartella compressa (ZIP, o RAR), denominata `Cognome_Nome_Matricola.estensione` (sostituire Cognome, Nome e Matricola con i propri dati e sostituire estensione con l'estensione utilizzata, es. zip o rar)
- La cartella compressa deve contenere le cartelle iniziali (*data*, *static*, *templates*), e i file richiesti nel testo dell'esame posizionati nelle cartelle apposite.
- Caricare su MOODLE il file zip
- Nella sezione "Commenti" su Moodle inserire il numero degli esercizi svolti (es. "Svolti gli esercizi numero 1,2,3,4,5,...")