**Basi di Dati (16AFQPL, 16AFQPI)**

**Anno Accademico 2024-2025**

**Politecnico di Torino**

# Utilizzo di Large Language Models per Text2SQL

## Obiettivo

La finalità di questa esercitazione è quella di, dato uno schema logico-relazionale, generare coppie di domanda e risposta relative al task Text2SQL (ovvero formulare, a partire da un testo in linguaggio naturale, la corrispondente query SQL).
L'esercitazione prevede di generare coppie di domanda e risposta in lingua italiana relative a pattern di difficoltà differenti, testando le abilità di tre differenti Large Language Models (LLMs) e confrontando le soluzioni generate con diversi prompt con la soluzione attesa.

## Descrizione della base di dati

La base di dati da utilizzare è denominata *Biblioteca* e raccoglie le informazioni relative alla gestione di una biblioteca ed è progettata per gestire e tenere traccia di libri, autori, lettori, prestiti, dipendenti e prenotazioni delle sale.

**Schema logico relazionale:**
LIBRO(id_libro, *id_autore*, titolo, anno_pubblicazione, genere, numero_copie)
AUTORE(id_autore, nome, cognome, nazionalità, data_nascita, data_morte*)
LETTORE(id_lettore, nome, cognome, email, telefono, indirizzo, data_iscrizione)
PRESTITO_LIBRO(*id_libro*, *id_lettore*, data_prestito, data_restituzione*, note*)
SALA(id_sala, *id_dipendente_responsabile*, nome, capacità, piano, tipologia)
DIPENDENTE(id_dipendente, nome, cognome, ruolo, data_assunzione, email, telefono)
PRENOTAZIONE_SALA(id_lettore, data, ora_inizio, ora_fine, *id_sala*, note*)

Le chiavi primarie sono sottolineate e le chiavi esterne sono in *corsivo*. L'asterisco indica i campi opzionali.

La tabella **Libro** raccoglie le informazioni su ogni libro disponibile nella biblioteca. Ogni libro è identificato da un codice univoco (`id_libro`) ed è caratterizzato dall'autore (`id_autore`), titolo (`titolo`) e anno di pubblicazione (`anno_pubblicazione`), genere letterario (`genere`) e numero di copie disponibili (`numero_copie`). Si suppone che un autore possa scrivere più libri ma che un libro sia scritto da un unico autore.

La tabella **Autore** memorizza le informazioni degli autori. Ogni autore è identificato da un codice univoco (`id_autore`) ed è caratterizzato dal nome (`nome`), cognome (`cognome`), nazionalità (`nazionalità`), data di nascita (`data_nascita`) ed eventuale data di morte (`data_morte`).

La tabella **Lettore** contiene le informazioni degli utenti registrati alla biblioteca. Ogni lettore è identificato da un codice univoco (`id_lettore`) ed è caratterizzato dal nome (`nome`), cognome (`cognome`), indirizzo email (`email`), numero di telefono (`telefono`) e indirizzo di residenza (`indirizzo`). Inoltre, viene registrata la data di iscrizione alla biblioteca (`data_iscrizione`).

La tabella **Prestito_Libro** registra i prestiti dei libri effettuati dai lettori. Ogni prestito è identificato dalla combinazione dell'identificativo del libro prestato (`id_libro`), del lettore che lo ha richiesto (`id_lettore`) e dalla data di inizio prestito (`data_prestito`). Si memorizza inoltre la data di restituzione (`data_restituzione`) ed eventuali note o informazioni aggiuntive (`note`).

La tabella **Sala** raccoglie le informazioni sulle sale presenti nella biblioteca. Ogni sala è identificata da un codice univoco (`id_sala`) e ha un dipendente responsabile assegnato (`id_dipendente_responsabile`). Le sale sono caratterizzate da un nome (`nome`), una capacità massima di persone (`capacità`), il piano in cui si trovano (`piano`) e la tipologia di utilizzo (`tipologia`).

La tabella **Dipendente** memorizza le informazioni sui lavoratori della biblioteca. Ogni dipendente è identificato da un codice univoco (`id_dipendente`) ed è caratterizzato dal nome (`nome`), cognome (`cognome`), ruolo svolto nella biblioteca (`ruolo`), data di assunzione (`data_assunzione`), indirizzo email (`email`) e numero di telefono (`telefono`).

Infine, la tabella **Prenotazione_Sala** registra le prenotazioni delle sale effettuate dai lettori. Ogni prenotazione è identificata dalla combinazione dell'identificativo del lettore che l'ha prenotata (`id_lettore`) e dalla data (`data`) e ora di inizio (`ora_inizio`) della prenotazione. Si registra inoltre l'ora di fine della prenotazione (`ora_fine`), l'identificativo della sala prenotata (`id_sala`) ed eventuali note o informazioni aggiuntive sulla prenotazione (`note`).

## Svolgimento

I form da compilare per l'esercitazione sono predisposti per memorizzare i seguenti campi:

- **Pattern di difficoltà**: pattern caratterizzante la query proposta
- **Domanda**: query proposta in linguaggio naturale (in lingua italiana)
- **Soluzione attesa**: soluzione attesa (in SQL) rispetto alla domanda proposta
- **Prompt**: testo in lingua italiana da inviare all'LLM
- **GPT-4o/Gemini/Codestral**: codice SQL generato come risposta dai rispettivi LLM
- **Analisi GPT-4o/Gemini/Codestral**: commento dello studente alla soluzione proposta dai rispettivi LLM

Al fondo del secondo form è presente un campo nel quale lo studente può riportare un commento generale sulle principali risultanze sperimentali evidenziate tra cui, ad esempio:

- Punti forti e punti deboli di ciascun LLM
- Confronti tra LLM
- Analisi e confronti tra tipologie diverse di prompt

Durante il laboratorio, a ciascuno studente è richiesto di:

- Creare **una domanda per ogni tipologia di pattern** indicata nel form.
- Scrivere la soluzione attesa alla query proposta.
- Formulare i prompt testuali da inviare all'LLM.

  Occorre partire dalla struttura del prompt fornito in calce e completarlo per ogni tipologia di pattern richiesto. I prompt devono essere scritti in **lingua italiana** e devono includere lo schema logico delle tabelle della base dati.

  **Nota**: Per alcuni pattern (3, 6, 7), per rispondere alla stessa domanda occorre preparare versioni diverse del prompt in cui chiedere esplicitamente all'LLM di risolvere la query utilizzando una determinata strategia (es. tramite Join, IN, EXISTS, INTERSECT).
- Interrogare **i tre Large Language Models** utilizzando i prompt generati:
  - **GPT-4o:** https://chatgpt.com/
  - **Gemini:** https://gemini.google.com/app
  - **Codestral:** https://chat.mistral.ai/chat

  **Nota**: Prima di interrogare Codestral, disattivare dall'interfaccia web tutte le funzionalità che risultano attive cliccando su "Strumenti" e togliendo le spunte corrispondenti.
- Inserire l'output fornito dai vari LLM all'interno dei campi di testo delle corrispondenti domande del form (per l'esercizio 0 di esempio, che bisogna completare, si può prendere spunto dall'esempio di prompt alla fine del testo).
- **Analizzare** le risposte fornite dai modelli e riportare i relativi commenti nei campi predisposti.

Per poter utilizzare i vari modelli, è consigliato registrarsi tramite la propria e-mail personale (es. utilizzando l'indirizzo email PoliTo) o effettuare l'accesso tramite account terzi (es. Google).

Lo svolgimento del laboratorio prevede **due fasi distinte**, ciascuna delle quali consente di ottenere i punti associati all'homework. Il punteggio totale massimo assegnabile al primo homework è **1 punto**, **diviso equamente tra le due fasi**.

## Fase 1 – in laboratorio (0.5 punti)

- Seguendo le indicazioni degli esercitatori di laboratorio, collegarsi alle interfacce web di ciascun LLM e generare le coppie di domanda e risposta con i relativi prompt, soluzioni e commenti.
- Prima del termine dell'esercitazione, inviare il form adibito al laboratorio compilato arrivando all'ultima pagina e premendo **INVIA**.
    - Se il form non è stato completato nello slot di laboratorio assegnato, **salvare** le risposte inviate cliccando sul pulsante "Salva la mia risposta per modificare" mostrato dopo l'invio del modulo. Se non si effettua il salvataggio delle risposte inviate, non sarà possibile continuare la compilazione in un secondo momento.
    - Per continuare la compilazione, accedere a Microsoft Forms da forms.office.com.
      Si troverà il modulo inviato nella scheda "Recente" o "Moduli compilati".
- La consegna è ritenuta valida **se e solo se**:
    - Il form è stato inviato;
    - Contiene le domande, risposte e i relativi prompt, soluzioni, e commenti degli esercizi **da 0 a 3** (inclusi);
    - Data e orario di consegna non superano l'orario di conclusione del proprio slot di laboratorio.
- La **consegna del primo form** può essere effettuata in modo **individuale** o a **coppie**.
    - **Individuale**: compilare i campi "Nome/Cognome/Matricola/Email (1)", lasciando vuoti i campi "Nome/Cognome/Matricola/Email (2)".
    - **A coppie**: compilare sia i campi "Nome/Cognome/Matricola/Email (1)" che i campi "Nome/Cognome/Matricola/Email (2)".

## Fase 2 – completamento del lavoro (0.5 punti)

- **Completare il primo form** qualora non già completato in laboratorio **con la stessa modalità utilizzata in laboratorio** (se svolto in modo individuale continuare in modo individuale, se svolto a coppie continuare a coppie).
- Completare il secondo form adibito allo svolgimento a casa inserendo anche i **commenti finali**.
- La **consegna del secondo form** può essere effettuata esclusivamente in modo **individuale**.
- **Entro la scadenza prefissata** inviare il secondo form compilato.
    - Scadenza: **23 aprile 2025 ore 23:59 CET**
- La consegna è ritenuta valida **se e solo se**:

- Il form è stato inviato;
- Contiene le domande, risposte e i relativi prompt, soluzioni, e commenti degli esercizi **da 4 a 7** (inclusi);
- La data di consegna non supera la scadenza prefissata.

## Pattern

Le query proposte prevedono livelli di difficoltà diversa, secondo i seguenti pattern:

0. **[Laboratorio] Proiezione**: la (vostra) soluzione attesa alla domanda proposta deve contenere la clausola WHERE, più eventualmente ORDER BY.

1. **[Laboratorio] Join**: la (vostra) soluzione attesa alla domanda proposta deve prevedere l'operazione di JOIN, più eventualmente altre condizioni di selezione.

2. **[Laboratorio] Raggruppamento con condizione e calcolo di funzione aggregata**: la (vostra) soluzione attesa alla domanda proposta deve contenere le clausole GROUP BY e HAVING e deve prevedere il calcolo di una funzione aggregata.

3. **[Laboratorio] Soluzioni multiple Join/IN/EXISTS/INTERSECT**: la domanda proposta deve poter essere risolta utilizzando strategie diverse su cui interrogare separatamente l'LLM:
   - Join
   - IN
   - EXISTS
   - INTERSECT

4. **[Casa] Costruttore di tupla**: la (vostra) soluzione attesa alla domanda proposta deve prevedere l'utilizzo del costruttore di tupla.

5. **[Casa] Divisione**: la (vostra) soluzione attesa alla domanda proposta deve prevedere un'operazione di divisione.

6. **[Casa] Soluzioni multiple NOT IN/NOT EXISTS/EXCEPT**: la domanda proposta deve poter essere risolta utilizzando strategie diverse su cui interrogare separatamente l'LLM:
   - NOT IN
   - NOT EXISTS
   - EXCEPT

7. **[Casa] Soluzioni multiple TABLE FUNCTION/CTE/Correlazione**: la domanda proposta deve poter essere risolta utilizzando strategie diverse su cui interrogare separatamente l'LLM:
   - TABLE FUNCTION
   - CTE
   - Correlazione

## Piattaforma

Per svolgere l'homework, è necessario compilare i seguenti due form, accessibili esclusivamente attraverso l'indirizzo email istituzionale fornito dal Politecnico (sMATRICOLA@studenti.polito.it):

- Primo form (da svolgere in laboratorio): https://forms.office.com/e/RnuZ0vcB9i
  Scadenza: termine slot laboratorio
- Secondo form (da svolgere a casa): https://forms.office.com/e/B2uDdyhwq1
  Scadenza: **23 aprile 2025 ore 23:59 CET**

Per poter inviare e sottomettere correttamente le risposte, occorre arrivare all'ultima pagina e premere il pulsante **INVIA**.
Sottomissioni precedenti all'orario di inizio del proprio slot di laboratorio o dopo la scadenza non saranno considerate.

## Note

- Assicurarsi di inviare e salvare correttamente il form al termine dell'esercitazione, altrimenti non sarà possibile completarlo.
- Sottomissioni parziali saranno valutate con eventuali punteggi parziali.
- Sottomissioni largamente incomplete non riceveranno punti.

## Esempio di prompt

### Pattern 0 – Proiezione

"Sei un assistente AI per la risoluzione di query in linguaggio SQL partendo da una domanda testuale.
Ti verranno forniti in input lo schema logico delle tabelle da utilizzare e la domanda testuale.
Tu dovrai restituire come output solamente il codice SQL.

Tabelle:

LIBRO(id_libro, id_autore, titolo, anno_pubblicazione, genere, numero_copie) con chiave primaria id_libro
AUTORE(id_autore, nome, cognome, nazionalità, data_nascita, data_morte*) con chiave primaria id_autore
LETTORE(id_lettore, nome, cognome, email, telefono, indirizzo, data_iscrizione) con chiave primaria id_lettore
PRESTITO_LIBRO(id_libro, id_lettore, data_prestito, data_restituzione*, note*) con chiave primaria (id_libro, id_lettore, data_prestito)
SALA(id_sala, id_dipendente_responsabile, nome, capacità, piano, tipologia) con chiave primaria id_sala
DIPENDENTE(id_dipendente, nome, cognome, ruolo, data_assunzione, email, telefono) con chiave primaria id_dipendente
PRENOTAZIONE_SALA(id_lettore, data, ora_inizio, ora_fine, id_sala, note*) con chiave primaria (id_lettore, data, ora_inizio)

L'asterisco indica i campi opzionali.

Domanda:

Mostra il titolo e l'anno di pubblicazione dei libri del genere "fantasy" con almeno 3 copie disponibili ordinati a partire dal più recente.

Codice SQL:"