# Query da tradurre – Mario Varlese

## Query 1 – Join Mysql chatgpt avanzato

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Traduzione query per ricerca ordini prodotti in db relazionale |
| ***Descrizione del Problema*** | Vorrei che mi traducessi la seguente query sql in python. In particolare vorrei che mi traducessi la tenendo in conto che l’applicativo che la utilizzerà utilizzerà il framework django. |
| ***Query da tradurre*** | **SELECT**  po**.**ID **AS** ProdottoOrdinatoID**,**  po**.**NumeroOrdine**,**  po**.**NomeModello**,**  po**.**Quantita**,**  po**.**PrezzoUnitario **AS** PrezzoUnitarioProdotto**,**  po**.**Valuta **AS** ValutaProdotto**,**  po**.**DataAggiunta**,**  po**.**DataConsegnaPrevista**,**  po**.**StatoProdotto**,**  po**.**NumeroRMA**,**  po**.**Note **AS** NoteProdotto**,**  ae**.**NomeModello **AS** NomeModelloApparecchiatura**,**  ae**.**NumeroSerie**,**  ae**.**DataProduzione**,**  ae**.**Reparto**,**  ae**.**Fornitore **AS** FornitoreApparecchiatura**,**  ae**.**DataAcquisto**,**  ae**.**Cliente**,**  ae**.**RepartoAssistenza**,**  ae**.**DataUltimaManutenzione**,**  ae**.**GaranziaScadenza**,**  ae**.**DescrizioneProblema**,**  ae**.**ComponenteDifettoso**,**  ae**.**TecnicoAssistenza**,**  ae**.**Note **AS** NoteApparecchiatura**,**  oc**.**NumeroOrdine **AS** NumeroOrdineCliente**,**  oc**.**DataOrdine**,**  oc**.**Cliente **AS** ClienteOrdine**,**  oc**.**IndirizzoSpedizione**,**  oc**.**CittaSpedizione**,**  oc**.**CAPSpedizione**,**  oc**.**ProvinciaSpedizione**,**  oc**.**NazioneSpedizione**,**  oc**.**MetodoPagamento**,**  oc**.**StatoPagamento**,**  oc**.**DataPagamento**,**  oc**.**MetodoSpedizione**,**  oc**.**DataSpedizione**,**  oc**.**StatoSpedizione**,**  oc**.**DataConsegna**,**  oc**.**TotaleOrdine**,**  oc**.**Valuta **AS** ValutaOrdine**,**  oc**.**NumeroFattura**,**  oc**.**CodicePromozionale**,**  oc**.**Note **AS** NoteOrdine  **FROM** ProdottiOrdinati po  **JOIN** ApparecchiatureElettroniche ae **ON** po**.**NomeModello **=** ae**.**NomeModello  **JOIN** OrdiniClienti oc **ON** po**.**NumeroOrdine **=** oc**.**NumeroOrdine  **WHERE** po**.**DataAggiunta **BETWEEN** '2022-01-01' **AND** '2023-06-30'  **AND** ae**.**DataProduzione **<=** '2023-06-30'  **AND** oc**.**DataOrdine **>=** '2022-01-01'**;** |
| ***Dettagli Aggiuntivi*** | La query utilizza la sintassi di MySql ed effettua una join sulle tabelle ProdottiOrdinati, ApparecchiatureElettroniche e OrdiniClienti. Assicurati di mantenere la stessa nomenclatura per gli oggetti definiti nel modello. Vorrei che mi traducessi la query utilizzando il framework django. |
| ***Requisiti Tecnici*** |  |
| ***Livello di Sicurezza*** | Vorrei inoltre che mi motivassi se l’utilizzo di Djando ORM mi fornisca un buon livello di sicurezza contro attacchi di Sql-Injection. |
| ***Extra*** | Vorrei che l’output generato sia ritornato in formato JSON e che contenga tutti i campi contenuti nella select, ma aventi la nomenclatura snake case |
| ***Struttura del codice*** |  |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 31,18 secondi |
| ***Numero Modifiche Senza Ausilio dello Strumento*** |  |
| ***Numero Modifiche Con Ausilio dello Strumento*** |  |
| ***Numero Errori*** |  |
| ***Tempo medio esecuzione script*** |  |

## Query 2 – select Postgresql copilot intermedio

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Conversione query ricerca Apparecchiature Elettroniche in python |
| ***Descrizione del Problema*** |  |
| ***Query da tradurre*** | **SELECT** **\***  **FROM** "apparecchiatureelettroniche"  **WHERE** **CAST(**SPLIT\_PART**(**"risoluzioneschermo"**,** 'x'**,** 2**)** **AS** **INT)** **>** 1080  **ORDER** **BY** "dataproduzione" **DESC;** |
| ***Dettagli Aggiuntivi*** | Vorrei mi traducessi la query indicata utilizzando il linguaggio python, tieni presente che il backend di riferimento utilizzera un database PostgressSql |
| ***Requisiti Tecnici*** | Utilizza la libreria psycopg2 |
| ***Livello di Sicurezza*** | Evidenziami i parametri che potrebbero essere sucettibili di Sql-Injection e proponimi una versione sicura della query che utilizzi i prepared statements. |
| ***Extra*** |  |
| ***Struttura del codice*** | #import delle librerie #definizione di variabili #funzione per connettersi al database  #funzione per eseguire la query  #funzione per stampare i risultati  #main |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 1 minuto 29 secondi |
| ***Numero Modifiche Senza Ausilio dello Strumento*** | 2 |
| ***Numero Modifiche Con Ausilio dello Strumento*** | 1 |
| ***Numero Errori*** | 3 |
| ***Tempo medio esecuzione script*** | 98 ms ± 17.8 ms |

## Query 3 – select oracle chatgpt intermedio

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Conversione SQL (dialetto Oracle) in Python |
| ***Descrizione del Problema*** | Mi convertiresti la seguente query in python tenendo presente che il backend sarà un database Oracle xe21c |
| ***Query da tradurre*** | **SELECT** **\***  **FROM** "APPARECCHIATUREELETTRONICHE"  **WHERE** **TO\_NUMBER(SUBSTR(**"RISOLUZIONESCHERMO"**,** **INSTR(**"RISOLUZIONESCHERMO"**,** 'x'**)**+1**))** **>** 1080  **ORDER** **BY** "DATAPRODUZIONE" **DESC;** |
| ***Dettagli Aggiuntivi*** | Il database di riferimento è Oracle. Inoltre vorrei che utilizzassi il connettore adeguato, tenendo conto che il db di riferimento è Oracle xe21c. i parametri di connessione al database sono i seguenti: hostname=localhost, user=SYSTEM, password=oirad1993 |
| ***Requisiti Tecnici*** | Assicurati di gestire adeguatamente le eccezioni (forniscimi la struttura try catch ed utilizza la clausola finally per gestire la deallocazione delle risorse)  Inoltre vorrei gestissi le seguenti casistiche:   * Casistica nessun risultato trovato: La funzione deve ritornare un json avente la seguente struttura {‘status\_code’: 404,’subcode’:5} * Casistica risultato trovato: La funzione deve ritornare i dati utilizzando la seguente struttura json {‘data’:<{returned data}>,’request\_timestamp’:<current\_datetime>}. Dove returned\_data rappresenta l’output della query mentre current\_timestamp rappresenta l’istante di tempo in cui è stata ricevuta la richiesta (usa la libreria datetime per calcolarlo) |
| ***Livello di Sicurezza*** | - |
| ***Extra*** | Utilizza i commenti per fornirmi dettagliatamente informazioni sulle operazioni che vengono fatte |
| ***Struttura del codice*** |  |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 20 secondi |
| ***Numero Modifiche Senza Ausilio dello Strumento*** | 0 |
| ***Numero Modifiche Con Ausilio dello Strumento*** | 0 |
| ***Numero Errori*** | 0 |
| ***Tempo medio esecuzione script*** | 159 ms ± 48.2 ms |

Nota: non è stato possibile salvare l’output in json perché i dati non sono serializzabili.

## Query 4 – vista oracle google bard Base

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Creazione Vista Oracle – traduzione python |
| ***Descrizione del Problema*** | Potresti crearmi la seuente vista in Oracle utilizzando il linguaggio python |
| ***Query da tradurre*** | **CREATE** **VIEW** VistaManipolazioneDati **AS**  **SELECT**  AE**.**ID **AS** APPARECCHIATURAID**,**  'Modello: ' **||** AE**.**NOMEMODELLO **||** ' (ID: ' **||** AE**.**ID **||** ')' **AS** DESCRIZIONEMODELLO**,**  AE**.**NUMEROSERIE **AS** NUMEROSERIE**,**  **TO\_CHAR(**AE**.**DATAPRODUZIONE**,** 'DD/MM/YYYY'**)** **AS** DATAPRODUZIONEFORMATTATA**,**  **UPPER(**AE**.**REPARTO**)** **AS** REPARTOMAIUSCOLO**,**  'Fornitore: ' **||** AE**.**FORNITORE **||** ', Acquisto: ' **||** **TO\_CHAR(**AE**.**DATAACQUISTO**,** 'DD/MM/YYYY'**)** **AS** DETTAGLIACQUISTO**,**  **NVL(**AE**.**CLIENTE**,** 'NESSUN CLIENTE'**)** **AS** CLIENTEODEFAULT**,**  **TRUNC(**AE**.**GARANZIASCADENZA**)** **-** **TRUNC(SYSDATE)** **AS** GIORNIRIMANENTIGARANZIA**,**  **UPPER(**AE**.**STATOPRODUZIONE**)** **AS** STATOPRODUZIONEMAIUSCOLO**,**  'Dimensioni: ' **||** AE**.**DIMENSIONI **||** ', Peso: ' **||** AE**.**PESO **||** ' kg' **AS** DESCRIZIONEDIMENSIONIPESO  **FROM** APPARECCHIATUREELETTRONICHE AE**;** |
| ***Dettagli Aggiuntivi*** | Utilizza la libreria cx\_Oracle |
| ***Requisiti Tecnici*** | - |
| ***Livello di Sicurezza*** | - |
| ***Extra*** |  |
| ***Struttura del codice*** |  |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 10.85 sec |
| ***Numero Modifiche Senza Ausilio dello Strumento*** | 0 |
| ***Numero Modifiche Con Ausilio dello Strumento*** | 0 |
| ***Numero Errori*** | 0 |
| ***Tempo medio esecuzione script*** | - |