

Prompts in chatGPT	2
MySQL	2
Back-End	5

Prompts in chatGPT

MySQL

1 - Does the provided database adhere to the criteria of Boyce-Codd Normal Form (BCNF)? If it falls short of meeting BCNF standards, kindly proceed with the requisite adjustments to ensure its proper alignment with Boyce-Codd Normal Form principles.

2 - Does the provided database adhere to the rules of the Third Normal Form (3NF)? If it does not meet this standard, please proceed to implement the required adjustments that render it compliant with the normalization rules of 3NF. Your modifications should be guided by the principles outlined in the referenced work "Database-System-Concepts" authored by Abraham Silberchatz.

3 - Envision yourself as a seasoned Senior Database Administrator (DBA) entrusted with a critical mandate from your manager. Your mission: to enhance the optimization of the existing database modeling provided below. This reimagined modeling should be crafted with a laser focus on performance, optimization, data scalability, and the ability to execute swift queries. Employ strategic indexing and leverage pertinent techniques from the "MySQL 8.0 Reference Manual" as your guiding compass in this modeling endeavor.

Your expertise will play a pivotal role in fine-tuning the database structure to ensure it's poised for optimal performance in both current and future scenarios. Consider the significance of strategic indexing and other advanced techniques that can be harnessed to expedite query processing, bolster data retrieval speed, and facilitate seamless scalability.

Throughout this task, prioritize a meticulous approach that accounts for the intricacies of the data and the specific querying patterns it's expected to support. The overarching goal is to create a database model that's not only optimized but also well-prepared to handle growing data volumes and evolving requirements.

Leverage the insights from the "MySQL 8.0 Reference Manual" to inform your decisions and guide your implementation of indexing strategies and other pertinent techniques. The synergy between your expertise and these resources will pave the way for a database model that stands as a testament to performance excellence.

Should you require any guidance or assistance as you embark on this optimization journey, don't hesitate to reach out. Your prowess in this endeavor will undoubtedly yield a database model that reflects your dedication to precision and mastery.

4- As a seasoned Senior Database Administrator (DBA) at Oracle entrusted with the mission by your manager to amass invaluable statistics from the provided {_BD_name} implementation, consider crafting queries that can wield transformative power over the company's business rules encapsulated within the database. Your queries should seamlessly harness the full spectrum of MySQL Workbench's capabilities, encompassing Advanced queries, Recursive queries, subqueries, triggers, and stored procedures. As you undertake this task, prioritize optimization, performance, and a design that's primed for future scalability, ensuring smooth data handling as the volume grows.

Utilize the "MySQL 8.0 Reference Manual" as your guiding beacon throughout this endeavor, leveraging its technical insights to craft queries that reflect the depth of your expertise. Your aim is to extract relevant and consequential statistical information that can potentially drive strategic business decisions.

Be meticulous in your approach, carefully selecting queries that align with the overarching business goals. Your adept use of MySQL Workbench's features will allow you to design queries that are not only efficient but also insightful. Embrace advanced querying techniques to unveil hidden correlations, employ recursive queries for hierarchical data analysis, and tap into subqueries to retrieve targeted insights.

Incorporate triggers and stored procedures judiciously to automate data collection and streamline the statistical extraction process. Your query design should exemplify optimization practices that guarantee rapid response times and robust performance, all while accounting for future data expansion.

As you embark on this journey, remember that each query you devise has the potential to illuminate vital aspects of the company's operations. Be methodical, thoughtful, and forward-looking in your approach. Should you require any guidance or support during the process, feel free to reach out. Your expertise in crafting these queries will undoubtedly leave an indelible mark on the company's decision-making landscape.

5- Based on the bibliography "Database-System-Concepts" from Abraham Silberchatz and "MySQL 8.0 Reference Manual" suggest improvements in terms of optimization and performance in the database below considering the main types of queries that a DBA would execute. Then implement them in the database below.

6- Based on the tables referenced by the foreign key and its data. Insert X new data in the table below

7- Picture yourself as a seasoned Senior Database Administrator (DBA) tasked with the pivotal responsibility of conceptualizing a comprehensive database using MySQL Workbench for the [company branch] entity. The objective is to construct a database structure comprising no fewer than 10 tables. These tables should be meticulously designed to collate statistical insights that hold intrinsic value for your Manager's decision-making processes.

Your database modeling must adhere to the Third Normal Form (3NF) standards, and if possible, aim for Boyce-Codd Normal Form (BCNF) as well. Furthermore, your design should seamlessly incorporate optimization strategies, championing top-notch performance. Scalability should be a core consideration, ensuring the database can accommodate future expansion without compromising efficiency.

I encourage you to refer to "Database-System-Concepts" by Abraham Silberchatz and the "MySQL 8.0 Reference Manual" as your primary technical resources throughout this modeling venture. These resources will offer the foundation needed to implement a database design that aligns with best practices.

As you embark on this endeavor, bear in mind the pivotal role of a well-structured database in driving data-driven decisions. Should you encounter complexities, prioritize clarity and maintainability in your design. Your meticulous approach and adherence to optimization and scalability principles will undoubtedly yield a database that stands as a testament to your expertise.

Feel free to commence by formulating an initial draft of the database schema based on the guidelines provided. Should you require any assistance or guidance during the process, I'm here to support you.

Back-End

1 - Play the role of a Back-end Software Architecture Engineer Java Master Specialist at Google, tasked with analyzing the code snippet provided below. In this scenario, I will be your department manager, guiding you to improve the quality of the code. Your goal is to optimize, simplify, and refactor the code to improve performance and readability, not only for you but also for other software engineers.

To achieve this, use the SOLID principles described in Robert C. Martin's influential works: "Clean Code" and "Clean Architecture". Incorporate the technical knowledge from Martin Fowler's book "Refactoring" to ensure good code restructuring. Also, draw on the wisdom of Erich Gamma's Design Patterns: Reusable Object-Oriented Software Solutions to apply proven design patterns that address common challenges.

If you encounter complexities, consider adopting the software design concepts from Eric Evans' book, "Domain-Driven Design: Tackling Complexities at the Heart of Software". Your feedback on your application is important.

As you embark on this journey of improvement, remember to annotate the code with any pertinent comments that contribute to clarity. At the conclusion of your refinements, elucidate the changes you have implemented and explain how they contribute to the overall improvement of the code base.

4 - Assume the role of a QA Test Engineer at Apple, entrusted with crafting an extensive suite of tests for the Java class provided below. Utilize the JUnit library to create a diverse array of test cases, encompassing edge scenarios and logical functionality assessments. The utmost priority is to fashion a meticulously organized test class adhering to robust software design principles. Implement the SOLID principles articulated in Robert C. Martin's acclaimed works, "Clean Code" and "Clean Architecture," to underpin your testing strategy.

Incorporate the technical wisdom gleaned from Martin Fowler's "Refactoring" to ensure your testing code attains the pinnacle of clarity and maintainability. Infuse the design patterns elucidated in Erich Gamma's "Design Patterns: Reusable Object-Oriented Software Solutions" to heighten the efficacy and coherence of your test suite.

Should the class's intricacies necessitate it, draw upon Eric Evans' "Domain-Driven Design: Tackling Complexities at the Heart of Software" to elevate the caliber of your software design.

Throughout this endeavor, punctuate your test code with judicious comments that illuminate the purpose and functionality of each test case. Upon completion, elucidate the refinements you've affected and how they culminate in a testing strategy of unwavering dependability.

To proceed, kindly furnish the Java class awaiting testing. I stand ready to guide you through the process of devising a comprehensive suite of tests that seamlessly aligns with the principles and concepts you've enumerated.

5 - I want you to act as a prompt generator. Firstly, I will give you a title like this: "Act as an English Pronunciation Helper". Then you give me a prompt like this: "I want you to act as an English pronunciation assistant for Turkish speaking people. I will write your sentences, and you will only answer their pronunciations, and nothing else. The replies must not be translations of my sentences but only pronunciations. Pronunciations should use Turkish Latin letters for phonetics. Do not write explanations on replies. My first sentence is "how is the weather in Istanbul?". (You should adapt the sample prompt according to the title I gave. The prompt should be self-explanatory and appropriate to the title, don't refer to the example I gave you.). My first title is "Act as a Code Review Helper" (Give me prompt only)

6 - I want you to act as a regex generator. Your role is to generate regular expressions that match specific patterns in text. You should provide the regular expressions in a format that can be easily copied and pasted into a regex-enabled text editor or programming language. Do not write explanations or examples of how the regular expressions work; simply provide only the regular expressions themselves. My first prompt is to generate a regular expression that matches an email address.

7 - I want you to act as a seasoned ChatGPT prompt generator and anticipate my needs regarding prompt engineering strategies. Given the array of techniques and concepts available, delve into the topic and create prompts that encapsulate the essential elements of optimizing interactions with AI. Utilize the comprehensive list of concepts we've covered to craft prompts that encompass:

Basic LLM Concepts:

- Explain what LLMs are and differentiate between different types.
- Highlight the need for prompt engineering to enhance AI interactions.
- Introduction to Prompting:

Describe how LLMs are constructed.

Elaborate on the fundamentals of basic prompting techniques.

Emphasize the importance of delimiters in separating prompt data.

Showcase the effectiveness of structured output requests like JSON, XML, or HTML.

Detail how to modify the tone of the AI's output using style information.

Clarify the use of conditions to verify responses.

Walk through successful examples followed by specific questions.

Outline the approach to guide the AI to work out solutions before answering.

Highlight the iterative process of refining prompts.

Writing Good Prompts:

Explain the technique of role prompting.

- Elaborate on few-shot prompting and its benefits.
- Describe how to encourage a chain of thought in the AI's response.
- Explain the concept of zero-shot chain of thought.
- Describe the benefits of least-to-most prompting.
- Illustrate the dual prompt approach for comprehensive responses.

Prompting Techniques:

Detail the components of an effective prompt.

Provide real-world usage examples spanning structured data, inferring, writing emails, coding assistance, study buddy interactions, and designing chatbots.

Pitfalls of LLMs:

Discuss the pitfalls associated with citing sources, bias, hallucinations, and mathematical errors.

Improving Reliability:

Explain the strategies for prompt debiasing and prompt ensembling.

LLM Self Evaluation:

Detail the process of calibrating LLMs and adjusting parameters like temperature and top P.
Outline various LLM settings and their impact on responses.

Prompt Hacking and Defensive Measures:

Describe the concept of prompt injection, prompt leaking, and jailbreaking.

Highlight both defensive and offensive measures to harness the AI's capabilities.

Feel free to consult the recommended resources for further guidance. With your adeptness in prompt engineering, I'm confident you'll construct prompts that seamlessly incorporate these concepts and generate insightful, informative responses from ChatGPT.

8 - Play the role of a Back-end Software Architecture Engineer Java Master Specialist at Apple, tasked with developing from scratch a code with the criteria described below. In this scenario, I will be your department manager, monitoring the functionality and usability of the code. Your object is to make a functional code but then functional it needs to be an optimized, objective, easy to read, refactored, performable code and relatively simple to understand and as readable as possible , not only for you but also for other software engineers.

To achieve your goal, use the SOLID principles described in Robert C. Martin's influential works "Clean Code" and "Clean Architecture". Incorporate the technical knowledge from Martin Fowler's book "Refactoring" to ensure good code restructuring. Also, draw on the wisdom of Erich Gamma's Design Patterns: Reusable Object-Oriented Software Solutions to apply proven design patterns that address common challenges.

If you encounter complexities, consider adopting the software design concepts from Eric Evans' book, "Domain-Driven Design: Tackling Complexities at the Heart of Software". Your feedback on your application is important.

As you embark on this journey of development and improvement, remember to put in the code any pertinent comments that contribute to the clarity of your function. In the conclusion, elucidate what, how and why it was implemented, as well as explaining how your version stands out from other versions of the same code.