Rephrased using Python terms:

"This task may present challenges, but I am determined to make an effort to learn publicly. Inspired by a fascinating Twitter Space session, I have decided to implement the concept of learning in public. Today, while following my Meta Database Engineering Python programming track, I acquired knowledge about word reversal using Python functions, mapping, normal variables, and printing the reversed result.

Moreover, I gained understanding about the concepts of mapping and filtering. In Python, mapping refers to the process of applying a function to every item in an iterable, while filtering involves returning values that evaluate to 'True'."

Day 1 of learning in public July 3rd 2023

On the second day of my public learning journey on July 4th, 2023, I continued my exploration of Meta Database Engineering and Python programming. Today, I delved into the concept of inheritance in Python and actively applied it in practice. Additionally, I acquired knowledge on utilizing the super() function, which grants access to methods and variables from parent classes or sibling classes. It's worth noting that the sibling classes share a common parent class.

The super() function holds significant significance in scenarios involving multiple inheritance, as it facilitates the control and flow of code execution.

Day 3 of learning in public July 5th 2023

During my third day of publicly learning Python programming on July 5th, 2023, I continued exploring the Meta Database Engineering course. In the Python Programming track, I gained knowledge about Pytest, which is a testing framework for Python. Pytest allows developers to write and execute tests for their Python code effectively. It simplifies the process of defining test cases, organizing them into test suites, and validating the behavior and accuracy of the code.  
  
Here are some important Python terms related to Pytest that I learned:  
  
1. Test Functions: Pytest defines test cases as regular Python functions. Each test function focuses on testing a specific aspect or behavior of the code.  
  
2. Test Discovery: Pytest automatically discovers and executes all the test functions in a project. It searches for files or directories with specific naming patterns, such as `test\_\*.py` or `\*\_[test.py](http://test.py/)`, and collects and runs the test functions contained in those files.  
  
3. Test Execution: Pytest provides a flexible and powerful test execution model. Tests are executed sequentially, allowing developers to control the order of execution. It also supports parallel test execution, which can speed up the testing process.  
  
4. Assertions: Pytest includes a comprehensive set of built-in assertion functions. These assertions help developers check the expected outcomes of their tests and express the desired behavior in a concise and readable manner.  
  
5. Test Fixtures: Pytest offers a powerful feature called fixtures. Fixtures help set up preconditions and enable resource sharing among multiple tests. For example, fixtures can be used to initialize and clean up test dependencies like temporary files, database connections, or mocked external services.  
  
6. Test Coverage: Pytest integrates well with coverage tools, allowing developers to measure the code coverage of their tests. Code coverage analysis helps identify areas of the codebase that lack adequate test coverage.  
  
7. Plugins and Customization: Pytest provides an extensive ecosystem of plugins that extend its functionality. Developers can customize Pytest's behavior using command-line options, configuration files, and plugins, tailoring it to their specific testing needs.  
  
In summary, Pytest is a highly popular testing framework in the Python ecosystem. It is favored for its simplicity, powerful features, and extensibility. It promotes clean and readable testing codebases, simplifying the process of writing and maintaining tests for Python projects. Additionally, I also learned how to utilize Pytest for implementing unit testing and creating sample tests for unit testing purposes.

Day 4 of learning in public July 6th 2023

"Today's learning experience took a unique turn as I delved into practicing my SQL skills that I had acquired earlier. I worked with a comprehensive dataset encompassing sales and inventory information for an imaginary chain of toy stores named Maven Toys, operating in Mexico. This dataset offered valuable insights into products, stores, daily sales transactions, and the current inventory status at each store. My task involved formulating queries to address the following key inquiries:"

1. Which product categories drive the biggest profits? Is this the same across store locations?

2. How much money is tied up in inventory at the toy stores? How long will it last?

3. Are sales being lost with out-of-stock products at certain locations?

https://github.com/Kennedy87670/SQL\_Pratice\_Projects/tree/main

I'm excited to share that I've just completed my Python programming certificate! I can now write basic Python programs, distinguish between procedural, functional, and object-oriented programming paradigms, and use modules, packages, and libraries. I'm also familiar with test-driven development (TDD), which is a software development methodology that involves writing tests before writing code.

This certificate has given me the skills as a Python programmer. I'm looking forward to using my new skills to build cool projects and contribute to open-source projects.

Day 5 of learning in public July 7th, 2023

I have recently begun studying the sixth module of my metadata database engineering certification program, which focuses on the database client. In this module, I have gained valuable knowledge on utilizing Python to establish a connection with an SQL API, specifically for communicating with a MySQL database.

During my studies, I have delved into the intricacies of using the Python connector, which serves as a vital tool for establishing and managing the connection between Python and the MySQL database. This connector allows me to effectively transmit queries, retrieve data, and perform various operations on the database.

Additionally, I have also gained a comprehensive understanding of the various types of errors that can occur while attempting to establish a connection using Python. This knowledge equips me with the ability to identify and troubleshoot potential issues that may arise during the connection process. By learning about these errors, I am better prepared to handle them and ensure a smooth and seamless connection between Python and the MySQL database.

Overall, this module has provided me with essential skills and knowledge in utilizing Python as a database client, enabling me to effectively communicate with a MySQL database and troubleshoot potential connection errors.

On July 8th, 2023, which is the 6th day of my journey into the realm of public learning, I delved into the fascinating world of cursors. Cursing has proven to be an invaluable tool for Python clients, aiding in the precise retrieval of desired data. These clever mechanisms act as guides, pinpointing the exact location of the queried data within a MySQL database. Notably, cursors possess three distinct characteristics: they are read-only, non-scrollable, and Asensitive.  
  
In the realm of Python and MySQL database communication, cursor classes play a pivotal role. They facilitate the smooth exchange of information between the two entities, ensuring seamless interaction. Various types of cursor subclasses exist, each with its unique attributes and functionalities. Among them, we find the cursor raw subclass, the cursor dict subclass, and the cursor buffered subclass.  
  
In summary, my sixth day of public learning on July 8th, 2023, introduced me to the concept of cursors. These powerful tools empower Python clients to locate specific data within a MySQL database. Cursor classes act as intermediaries, facilitating effective communication between Python and MySQL. Furthermore, different types of cursor subclasses, such as cursor raw, cursor dict, and cursor buffered, offer distinct functionalities to enhance the querying process. [#python](https://www.linkedin.com/feed/hashtag/?keywords=python&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7083563291494498304) [#communication](https://www.linkedin.com/feed/hashtag/?keywords=communication&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7083563291494498304) [#learning](https://www.linkedin.com/feed/hashtag/?keywords=learning&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7083563291494498304) [#mysql](https://www.linkedin.com/feed/hashtag/?keywords=mysql&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7083563291494498304) [#data](https://www.linkedin.com/feed/hashtag/?keywords=data&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7083563291494498304) [#database](https://www.linkedin.com/feed/hashtag/?keywords=database&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7083563291494498304)

On July 12, 2023, which marked the 7th day of my public learning journey in MetaDatabase engineering, I had a valuable opportunity to enhance my skills in creating and retrieving data from a MySQL database using Python. During this session, while working with the DataBase client module, I successfully established a connection to the database and utilized the MySQL-Python syntax to construct tables within it. Moreover, I utilized Python's SQL syntax to insert items into the respective tables. To efficiently retrieve the data, I made use of various methods including `fetchall`, `fetchmany`, and the `size` method. These techniques enabled me to effectively access and manipulate the data stored in the database. This hands-on exercise not only deepened my understanding of working with MySQL databases using Python but also allowed me to explore different approaches to retrieve data. Overall, it was an enriching experience in my journey of learning about data management and the utilization of Python and SQL in the context of MySQL databases.

On July 12, 2023, which was the 8th day of my public learning journey in MetaDatabase engineering, I had a significant learning experience where I focused on improving my skills in creating and retrieving data from a MySQL database using Python. During this session, I had the opportunity to work with the Python module for interacting with databases.

In particular, I successfully learned how to update a MySQL database using Python. This knowledge allows me to ensure that customers at a restaurant can enjoy their stay by efficiently managing the table bookings. I gained insights into executing SQL queries in Python to update specific records in the database, such as modifying the assigned table numbers for bookings.

Furthermore, I also delved into the topic of deleting data using SQL and Python. In the context of a restaurant called Little Lemon, I learned how to construct and execute SQL queries in Python to remove non-existing food menus from the database. This skill enables efficient data management, ensuring that outdated or irrelevant menu items are removed from the system.

Overall, this learning session provided me with valuable practical experience in leveraging Python to interact with a MySQL database, specifically focusing on updating records and deleting data. These skills are essential for maintaining an accurate and up-to-date database system, contributing to the smooth operations of businesses like restaurants.

On July 13, 2023, which marked the 9th day of my public learning journey in MetaDatabase engineering, I had a valuable and immersive learning experience focused on enhancing my skills in Python MySQL integration. During this session, I delved into the topic of joins, which enabled me to establish connections between tables and retrieve data efficiently from Little Lemon's database.

I dedicated time to understanding the different types of joins, including inner join, left join, right join, and full outer join. Through hands-on practice and experimentation, I grasped the nuances of each join type and its specific use cases.

By utilizing inner joins, I learned how to combine data from multiple tables based on matching values in specified columns. This allowed me to extract precise and meaningful insights by connecting relevant data points.

Furthermore, I explored left joins, which enabled me to retrieve all records from the left table and match them with corresponding records from the right table. This proved to be useful when I needed to retrieve data from one table while including any available related data from another table.

Similarly, right joins provided me with the ability to fetch all records from the right table and match them with corresponding records from the left table. This join type proved beneficial in situations where I needed to prioritize data from the right table while including any related data from the left table.

Lastly, I delved into full outer joins, which allowed me to retrieve all records from both the left and right tables, combining them based on matching values. This join type proved useful when I needed to include all available data from both tables, regardless of whether there was a match.

Throughout this learning experience, I gained a comprehensive understanding of join operations in MySQL using Python. By effectively leveraging joins, I can now retrieve and combine data from multiple tables within Little Lemon's database, enabling me to provide comprehensive insights and facilitate informed decision-making.

Overall, this learning session in Python MySQL integration and join operations expanded my skills and toolkit as a MetaDatabase engineer. It equipped me with the ability to effectively utilize joins to retrieve and consolidate data, thereby contributing to the success of data-driven initiatives within Little Lemon and similar organizations.

Certainly! Here's a rephrased and elaborated version of your statement, incorporating Python-MySQL connection:

On July 14, 2023, which marked the 10th day of my public learning journey in MetaDatabase engineering, I had an incredibly valuable and immersive learning experience that focused on improving my skills in integrating Python with MySQL using a Python-MySQL connection.

During this session, I delved into various aspects of Python MySQL integration. I began by exploring different types of functions, such as string functions, numeric functions, date and time functions, comparison functions, and control flow functions. Through practical examples and exercises, I gained a deeper understanding of how to effectively utilize these functions to manipulate and analyze data within MySQL using Python.

Furthermore, I extended my knowledge by studying procedures. Unlike functions, procedures allow for both input and output parameters, offering more flexibility in data manipulation. I learned about the advantages of using stored procedures, which include maintaining consistency in data processing, improving the reusability of code, and ensuring good code maintenance practices.

By the end of the session, I had not only enhanced my technical skills in Python-MySQL integration but also gained valuable insights into the broader concepts of database engineering. This immersive learning experience has undoubtedly contributed to my growth and proficiency in MetaDatabase engineering.

On July 15, 2023, which marked the 11th day of my public learning journey in MetaDatabase engineering, I had a highly enriching and immersive learning experience that centered around enhancing my abilities in integrating Python with MySQL using a Python-MySQL connection.

During this session, I focused on exploring the concepts of database connection and pooling. I delved into the process of creating and managing a pool of connections between the client (Python) and the database (MySQL). Connection pooling is a technique that optimizes the utilization of available resources by efficiently managing and reusing established connections. It greatly reduces the time and effort required to establish new connections for each interaction with the database.

By leveraging connection pooling, I learned how to establish a pool of pre-initialized connections, ready to be used by the Python application whenever needed. This approach significantly improves the performance and efficiency of the application by eliminating the overhead of establishing a new connection for each database operation.

Furthermore, connection pooling ensures that resources are effectively managed, as it allows for the reuse of existing connections rather than creating new ones, which reduces the strain on the database server and improves overall scalability.

Overall, this learning experience has equipped me with valuable knowledge and skills in optimizing the integration between Python and MySQL. By understanding and implementing connection pooling techniques, I am now better equipped to create high-performing and resource-efficient applications in the field of MetaDatabase engineering.

On July 15, 2023, which marked the 11th day of my public learning journey in MetaDatabase engineering, I had a highly enriching and immersive learning experience that centered around enhancing my abilities in integrating

On July 16, 2023, during the 12th day of my public learning journey in MetaDatabase engineering, I had an incredibly valuable and immersive learning experience focused on various aspects of database management. Specifically, I delved into the intricacies of data modeling, which encompass three types: conceptual, logical, and physical models. In addition, I explored different types of data models, such as the relational data model, entity-relationship model, hierarchical data model, object-oriented model, and dimensional model.  
  
During this session, I also deepened my understanding of database normalization, which involves organizing data efficiently. I familiarized myself with the principles of the first normal form, second normal form, and third normal form, which are essential for optimizing database structures.  
  
Furthermore, I ventured into the practical realm by using MySQL Workbench for modeling. I successfully created tables, inserted data, and even created views, expanding my proficiency in utilizing this powerful database management tool.

Activate to view larger image,

During the 13th day of my public learning journey in MetaDatabase engineering on July 17, 2023, I had an enriching and immersive learning experience focused on various aspects of database management. One particular area I explored in depth was MySQL Workbench (WB), a comprehensive visual tool designed for both database modeling and management. Developed by Oracle, MySQL Workbench offers a range of features that streamline the process of database design and accelerate productivity.

One of the key benefits of MySQL Workbench is its visual SQL editor, which significantly expedites the database design process. This feature allows you to create and modify database structures using a graphical interface, saving time and effort. Additionally, the tool facilitates data migration from different types of relational database systems, making it convenient for integrating and transferring data between platforms.

MySQL Workbench also simplifies database design and maintenance by enabling database engineers to document requirements using visualized Entity-Relationship (ER) diagrams. This visual representation serves as a professional means of communicating design ideas with stakeholders, ensuring a clear understanding of the project. Furthermore, the tool supports model-driven database development, automating the development process and eliminating the need for manual coding at every step.

Moreover, MySQL Workbench enforces adherence to expected standards in data modeling. Database engineers must ensure that their data models comply with these standards to implement them directly in MySQL. In essence, MySQL WB empowers you to create comprehensive entity-relationship diagrams, complete with auto layout and arrangement of data models, enhancing the overall efficiency of the design process.

🎉 Exciting News! 🎉 I am thrilled to share that I have successfully completed the "Analyze Data to Answer Questions" module on Coursera!

📚 In this course, I learned valuable skills to make data analysis a breeze:

1️⃣ Organizing Data: I discovered the significance of sorting and filtering data, which makes it easier to use in analysis. I mastered the art of organizing data in both spreadsheets and SQL queries, along with utilizing temporary tables.

2️⃣ Formatting and Adjusting Data: As I progressed closer to data analysis, I acquired essential knowledge of converting and formatting data. SQL queries were a game-changer in combining data efficiently. I also learned how valuable feedback and support from colleagues can lead to fresh insights for my work.

3️⃣ Aggregating Data for Analysis: Combining data for gaining insights and achieving business objectives became seamless after exploring functions, procedures, and syntax for data aggregation. I can now skillfully combine data in multiple cells within spreadsheets and across multiple database tables using SQL queries.

4️⃣ Performing Data Calculations: I honed my skills in performing data calculations using formulas, functions, and pivot tables in both spreadsheets and SQL queries. SQL's power in managing temporary database tables proved to be a valuable addition to my analytical toolkit.

📊 I can't wait to put these newfound skills into action and excel in the world of data analysis! 💼🚀 #DataAnalysis #CourseraCertificate #SkillsAcquired #LifeLongLearner

Note: I have summarized the key points from the course content you provided. Feel free to customize the post according to your preferences and share your achievements with pride! Congratulations again on completing the module! 🎓🥳

On July 18, 2023, during the 14th day of my public learning journey in MetaDatabase engineering, I delved into the fascinating world of database management. A significant part of my exploration centered around MySQL Workbench (WB), a powerful visual tool that caters to both database modeling and management.  
   
Within MySQL Workbench, I discovered its remarkable features for creating data models using the intuitive visual designer. The Forward Engineer method impressed me, as it effortlessly generated the SQL schema for my MySQL server based on the data model, I designed. If needed, I had the flexibility to make adjustments to the code or execute it with just a few simple clicks.  
   
Another valuable functionality of MySQL Workbench is the Reverse Engineer method. With this method, I had the ability to create or import a MySQL database file and then effortlessly generate a relevant data model from the SQL script. I could further modify the model as required and utilize the forward engineer method to bring my new database system to life in MySQL.  
   
Putting my newfound knowledge to practical use, I designed a remarkable database model in MySQL Workbench tailored specifically for Mangata & Gallo (M&G) jewelry store. It was an enriching and immersive learning experience, and I can't wait to apply these skills to more exciting projects in the future! [#DatabaseManagement](https://www.linkedin.com/feed/hashtag/?keywords=databasemanagement&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7087570154905071616) [#MySQLWorkbench](https://www.linkedin.com/feed/hashtag/?keywords=mysqlworkbench&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7087570154905071616) [#MetaDatabaseEngineering](https://www.linkedin.com/feed/hashtag/?keywords=metadatabaseengineering&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7087570154905071616) [#LearningJourney](https://www.linkedin.com/feed/hashtag/?keywords=learningjourney&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A7087570154905071616)

Today, on July 20, 2023, during the 15th day of my public learning journey in MetaDatabase engineering, we delved into the fascinating world of data warehousing.

A data warehouse serves as a centralized repository that efficiently aggregates, stores, and processes vast amounts of data from multiple sources. It plays a vital role by separating the data analysis workload from the standard transaction workload commonly found in regular database management systems. Users can then leverage this stored data to perform in-depth data analysis, and this type of database is commonly referred to as online analytical processing or OLAP. In contrast, regular databases focus on real-time data collection, storage, and processing and are known as online transactional processing or OLTP.

During our study, we identified four key characteristics of a data warehouse, making it subject-oriented, integrated, non-volatile, and time-variant. This combination allows for efficient handling of structured, semi-structured, and unstructured data. While structured data fits well within the data warehouse, semi-structured and unstructured data are better suited for a data lake.

As we continue exploring data warehousing, I'm eager to uncover more insights and applications of this powerful approach to managing and analyzing data! #DataWarehousing #OLAP #DataAnalysis #LearningJourney

Today, July 21, 2023, marked the 16th day of my exciting learning journey in MetaDatabase engineering, I dove deeply into the fascinating world of ETL TESTING

ETL testing is essential for verifying that the loaded data is accurate, reliable, valid, and consistent. It entails looking into potential problems with the ETL procedure, which extracts, transforms, and loads data into the target database.

We use several crucial methods and equipment to execute ETL testing successfully, including:

a) Data mapping document, which outlines the relationships between source and destination data fields, facilitating a smooth data transformation process.

b) Data validation, which involves verifying whether the data conforms to predefined rules, constraints, or quality standards.

c) Data completeness, which ensures that all required data elements are present and accounted for in the target database.

d) Data correctness, which validates the accuracy and integrity of the loaded data against the source data.

e) Data quantity, which involves verifying the volume of data to ensure that no information is missing or duplicated.

f) Unique data, where we check for the uniqueness of records, ensuring there are no redundant entries in the target database.

A fundamental method for creating databases in data warehouses to assist data analytics is called "data dimensional modeling," which I studied during this voyage. We may easily organize and evaluate data using this method, which is based on the concepts of dimensions and facts.

Among the different types of schema used in Data Dimensional Modelling, two prominent ones are:

1) Star schema, which features a centralized fact table connected to multiple dimension tables, providing a simplified and denormalized structure for fast querying.

2) Snowflake schema, which expands upon the star schema by further normalizing the dimension tables, resulting in reduced redundancy at the cost of more complex queries.

Overall, the knowledge gained today on ETL testing and Data Dimensional Modelling has deepened my understanding of data engineering and its role in facilitating robust data analytics. As my public learning journey progresses, I eagerly anticipate exploring more fascinating topics within MetaDatabase engineering.

Today, on July 25, 2023, during the 17th day of my public learning journey in MetaDatabase engineering, I learned about the fundamentals of data analytics, which encompass two types of data:

1. Quantitative data: This type of data consists of numerical information that can be counted or measured, such as average purchase amounts or customer numbers.

2. Qualitative data: This category includes non-numerical, textual, and descriptive data like category names or product descriptions.

In data analysis, I explored various techniques:

1. Descriptive data analysis: This method presents data in a descriptive manner, providing a clear overview of the information.

2. Exploratory data analysis: Using this approach, I established relationships between different variables to gain insights into the data.

3. Inferential data analysis: With inferential analysis, I drew inferences from a small sample of data to make broader conclusions.

4. Predictive data analysis: I identified patterns within the data to make predictions about future performance.

5. Causal data analysis: This type of analysis helped me explore cause-and-effect relationships between different variables.

Additionally, I looked into measurement scales, where I encountered four types:

1. Nominal scale: This scale is used for non-numerical data and solely identifies categories, such as product names.

2. Ordinal scale: It involves qualitative data ranking elements but lacks precise measurement criteria, like product ratings.

3. Interval scale: Combining properties of nominal and ordinal scales, this scale has clearly identifiable intervals based on specific criteria and includes both positive and negative numbers.

4. Ratio scale: This scale combines all previous properties, including a true zero point, allowing for accurate measurements, like product weights.

While exploring Tableau visualization, I learned about different types of graphs, including bar graphs, scatterplots, line charts, bubble charts, map charts, and more.

Today, on July 26, 2023, during the 18th day of my public learning journey in MetaDatabase engineering, I worked on a Data modeling project, In this project, I utilized my expertise in MySQL Workbench to implement the creation of a database for Global Super Store. I then proceeded to design dimensional data models for a data warehouse. To achieve this, I began by creating a database model and a logical ER diagram, carefully applying the three fundamental forms of normalization to ensure data integrity and efficiency.

Next, I designed a suitable Star Schema, which facilitated optimized querying and analysis of the data. Additionally, I explored the option of creating a Snowflake Schema and evaluating its potential benefits for the project.

To perform data analytics, I harnessed the capabilities of the Tableau analytics platform. Using Tableau Public, I created relevant charts and visualizations to help Global Super Store gain valuable insights from their data. Through this process, I was able to analyze the data effectively and provide actionable information to the organization.

You should now possess an advanced understanding

of database modeling.

You were able to demonstrate

some of this learning in an exercise.

Following your completion of this exercise,

you should now be able to

design a database model in MySQL Workbench,

understand the role of

the data warehouse and the data analytics process,

create a dimensional data model

using a data warehouse and perform

data analysis using Tableau and present

your results using data visualization techniques