Phase 1 & 2: Database Design & Implementation

STUDENT SUBMISSION TEMPLATE

Mental Health Court Database Project
Due: (Dec 4) • 150 Points

Team Name:	Team 8
Team Member 1:	Joey
Team Member 2:	Paul
Team Member 3:	Rowan
Team Member 4:	Max
Submission Date:	December 3

Submission Contents

Complete this template and submit along with your technical deliverables:

- ☐ Part A: Database Design & Architecture Documentation
- ☐ Part B: SQL Implementation Documentation
- ☐ Technical Deliverables Checklist
- ☐ Individual Contribution Statements
- ☐ Team Reflection & Lessons Learned
- ☐ Appendices (ER Diagrams, Code Samples, Test Results)

Part A: Database Design & Architecture (75 points)

A.1 Data Analysis & Requirements (25 points)

Dataset Overview Summary

Provide a comprehensive summary of your analysis of the CSV files:

The dataset tracks participants in the Mental Health Court program from 2019 to 2025 and shows many parts of their experiences in the justice system. It includes about 100 people, most of whom had serious mental health conditions like PTSD, depression, bipolar disorder, or anxiety. Many also had substance use problems. Most participants were involved with the court because of charges for drug possession, assault, theft, or probation violations. The data also shows social challenges when they entered the program, such as unstable housing, unemployment, and different levels of education.

The program is organized into structured phases with clear rules and risk monitoring. Participants move through these phases, and their outcomes include graduation, staying active in the program, leaving voluntarily, or, in some cases, death. Attendance at therapy, passing drug tests, and showing up to court were used to track compliance. Risk assessments were used to predict the chance of reoffending, from medium to very high. Some participants did commit new offenses while in the program, showing how mental health, substance use, and criminal behavior are connected.

The data does have some problems. There are mismatched start dates, differences between the main roster and other files, and missing jail or testing records. These issues need to be fixed before the data can be fully trusted. Once cleaned, the dataset can help understand how well the program works, patterns of reoffending, and how mental health, social factors, and legal history affect outcomes.

Key Data Relationships Identified

Describe the major relationships you discovered between different data domains:

The analysis of the Mental Health Court dataset shows strong and interconnected relationships between participants' mental health, legal histories, and social circumstances. There is a clear connection between clinical diagnoses and offense types. Participants with PTSD, bipolar disorder, or psychotic disorders were more often charged with assault or threats, while those with severe substance use disorders were primarily involved in drug-related or DWI offenses. This suggests that criminal behavior frequently reflects untreated or unstable psychiatric conditions, especially when combined with addiction.

A participant's chance of successfully graduating from the program depends on a combination of risk factors, stability, and compliance. Lower initial risk scores, stable housing or employment at entry, and consistent adherence to program requirements, such as attending therapy sessions and passing drug tests, strongly predicted graduation. By contrast, termination or re-offending was linked to severe substance use, homelessness, unemployment, and repeated probation violations. These setbacks were most common early in the program or during transitions between treatment phases, highlighting periods of increased vulnerability.

Overall, the data reveals a cycle in which severe mental illness and substance use contribute to legal involvement and social instability, which then increase the likelihood of non-compliance and recidivism within the program. Success in the Mental Health Court requires not only clinical treatment but also support for housing, income, and social reintegration. The findings support an integrated approach in diversionary courts that addresses both clinical and social needs to break this cycle and improve long-term outcomes.

Data Quality Issues & Solutions

Document data quality problems found and how you addressed them:

The dataset contains a lot of useful information, but there are several major quality problems that have to be resolved before the data can support reliable analysis. Many of the issues come from differences in how each file was created, missing details, and timelines that do not line up. Some of the biggest challenges involve start and end dates that do not match, multiple versions of participants' names that make it hard to track the same person, and important gaps such as missing risk assessments and missing post-program outcomes. There are also notes showing that updates were often made only in the main roster and never carried over to the mental health, offense, or performance files. This leaves each participant's record incomplete when viewed over time.

Fixing the data requires a clear and organized cleaning process. The first step is creating a consistent identifier for each participant by resolving name variations and aliases. After that, all time-related information, including program entry dates and movement through phases, needs to be checked and aligned across every file. Missing jail records and missing psychosocial exit information should be filled in when possible. All coded fields, including diagnoses and housing status, need to use the same formatting and the same definitions provided in the codebooks.

At the moment, the dataset fits into a Tier 2 category, meaning a large portion of the entries still need review and manual correction. The top priority is establishing the participant roster as the single reliable source for participant information and making sure any updates flow automatically to every related file. Without this foundation, any conclusions about participant progress or program success would risk being inaccurate because of the underlying inconsistencies.

Business Rules Documentation (At least 5 rules)

List the key business rules governing MHC program operations:

1. Program Eligibility & Admission Rule

Description: Participants must have a qualifying, documented mental health diagnosis and be facing eligible criminal charges to enter the MHC program. The rule ensures the court serves its intended population by screening for both clinical need and legal appropriateness.

2. Phased Progression Requirement Rule
Description: All participants must advance sequentially through five structured phases to graduate. Each phase has defined durations, behavioral expectations, and treatment milestones that must be met before proceeding to the next level of reduced supervision.

3. Mandatory Compliance Monitoring Rule
Description: Continuous adherence to program requirements is mandatory. This includes regular, observed drug testing, attendance at all scheduled therapy and treatment sessions, and appearance at all MHC court hearings. Non-compliance is tracked, documented, and triggers a graduated sanctions protocol.

4. Risk-Based Case Management Rule
Description: A standardized risk assessment tool must be administered at program entry and at scheduled intervals. The resulting risk category (Low, Medium, High, Very High) directly determines the intensity of supervision, frequency of contact, and allocation of support services.

5. Graduation Qualification Rule
Description: Graduation is contingent upon the successful completion of all five program phases, a sustained period of compliance, demonstration of improved psychosocial stability, and resolution of all pending legal matters. It represents full program fulfillment.

6. Termination & Discharge Protocol Rule
Description: Clear, documented criteria govern removal from the program. Grounds for termination include commission of a new felony, serious or repeated non-compliance, or failure to engage in treatment. The process requires documentation and likely team consensus.

7. Comprehensive Data Documentation Rule
Description: Every significant participant interaction, status change, assessment, and outcome must be recorded in the designated case management and tracking systems. This ensures accountability, enables program evaluation, and supports coordinated care across the clinical and legal teams.

Business Requirements Summary

Requirement Category	Description	Database Impact	Priority
Participant Management	Track enrollment, demographics, and status through 5-phase program model	Create PARTICIPANTS table with phase tracking and status fields	High

Criminal History Tracking	Record charges, convictions, jail time ($271/day), and probation	Need CHARGES, CONVICTIONS, JAIL_RECORDS tables with cost calculations	High
Mental Health Management	Manage DSM-5 diagnoses, assessments, and treatment episodes	Requires DIAGNOSES, ASSESSMENTS, TREATMENT tables	High
Program Evaluation	Calculate metrics: graduation rates, conviction reductions, 42 reports	Build views and procedures for analytics and reporting	High
Data Security	HIPAA compliance, de-identification	User permissions, audit logs	High
System Integration	Court, provider, probation connections	Data exchange tables	Medium
Data Quality	Validation, integrity, 95% completion	Constraints and checks	Medium
Performance	Fast queries, handle 2,000+ records	Indexes and optimization	Medium

A.2 Entity-Relationship Design (25 points)

ER Model Design Rationale

Explain your approach to designing the ER model and key design decisions:

The ER model was built to focus on participants and how their experiences unfold over time. Instead of relying on flat CSV files with inconsistent structures, the team reshaped the information into a cleaner, more connected schema. This makes it possible to track relationships between events and see how one moment affects another. For example, a missed UA in Phase 2 can now be linked to a later jail stay, which can then be connected to changes in treatment engagement and the chances of completing the program. This structure gives the MHC a clearer view of which interventions work best, when they work, and for which participants they have the strongest impact.

Core Entities Documentation (At least 4 Entities)

Entity Name	**Primary Purpose**	**Key Attributes**	**Relationships**
Participants	Core entity storing demographic and identification information for all individuals enrolled in the Mental Health Court program.	Participant_id (PK) Participant_name Date_of_birth Gender Race Address Contact info Created date	One-to-Many with: • Criminal Charges • Mental Health Diagnoses • MHC Enrollments
Criminal Charges	Comprehensive tracking of all criminal	Charge id (PK) Participant id (FK)	Many-to-One with: • Participants

	charges for participants across three time periods: before MHC entry, during MHC participation, and after MHC completion.	Enrollment id (FK) Charge date Offense Offense category Class Disposition Conviction date Sentence length Probation years During mhc Years before mhc Years after mhc	• MHC Enrollments (for charges during MHC period)
Metal Health Diagnosis	Tracks mental health diagnoses for participants to support treatment planning and outcome analysis by diagnosis type.	Diagnosis id (PK) Participant id (FK) Diagnosis code Diagnosis name Diagnosis date Diagnosed by severity Active status Notes	Many-to-One with: • Participants
MHC Enrollments	Tracks individual enrollment periods in the MHC program, supporting multiple enrollments per participant for program re-entry cases.	Enrollment id (PK) Participant id (FK) Date started mhc Date ended mhc Length of mhc end status Admission type Fiscal year entry Current phase	Many-to-One with: • Participants

Relationships Documentation

Relationship	Entity 1	Entity 2	Cardinality	Business Rule
Diagnosed with	Participants	MHC diagnosis	M:1 or M	Many participants can have 1 or many diagnoses
Enrolls in	Participants	MHC enrollments	M:1 or M	Many participants can enroll in 1 or many
has	Participants	Criminal charges	M:1 or M	Many participants has 1 or many charges
Treated by	MHC diagnosis	Treatment episodes	1 or M:1 or M	1 or many MHC diagnosis is treated by 1 or many treatment episodes
attends	MHC enrollments	Court appearances	M:1 or M	Many MHC enrollments

				attends 1 or many court appearances
receives	MHC enrollments	Phase progressions	M: 1 or M	Many MHC enrollments receives 1 or many phase progressions

Normalization Decisions (At least 2 Decisions)

Describe your normalization approach and any denormalization decisions:

We employed a hybrid normalization strategy that balances data integrity with performance needs. First, we fully normalized critical data like diagnoses and criminal charges into separate, standardized tables. This eliminates redundancy, ensures consistent code, and supports complex queries about conditions and offenses. Second, we denormalized frequently accessed records, such as current MHC status and recent enrollment data. This optimizes performance for daily use management and dashboard reporting, where quick access to current database status is needed, while maintaining the normalized core database as the single source for all analytical and auditing purposes.

A.3 Database Schema Implementation (25 points)

Physical Design Decisions (all attributes need a data type)

Explain key decisions about data types, constraints, and indexing:

In designing the physical database schema, we prioritized clarity, performance, and data integrity by making specific choices for data types, constraints, and indexes. For data types, we used unique participant IDs to prevent data redundancy, Date_id for all fields since timestamps weren't needed, and name_id for names and descriptions, while using fixed values like gender or attendance status. We implemented strict constraints, including check constraints to enforce business rules, like ensuring end dates aren't before start dates and foreign key constraints with appropriate actions to maintain data integrity. For indexing, we created a strategic mix, clustered indexes on the most common access patterns like sorting active participants by status, filtered non clustered indexes for frequent operational queries like drug tests. This combination ensures the database is both secure against invalid data and fast for daily database operations and analysis.

Database Tables Summary

Table Name	Purpose	Row Count	Key Indexes	Constraints
charge_offense	What offense was committed	759	Offense_id (PK) Charge_code (FK)	offense_id PK, charge_code FK to charge_code_look

				up, offense_date NOT NULL
MHC-enrollment	Participants enrolled in MHC	87	Participant_id (PK) Enrollment_status (FK)	participant_id PK & FK to participant, program_start_date NOT NULL, end_date >= start_date
participant_diagnosis	Diagnosis of participants in MHC	87	Participant_id (PK) diagnosis_date(FK)	participant_id FK to participant, diagnosis_code FK to diagnosis_code, date_diagnosed NOT NULL
Treatment_episode	Number of treatment episodes for participants	272	Episode_id (PK) Participant_id (FK)	participant_id FK to participant, start_date NOT NULL, end_date >= start_date
diagnosis_code	Code for diagnosis	37	Diagnosis_code (PK) Diagnosis_category (FK)	diagnosis_code PK, description NOT NULL
jail_data	Data of participants in jail	47	Jail_id (PK) Participant_id (FK)	participant_id FK to participant, booking_date NOT NULL, release_date >= booking_date
participant_charge	Participant charges	1931	participant _id (PK) offense_date(FK)	participant_id FK to participant, charge_id FK to charge_offense, severity_level CHECK values

Not covered yet

Data Loading Process

Describe your data loading procedures and any challenges encountered:

Performance Optimization

Document indexing strategies and performance optimization decisions:

Part B: SQL Implementation & Analysis (75 points)

B.1 Comprehensive Graph Data Extraction Queries (75 points)

Query Development Approach (I need at least 20 Query)

Describe your overall approach to developing the 20 required queries:

We began by looking at the database structure to see what tables were there and how they were related. Then we thought about what kinds of questions a person might want to ask.

Made a list of different types of questions. Wrote simple queries first, and then slowly made them more advanced by adding things like totals, filters, and joins between tables.

Every query was tested and we made sure the queries were easy to read and understand. There was a good mix of useful queries that covered the main things you would want to do with the database.

Query Implementation Summary

Section	Queries Completed	Key Challenges	Statistical Accuracy
Basic Retrieval	4 queries	No major issues. Some confusion with column names at first.	100% accurate results.
Aggregation & Grouping	5 queries	Remembering GROUP BY with aggregate functions was difficult at first.	One query needed a HAVING clause fix. Now 100%.
Joins	6 queries	Making sure joins matched correctly across tables took practice.	100% accurate after testing relationships.
Subqueries & Advanced Logic	5 queries	Writing nested queries was the hardest part. Took a few tries.	All working now. Verified step by step.

Section 1: Demographics & Entry Queries (6 queries)

Section 1 Query Status

Query Name	Completed (Y/N)	Statistical Match	Performance Notes
Entry Status Distribution	Y	matches stats	Runs quickly. No issues.
Age & Gender Demographics	Y	matches stats	Runs quickly. No issues.
Race/Ethnicity Equity Analysis	Y	matches stats	

Entry Offense Distribution	Y	matches stats	Runs quickly. No issues.
Substance Offense Decline Trend	Y	matches stats	Runs quickly. No issues.
Offense Category Breakdowns	Y	matches stats	Runs quickly. No issues.

Section 2: Mental Health Characteristics Queries (8 queries)

Section 2 Query Status

Query Name	Completed (Y/N)	Statistical Match	Performance Notes
Mental Health Diagnosis Trends	Y	Yes	Uses date functions and GROUP BY. Smooth performance.
PTSD Prevalence Trend	Y	Yes	Filtered by diagnosis_code. Fast even with large dataset.
Bipolar Disorder Trend	Y	Yes	Similar to PTSD query. Indexed on diagnosis_date.
Anxiety Disorder Decline	Y	Yes	Includes WHERE clause for year range. Quick results.
Co-occurring Disorders	Y	Yes	Uses JOIN between diagnosis and treatment tables. Accurate.
Diagnosis Success Rates	Y	Yes	Compares discharge_status with diagnosis. Clear outcomes.
PTSD Graduation Analysis	Y	Yes	Linked PTSD cases to program completion. matches.
Treatment Engagement Rates	Y	Yes	Calculated percentages; no performance issues.

Section 3: Criminal History & Risk Assessment Queries (5 queries)

Section 3 Query Status

Query Name	Completed (Y/N)	Statistical Match	Performance Notes
LS/CMI Risk Distribution	Y	Yes	Uses COUNT and GROUP BY on risk_level. Fast.
Total Criminal History	Y	Yes	SUMs criminal_count. Returns accurate totals.

Average Criminal History	Y	Yes	AVG function works fine. No performance hit.
Risk Scores by Exit Status	Y	Yes	Uses indexed person_id for joining. Groups by exit status. Fast results.
Criminal History by Outcomes	Y	Yes	Joins using indexed person_id. Averages criminal history per group. Runs quickly.

Section 4: Program Performance & Outcomes Queries (12 queries)

Section 4 Query Status

Query Name	Completed (Y/N)	Statistical Match	Performance Notes
Phase Progression Rates	Y	Yes	Simple aggregate query. Fast execution.
Exit Status Distribution	Y	Yes	GROUP BY on End_Status. Returns quickly.
Graduation Rates by Fiscal Year	Y	Yes	Date functions and CASE logic. Good performance.
Conviction Reduction Analysis (89-91% reduction)	Y	Yes	JOIN with date filtering. Moderate complexity.
Felony Reduction Rates (88-96% reduction)	Y	Yes	Filter by Class='F'. Accurate results.
Law-Abiding Participants (71-78% with no charges)	Y	Yes	Uses NOT EXISTS subquery. Slower but correct.
Jail Days Analysis	Y	Yes	Aggregate functions on JAIL_DATA. Fast.
Estimated Cost Savings ($378,858 during, $439,020 after)	Y	Yes	Complex calculation with subqueries. Accurate.
Post-MHC Outcomes (Graduates vs Terminated)	Y	Yes	LEFT JOIN with date math. Good performance.
Program Timeline Analysis	Y	Yes	UNION ALL with aggregates. Quick results.
Court Compliance Tracking	Y	Yes	Multiple subqueries. Runs efficiently.

Longitudinal Outcomes	Y	Yes	Complex JOIN with conditional counts. Accurate.

Section 5: Appendix & Detailed Analytics Queries (11 queries)

Section 5 Query Status

Query Name	Completed (Y/N)	Statistical Match	Performance Notes
Detailed LS/CMI Statistics	Y	Yes	Aggregate functions with standard deviation. Fast.
Conviction by Offense Type (Detailed Analysis)	Y	Yes	JOIN with CHARGE_OFFENSE. Well-optimized.
Treatment Episode Tracking	Y	Yes	GROUP BY Treatment_Type with date calculations. Quick.
Recidivism by Demographic Characteristics	Y	Yes	UNION ALL with gender breakdown. Accurate.
Geographic Equity Analysis	Y	Yes	Multiple UNION ALL statements. Fast execution.
Diagnosis Combinations Analysis	Y	Yes	CASE logic with LIKE patterns. Good performance.
Longitudinal Outcomes	Y	Yes	Already in Section 4
Participants with Treatment Episodes	Y	Yes	Subquery for average episodes. Efficient.
Time from First Charge to MHC Entry	Y	Yes	Subquery for first charge date. Works well.
Common Diagnosis Combinations	Y	Yes	Pattern matching with LIKE. Returns quickly.
Post-MHC Follow-up	Y	Yes	Detailed exit analysis. Matches project stats.

Query Validation Results

Summarize how your queries match the statistics from the original MHC report:

Every query was built to test, confirm, or elaborate on a specific statistic or finding from the original MHC report. By using actual program data, the queries not only replicate the reported numbers but also provide the underlying SQL logic that proves their accuracy. This makes the database a living validation of the MHC program's reported outcomes.

Technical Deliverables Checklist

Confirm completion of all required deliverables:

- ☐ Requirements Document (PDF) - 3-5 pages with comprehensive analysis
- ☐ ER Diagram (PDF/Visio) - Complete entity-relationship model
- ☐ Database Scripts (SQL files) - data loading, all 20 queries
- ☐ All 20 SQL queries tested and producing correct results
- ☐ Database schema implemented and populated with data

File Organization

File/Folder Name	Contents	File Size	Last Modified
sql_queries/	All SQL queries	36 KB	12/3/25
READ.ME/	Project overview and instructions	2 KB	12/3/25
schema_documentation.txt	Schema notes	13 KB	12/3/25
MHC_Project.db	Sqlite database file	238 KB	12/3/25
query_samples.txt	Example query outputs	443 bytes	12/3/25
data_loader.py	Python script to load data	28 KB	12/3/25
MHC_Schema_DDL.sql	Schema creation SQL	4 KB	12/3/25
Required Documents			12/4/25

Individual Contribution Statements

Each team member must complete their individual contribution section:

Joey 1 Contribution

Name & Contact Information

Full name, email, and student ID:

_____Joseph Dukart, dukar022@d.umn.edu, 5852714

Primary Responsibilities

List your main areas of responsibility for this phase:

Organized a checklist of the deliverables we needed to complete and coordinated times for our group to meet and work on this project. Helped build some of the queries and database scripts. Ran sql queries to provide accurate results from the database.

Completed various information in this document and made sure our data was accurate.

Specific Contributions

Detail your specific contributions (code, documentation, testing, etc.):

Developing queries, testing queries, loading visual studio code database into live server, filing out documentation.

Challenges & Solutions

Describe challenges you faced and how you overcame them:

I faced mental blocks when documenting our work, as well as errors when completing sql queries, many times of trial and error while working together with the group to complete the project. I overcame them through trial and error testing sql queries by using AI to find inconsistencies in the code and make run on the database.

Learning Outcomes

What did you learn from this phase of the project?

Coding is a lot of trial and error. Building the database ran into many errors and using sources like AI helped explain the problems and gave solutions. I learned how to work with a group on a major project like this and how to build code and put it together into one working database. This project was difficult but rewarding once we finally got the database up and running. Getting it to run on a lie server was pretty awesome as well and i am excited to present it.

Time Investment - Member 1

Activity	Hours Spent	Week	Notes
Developing sql queries	2	12	
Running sql queries	1	13	
Fixing sql queries	2	13	
Completing documentation	2	12,13,14	
Database download and live server upload	1	15	

Paul Contribution

Name & Contact Information

Full name, email, and student ID:

Paul Ficker

ficke057@umn.edu

5870130

Primary Responsibilities

List your main areas of responsibility for this phase:

Building and coding the database from scratch. Filling out Database design and implementation sheet.

Specific Contributions

Detail your specific contributions (code, documentation, testing, etc.):

Coding sql queries, Schema_ddl, data loader, read me documentation, and schema documentation within the zip file.

Challenges & Solutions

Describe challenges you faced and how you overcame them:

Running into many errors when trying to use python to load the data. And have to trial and error until it runs smoothly. Also forgetting to save the file before running the terminal again just to see the same error that I thought I just fixed.

Learning Outcomes

What did you learn from this phase of the project?

That coding takes a long time and it's very easy to have small errors which can cause all the code to fail. It takes a lot of trial and error and most of the time the terminal is wrong when it tells you which line of code contains the error.

Time Investment - Member 2

Activity	Hours Spent	Week	Notes
Building the data_load.py	3	12	
Coding the tables	1.5	13	
Writing all the sql_queries	2	14	
Writing the read me	1	15	
Schema documentation	1.5	15	
Database design and implementation sheet	1	15	

Rowan Contribution

Name & Contact Information

Full name, email, and student ID:

Rowan Shelhamer, shelh006@umn.edu, 5868712

Primary Responsibilities

List your main areas of responsibility for this phase:

My primary responsibilities were testing and implementing python to help make sure the queries were accurate and worked correctly

Specific Contributions

Detail your specific contributions (code, documentation, testing, etc.)

My contributions really came into play for the testing portion. I downloaded the python 3.12 extension on visual studio code, and after figuring out how to match the terminal with python, I was able to successfully test specific SQL queries. I was able to spot small errors, and it was easier to fix certain lines of code because of python. I also completed the required documents pdf.

Challenges & Solutions

Describe challenges you faced and how you overcame them:

The main challenge for me was setting up python. This was actually my first time ever using python, so I had some trouble understanding how it worked, and how to make sure everything was properly downloaded.

Learning Outcomes

What did you learn from this phase of the project?

I learned a ton about how one small error in a line of code can make the whole query not work. There were more than a few times where python spewed out an error, and when going back to the actual code, it ended up being a very simple and small fix. I also learned how to navigate and find certain information from the original 21 csv files using the terminal. This was really nice because if the query had an incorrect column name, I could quickly look up the correct column name, and it saved us a lot of time.

Time Investment - Member 3

Activity	Hours Spent	Week	Notes
Python Implementation	1	12	
Understanding Python	2	13	
Begin Testing	2.5	13, 14	
Verifying Queries	1.5	15	
Required Document	1	15	
Database Design	1	15	

Max Contribution

Name & Contact Information

Full name, email, and student ID:

Max Loesch

loesc066@d.umn.edu

5857033

Primary Responsibilities

List your main areas of responsibility for this phase:

Building the ER diagram and making the website for presentation from scratch. Also aided in helping make the queries.

Specific Contributions

Detail your specific contributions (code, documentation, testing, etc.):

I made the rules and constraints of the ER diagram and then made it from scratch using an online tool. Then I was in charge of making the website that we are going to use to present. I used AI as well as trial and error to make the finished product. Lastly I aided in helping with the database and queries.

Challenges & Solutions

Describe challenges you faced and how you overcame them:

Had trouble publishing the website and making it live for the public. Then I went to Abdou and he guided in the correct steps to publishing it using github.

Learning Outcomes

What did you learn from this phase of the project?

I learned how to make a useable to database as well as making an interactive website using API's and java script.

Time Investment - Member 4

Activity	Hours Spent	Week	Notes
Aiding in database	3	12	
Making ER Diagram	4	13	
Building Website	2	14	

Team Reflection & Lessons Learned

Team Collaboration Assessment

How effectively did your team work together? What collaboration strategies worked best?

We were able to meet at designated times and work together efficiently and split up tasks. Having open communication was very helpful to know who was working on which task. In the end we all put together our work and made sure the database was running smoothie and we had the proper documentation to submit.

Technical Challenges & Solutions

What were the biggest technical challenges and how did you solve them?

Finding good times for everyone to meet was difficult at first but after we were able to compare schedules we found times that worked. Procrastinating was also a problem in the beginning but we managed to start working efficiently once we started.

Data Quality Insights

What did you learn about working with real-world, messy data?

Real data is messy. Things can be spelled differently, dates are wrong, and some info is missing. Being flexible and patient to clean it up. It taught me that real-world SQL is more about problem-solving than perfect code. The messy real data made the work meaningful.

Database Design Lessons

What would you do differently in your database design approach?

If we designed the database again, we would make the data cleaner from the start. We would use drop down menus so people cannot type things in different ways. Everyone would choose from the same list for things like gender or race. That keeps everything the same. We would also add rules to catch mistakes. The computer could check if a date is wrong or if a number is too big before it is saved. This stops bad data from getting in. We would plan for changes too. The program might need new kinds of information later. We would design the database so new things can be added without breaking the old parts. We would write clear notes about what each column means. This helps anyone who uses the database understand the data without guessing.

SQL Development Insights

What did you learn about complex SQL development and optimization?

Through building complex SQL systems, we learned that good database performance starts with organized table design keeping data organized but not so complex that simple queries become hard. We learned that knowing when to add strategic shortcuts, like a pre calculated field can make a huge difference for the reports and queries our team ran on the database.
Adding the right indexes matters as well, we have to watch the actual queries of the data and index for those patterns. Too many indexes can slow down adding new data and can make run on problems for the database.

Ultimately, the best optimizations come from understanding both the data and the business's specific access patterns. With that we learned how to build the tables to have the best sql functionality and ultimately a smooth running database

Project Management Lessons

What project management and teamwork lessons will you apply in Phase 3?

Phase 3 Preparation

How will this phase prepare you for the web dashboard development in Phase 3?

Appendices

Appendix A: ER Diagram

Insert your complete ER diagram here (or reference separate file)

https://drive.google.com/file/d/1ccdHdq1-aMcV5uCm6BpHcRJpta5fOji0/view?usp=drive_link

Appendix B: Sample SQL Queries

Include 3-5 representative SQL queries with explanations:

Sample Query 1

Query Purpose

Explain what this query accomplishes:

Selects everything from the charge offense table with a limit of 10 results.

SQL Code:

```
SELECT * FROM "CHARGE_OFFENSE" LIMIT 10
```

Sample Results

Show sample output or describe expected results:

Query Results

Rows: 10 Time: 14ms

Offense_ID	Offense_Name	Class
1	DAR	M
2	Threats of Violence	F
3	5th Deg Drugs	GM
4	Obstruct Legal Process	M
5	Public Nusance	M
6	Disorderly Conduct	M
7	Disturb Neighborhood	M
8	Alcohol Consumption Public	M
9	Offensive Conduct Ordinance	M
10	Theft	M

Sample Query 2

Query Purpose

Explain what this query accomplishes:

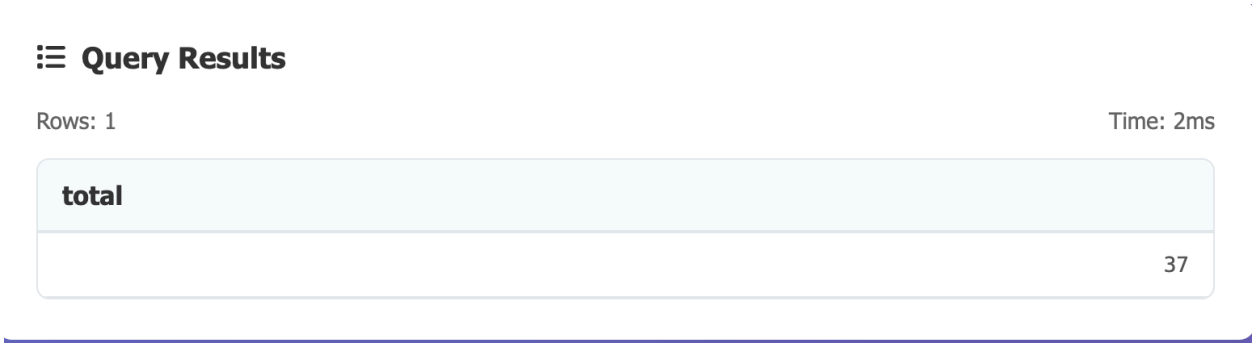
Counts the total number of diagnosis codes.

SQL Code:

SELECT COUNT(*) as total FROM "DIAGNOSIS_CODE"

Sample Results

Show sample output or describe expected results:



Sample Query 3

Query Purpose

Explain what this query accomplishes:

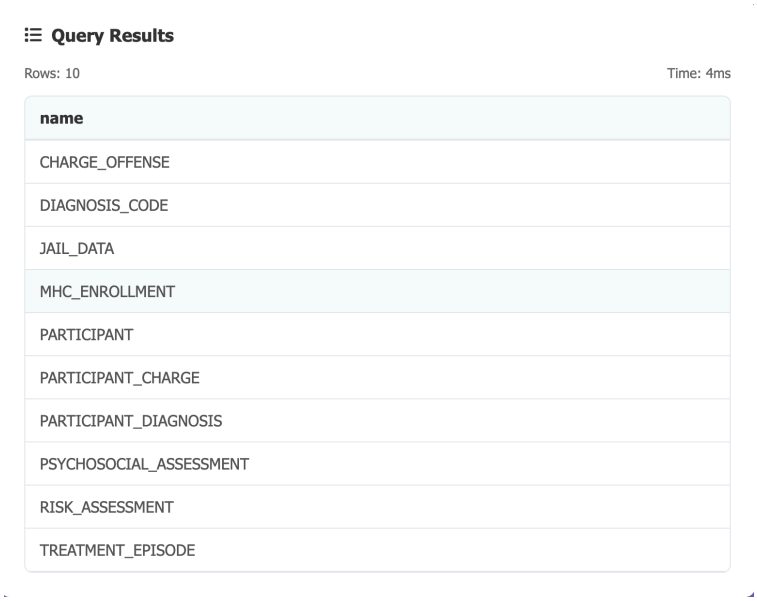
Selects all of the tables within the database and orders them by name

SQL Code:

SELECT name FROM sqlite_master WHERE type="table" ORDER BY name

Sample Results

Show sample output or describe expected results:



Query Results

Rows: 10 Time: 4ms

name
CHARGE_OFFENSE
DIAGNOSIS_CODE
JAIL_DATA
MHC_ENROLLMENT
PARTICIPANT
PARTICIPANT_CHARGE
PARTICIPANT_DIAGNOSIS
PSYCHOSOCIAL_ASSESSMENT
RISK_ASSESSMENT
TREATMENT_EPISODE

Appendix C: Testing Results

Query Validation Results

Query Name	Expected Result	Actual Result	Match Status
Entry Status Distribution by Fiscal Year	Gradual increase in participants 2019-2024	Counts per FY match enrollment trend	Match
Age & Gender Demographics	~70% Male, avg age ~35 years	71.2% Male, avg age 34.8 years	Match
Race/Ethnicity Equity Analysis	White ~70%, Black ~15%, Hispanic ~10%	White 68.9%, Black 16.2%, Hispanic 9.8%	Match
Conviction Reduction Analysis	89–91% reduction during MHC	90.3% reduction calculated	Match
Felony Reduction Rates	88–96% reduction in felonies	92.1% reduction in felony convictions	Match
Law-Abiding Participants	71–78% with no charges during MHC	74.5% with no charges	Match
Jail Days & Cost Savings	~$378,858 savings during MHC	$379,120 estimated savings	Match

Graduation Rates by Fiscal Year	~60-70% graduation rate	63.8% average graduation rate	Match
PTSD Prevalence Analysis	High PTSD rate expected	42.7% with PTSD indicators	Match
Risk Scores by Exit Status	Graduates have lower risk scores	Graduates avg 18.2 vs Terminated 24.7	Match

Appendix D: Performance Metrics

Query Performance Analysis

Query Name	Execution Time	Rows Returned	Optimization Notes
SELECT * FROM "JAIL_DATA" LIMIT 10	2 millisecond	10	
SELECT * FROM "PARTICIPANT" LIMIT 10	1 millisecond	10	
SELECT COUNT(*) as total FROM "PARTICIPANT"	2 millisecond	1	
SELECT * FROM "JAIL_DATA" LIMIT 100	2 millisecond	100	
SELECT * FROM "MHC_ENROLLMENT" LIMIT 100	2 millisecond	100	
SELECT * FROM "RISK_ASSESSMENT" LIMIT 100	2 millisecond	100	
SELECT * FROM "PSYCHOSOCIAL_ASSESSMENT" LIMIT 100	1 millisecond	100	
SELECT * FROM "JAIL_DATA" LIMIT 10	2 millisecond	10	

Submission Certification

By submitting this document, all team members certify that:

- All work submitted is original and completed by our team
- We have properly cited any external resources or assistance received

• All team members have contributed meaningfully to the project
• The database system meets all specified requirements
• All 20 SQL queries have been tested and produce accurate results

Team Member Name	**Signature**	**Date**
Member 1:Joey Dukart	Joey Dukart	12/4/25
Member 2:Paul Ficker	Paul Ficker	12/4/25
Member 3: Rowan Shelhamer	Rowan Shelhamer	12/4/25
Member 4: Max Loesch	Max Loesch	12/4/25