Databases-Week03

-Tao Nijia, He Yiyang, Liang Ruyi

BASIC TASKS

Task 1.

Question:

1. Define the following terms:
   a. Candidate key
   b. Composite key
   c. Foreign key
   d. Functional dependency

The answer:

a. Candidate key: A candidate key is a minimal set of attributes within a table that can uniquely identify each record in that table. There can be one or more candidate keys in a table.

b. Composite key: A composite key is a key that consists of two or more attributes that together uniquely identify a record in a table.

c. Foreign key: A foreign key is an attribute or a set of attributes in one table that refers to the primary key of another table. It is used to establish a relationship between two tables and maintain referential integrity.

d. Functional dependency: In a database, a functional dependency is a relationship between two sets of attributes in a table, where the value of one set of attributes (the dependent attributes) is determined by the value of another set of attributes (the determinant attributes). It is denoted as A -> B, where A is the determinant and B is the dependent.

Task 2.

Question:

2. Identify and define the three integrity rules/constraints in the relational model

The answer:

In the relational model, the three integrity rules/constraints are:

1. Entity Integrity: This rule states that the primary key of a table must have a unique value and cannot be null. Each row in the table must be uniquely identifiable by the primary key.

2. Referential Integrity: This rule ensures that the values of foreign keys in a table must match the values of the primary keys in the referenced table. If a foreign key value exists in a table, it must refer to an existing row in the referenced table.

3. Domain Integrity: This rule specifies that the values entered into a column must be of the correct data type and must fall within a specified range or set of values.

It ensures the validity and consistency of the data stored in the database.

Task 3

Question:

3. For the following relational tables where the primary key is underlined identify any violation(s) of the relational integrity rules:

Table 1

Film

filmNo	title	directorNo	country	year	genre
005	Reservoir Dogs	101	US	1992	Crime
006	Pulp Fiction	101	US	1994	Crime
008	Trainspotting	322	UK	1996	
109	Infernal Affairs	166	China	2002	Crime
111	Snakes on a Plane	753	US	2006	Disaster

Director

directorNo	dName
101	Tarantino
322	Boyle
166	Wai-Keung

Table 2

Supplier	PartNo	Quantity	Price
JKC	1234	750	35.00
JKC	67A	500	20.74
JKC	67B	1000	10.23
Aziz	Cd12	ABC	70.12
Smith	0242	30	54.34
Smith	AB/1	2000	45.67

Table 3

Song	Artist	Position	Month	Year
Woman	John Lennon	1	02	1981
Make Your Mind Up	Bucks Fizz	2	04	1981
You Can't Hurry Love	Phil Collins	11	02	1983
It's a Hard Life	Queen	9	07	1984
A Kind of Magic		5	04	1985
Toy Boy	Sinitta	6	09	1987
You Got It	Roy Orbison	4	02	1989

The answer:

Table1：

1. The line filmNo: 008 is missing the value of genre.
2. There is no directorNo value for "Snakes on a Plane" stored in the Table Directors: 753.

Table2：

Violation of referential integrity: The value of the Price column of the Aziz row is ABC, which does not meet the numerical requirements, which may lead to data inconsistency and affect the referential integrity (assuming that the correct data is a numerical value).

Table3：

The song A Kind of Magic is missing the primary key Artist value

MEDIUM TASKS

Task 4

Question:

4. Consider the following relational table on projects and the employees who work on them (Project-Employee). The schema of the relations is as follows, with the primary key underlined: Project-Employee (<u>ProjectCode</u>, ProjectTitle, ProjectManager, ProjectBudget, <u>EmployeeNo</u>, EmployeeName, DepartmentNo, DepartmentName, HourlyRate):

ProjCode	ProJTitle	ProjManager	ProjBudget	EmpNo	EmpName	DeptNo	DeptName	HrlyRate
PRC10	Payroll System	M Scott	24500	S10001	J Kirk	L004	IT	£22.00
PRC10	Payroll System	M Scott	24500	S10030	L Jones	L023	Pensions	£18.50
PRC10	Payroll System	M Scott	24500	S21010	P Lewis	L004	IT	£21.00
PRC45	Pension System	L McCoy	17400	S10010	B Jones	L004	IT	£21.75
PRC45	Pension System	L McCoy	17400	S10001	J Kirk	L004	IT	£18.00
PRC45	Pension System	L McCoy	17400	S31002	T Gilbert	L028	Database	£25.50
PRC45	Pension System	L McCoy	17400	S13210	W Richards	L008	Salary	£17.00
PRC64	CRM System	P Chekov	12250	S31002	T Gilbert	L028	Database	£23.25
PRC64	CRM System	P Chekov	12250	S21010	P Lewis	L004	IT	£17.50
PRC64	CRM System	P Chekov	12250	S10034	B James	L009	HR	£16.50

Identify the anomalies that may result from the following operations on the table (hint: INSERT, UPDATE, DELETE anomalies)

a. Insertion Management wishes to create and insert a new project into the table: Projectcode= PrC30, ProjectTitle= Skills Matrix, ProjectManager = M. Uhura, ProjectBudget = 20000;
b. Deletion Project ‘PrC10’ ended abruptly and was deleted from the table.
c. Modification Due to her outstanding performance ‘J Kirk’ was moved from department ‘L004’ to department ‘L009’
d. Derive 1NF
e. Derive 2NF
f. Derive 3NF

The answer:

a. The new project lacks a primary key called EmployeeNo

b. Deleting Prc10 will lose the information of employees

c. Moving J Kirk from department L004 to Department L009 results in inconsistent department information in multiple related records

d. 1NF

Analyze the original table structure

The original ProjectEmployee table contains a variety of information about projects and employees, For example, ProjectCode, ProjectTitle, ProjectManager, ProjectBudget, EmployeeNo, EmployeeName, DepartmentNo, DepartmentName, and HourlyRate. Where there may be multiple employees involved for each project, there are issues of data redundancy and duplicate groups.

Steps to convert to 1NF

Split the duplicate sets of data so that each tuple (row) represents an employee's information on a project. In this way, each cell in the table contains only one atomic value, meeting the 1NF requirement. For example, if there is a project PRC10 in the original table with three employees involved, there will be three rows of data in the table 1NF that correspond to the information of those three employees in the PRC10 project.

e. 2NF

Project: Includes ProjectCode (primary key), ProjectTitle, ProjectManager, and ProjectBudget. This relationship mainly stores basic information about the project.

Employee relationship: Contains EmployeeNo (primary key), EmployeeName, DepartmentNo, DepartmentName, and HourlyRate. This relationship mainly stores basic information about employees.

f. 3NF

Employee relationship is further decomposed into two relationships:

EmployeeBasic: Contains EmployeeNo (primary key), EmployeeName, and DepartmentNo. This relationship stores the employee's basic information and department number.

Department: Includes DepartmentNo (primary key) and DepartmentName. This relationship stores department information separately.

Through such decomposition, the transfer function dependence is eliminated and the requirement of 3NF is satisfied.

Task 5

Question:

5. Consider the following scenario. A bakery uses Excel to keep track of customer information and their orders. The bakery is growing and they want to store their data in a relational database. The first step to creating a database is to take the unnormalized dataset below and apply the normalisation process to create the database schema for the bakery. Normalization will result in a number of tables (entities) and columns (attributes). You will define keys for each of the tables, both primary and foreign (the columns used to link tables).

Order No.	Acc. No.	Customer	Address	Date	Item	Qty.	Price	Total Cost
7823	178	Daisy's Café	27 Bay Drive, Cove	16-Jul	Bakewell Tart	20	0.15	£12.35
7823	178	Daisy's Café	27 Bay Drive, Cove	16-Jul	Danish Pastry	13	0.20	£12.35
7823	178	Daisy's Café	27 Bay Drive, Cove	16-Jul	Apple Pie	45	0.15	£12.35
4633	526	Smiths	12 Dee View, Aberdeen	16-Jul	Butteries	120	0.20	£24.00
2276	167	Sally's Snacks	3 High Street, Banchory	17-Jul	Apple Pie	130	0.15	£56.50
2276	167	Sally's Snacks	3 High Street, Banchory	17-Jul	Cherry Pie	100	0.18	£56.50
2276	167	Sally's Snacks	3 High Street, Banchory	17-Jul	Steak Pie	30	0.50	£56.50
2276	167	Sally's Snacks	3 High Street, Banchory	17-Jul	Meringue Pie	20	0.20	£56.50
1788	32	Tasty Bite	17 Wood Place, Insch	18-Jul	Apple Pie	15	0.15	£7.50
1788	32	Tasty Bite	17 Wood Place, Insch	18-Jul	Danish Pastry	50	0.20	£7.50

a. What anomalies can you identify in the table? (HINT: INSERT, UPDATE, DELETE anomalies)
b. Derive 1NF
c. Derive 2NF
d. Derive 3NF

The answer:

a. nomaly analysis

1. Insert exception: When inserting a new order, if there is No customer information (such as ac.no. Is empty), it may not be inserted correctly.

2. Deletion exception: Deletion of an order may result in loss of customer information.

3. Modification exception: To modify the customer address, it needs to be modified in multiple order records, which is prone to error

b. 1NF

Split the item information in each order so that each item has its own separate row and each cell in the table contains only one atomic value. In this way, the data in the table meets the requirements of 1NF. For example, if an order contains three items, there will be three rows of data in the 1NF table for each item.

b. 2NF

Decompose the table into three new relationships:

1. Order relation (Order) : Contains Order No. (primary key), Date, Acc.No., Customer, Address. This relationship mainly stores the basic information of the order and the customer information.
2. Item relation: Contains Item (primary key), Qty., and Price. This relationship mainly stores basic information about the product.

3. OrderItem relation: contains Order No. (foreign key, refer to the Order No.), Item (foreign key, refer to the Item of the commodity relation). This relationship is used to establish many-to-many connections between orders and goods.

d. 3NF

The order relationship is further decomposed into two relationships:

1. OrderBasic information relationship (OrderBasic) : contains Order No. (primary key), Date, Acc.No. This relationship stores the basic information of the

order and the customer account number.

2. Customer relationship (Customer) : Contains AC.no. (primary key), Customer, and Address. This relationship stores customer information separately.

Task 6

Question:

6. Given the following tables, see below:

   a. Produce the schema for each table
   b. Identify the relations between the tables and specify the links between the foreign keys and their corresponding primary keys
   c. Generate the corresponding ER diagram

Patient

PatientNo	Surname	FirstNname
A102	Losey	Josef
B372	Shahine	Youcef
B543	Kurusawa	Akira
B444	Ray	Satyajit
S555	Renoir	Jean

Admission

PatientNo	Admitted	Discharged	Ward
A102	2/05/2017	9/05/2017	A
A102	2/12/2016	2/01/2017	A
S555	5/10/2017	3/12/2017	B
B444	1/12/2016	1/01/2017	B
S555	5/07/2017	1/08/2017	A

Doctor

DoctorNo	Surname	FirstName	Ward
203	Ousmane	Sembene	A
501	Coppola	Sofia	A
574	Almodovar	Pedro	B
461	Kubrik	Stanley	B
530	Fellini	Federico	C
405	Boyle	Danny	A

Ward

Ward	WardName	DoctorNo-InCharge
A	Surgical	203
B	Paediatric	574
C	Medical	530

The answer:

a. Patient(PatientNo,Surname,FirstName)

Admission(PatientNo,Admitted,Discharged,Ward)

Doctor(DoctorNo,Surname,FirstName,Ward)

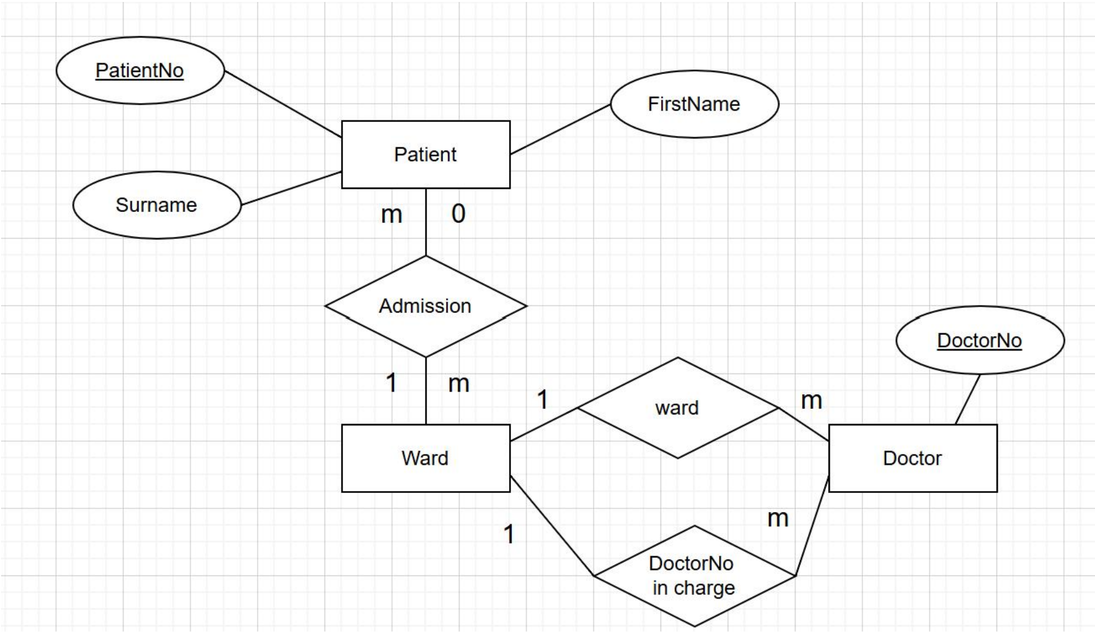
Ward(Ward,WardName,DoctorNo-InCharge)

b. The relationships between tables are as follows: The Patient table and the Admission table are associated through PatientNo.

Admission tables and Ward tables are associated by Ward.

The Doctor table and Ward table are associated by Ward.

c.


PatientNo
FirstName
Patient
Surname
m
0
Admission
DoctorNo
1
m
1
ward
m
Ward
Doctor
m
1
DoctorNo
in charge

ADVANCED TASKS

Task 7.

Question:

7. For each of the following tables (see Table 1 and Table 2 below):

   a. Identify the functional dependencies
   b. Use the example considered in the lecture and generate with justification 1NF, 2NF and 3NF tables

Table 1: Catering

Order No.	Account No.	Customer	Address	Date	Item	Quantity	Item Price
7823	178	Daisy's Café	27 Bay Drive, Coventry	16/7	Bakewell Tart	20	0.15
					Danish Pastry	13	0.20
					Apple pie	45	0.15
4633	526	Smiths	12 Dee View, Aberdeen	16/7	Butteries	120	0.20
2276	167	Sally's Snacks	3 High Street, Banchory	17/7	Apple pie	130	0.15
					Cherry Pie	100	0.18
					Steak pie	30	0.50
					Meringue Pie	20	0.20
1788	032	Tasty Bite	17 Wood Place, Liverpool	18/7	Apple Pie	15	0.15
					Danish Pastry	50	0.20

Table 2: Student records

Student No	Name	Course	Course Duration	Module No	Module name	Lecturer
1002	Salif Keita	G701	4	COF104	Java	Asimov
				COF118	Distributed Systems	Patel
1005	Emma Wilson	G504	3	COF105	Computer Architecture	Zidane
				COF118	Distributed Systems	Patel
				COF120	Operating Systems	Brando
1006	Hong Wang	G701	4	COF111	Networks	Austin
				COF105	Computer Architecture	Zidane
1010	Kiri Anahera	G722	2	COF111	Networks	Austin
				COF105	Computer Architecture	Zidane

The answer:

1.

Table 1

- Order No → Account No, Customer, Address, Date (Order

No determines the customer and their details)

- Order No, Item → Quantity, Item Price (The combination of Order No and Item determines the quantity and item price)

Table 2:

- Student No → Name, Course, Course Duration (A student number determines the student's details)
- Student No, Module No → Module Name, Lecturer (The combination of student number and module number determines the module name and lecturer)

2

Table 1

1NF

Table 1 is already in 1NF as all attributes contain atomic values, and there are no repeating groups.

2NF

- **Orders Table**:
    - Order No → Account No, Customer, Address, Date

- **Order Details Table**:
    - Order No, Item → Quantity, Item Price

3NF:

In this case, there are no transitive dependencies, as all non-key attributes directly depend on the primary key, so the table is already in 3NF after the 2NF step.

Table 2

1NF:

The table is already in 1NF since all the data values are atomic and there are no repeating groups.

2NF:

The primary key here is Student No, Module No. The non-key attributes Name, Course, Course Duration depend only on Student No, not Module No. So, we split the table to achieve 2NF:

- **Students Table**:
    - Student No → Name, Course, Course Duration
- **Student Modules Table**:
    - Student No, Module No → Module Name, Lecturer

3NF (Third Normal Form):

To ensure 3NF, we must remove any transitive dependencies. In this case, there are no transitive dependencies, so the tables are already in 3NF after the 2NF step.

8

a. Branch Code → Branch Name, Supervisor ID, Supervisor Name

- Car Plate No → Car Type
- Bill No → Bill Date, Penalty, Final Bill Amount
- Supervisor ID → Supervisor Name

b.

1NF

The data already meets the 1NF criteria, as each field contains atomic values and there are no repeating groups.

2NF

1. **Branch Table**:
    - Branch Code → Branch Name, Supervisor ID, Supervisor Name
2. **Car Table**:
    - Car Plate No → Car Type
3. **Bill Table**:
    - Bill No → Bill Date, Penalty, Final Bill Amount
4. **Branch-Car-Bill Table**:

- Branch Code, Car Plate No, Bill No (to associate branches, cars, and bills)

3NF:

1. **Branch Table**:
    - Branch Code → Branch Name, Supervisor ID
2. **Supervisor Table**:
    - Supervisor ID → Supervisor Name

Task 8

Question:

8. A car rental company produces quarterly reports containing information on each of the branches showing; branch, details, branch supervisor, car number plate, car type, bill number and date, penalty (late return or excess mileage for example), final bill amount, branch supervisor ID and name:

Report ID	Reporting Period	Branch Code	Branch Name	Car Plate Nr	Car Type	Bill Nr	Bill Date	Penalty	Final Bill	Supervisor ID	Supervisor Name
7686	January 2021 to March 2021	876734	Walsall	DS4049	SUV	166651	18.01.2021	50	1050	102	David Brown
7686	January 2021 to March 2021	876734	Walsall	DL3434	Sports Car	123111	19.02.2021	0	500	102	David Brown
7686	January 2021 to March 2021	876734	Walsall	OP9817	SUV	561909	06.03.2021	0	480	102	David Brown
7686	January 2021 to March 2021	876734	Walsall	SJ7182	Hatchback	565690	29.01.2021	0	680	102	David Brown
1056	October 2021 to December 2021	100023	Coventry	BN9745	SUV	128976	10.10.2021	0	710	871	Anna Smith
1056	October 2021 to December 2021	100023	Coventry	LA5142	Sedan	511899	25.11.2021	20	1500	871	Anna Smith
1056	October 2021 to December 2021	100023	Coventry	CB0098	Sports Car	141421	03.12.2021	0	850	871	Anna Smith
1056	October 2021 to December 2021	100023	Coventry	ZX7222	Coupe	514879	29.10.2021	0	1250	871	Anna Smith
1056	October 2021 to December 2021	100023	Coventry	DL3434	Sports Car	771100	16.11.2021	20	300	871	Anna Smith
4981	January 2022 to March 2022	456109	Leamington Spa	PO8123	SUV	675912	06.01.2022	50	350	149	John Cruise
4981	January 2022 to March 2022	456109	Leamington Spa	IU7878	Hatchback	991762	08.02.2022	0	950	149	John Cruise
4832	January 2022 to March 2022	981256	Warwick	NM8787	Sports Car	876234	14.02.2022	0	350	823	James Doherty
7002	July 2021 to September 2021	981256	Warwick	OP9817	SUV	110054	19.07.2021	100	1400	823	James Doherty
7002	July 2021 to September 2021	981256	Warwick	NM8787	Sports Car	378123	12.08.2021	20	450	823	James Doherty
7002	July 2021 to September 2021	981256	Warwick	VC1111	Sedan	808051	18.09.2021	0	670	823	James Doherty
7002	July 2021 to September 2021	981256	Warwick	FG7100	Hatchback	100023	21.07.2021	0	1030	823	James Doherty
7002	July 2021 to September 2021	981256	Warwick	RE6000	Sedan	611554	27.08.2021	50	520	823	James Doherty
3121	April 2021 to June 2021	555901	Wolverhampton	TR6199	SUV	888712	10.04.2021	0	490	111	Catherine Johnson
3121	April 2021 to June 2021	555901	Wolverhampton	DL3434	Sports Car	343412	28.05.2021	20	1230	111	Catherine Johnson
3121	April 2021 to June 2021	555901	Wolverhampton	BP9111	Coupe	222678	04.06.2021	0	1680	111	Catherine Johnson

a. Identify the functional dependencies
b. Complete the normalisation process for this dataset and derive 1NF, 2NF and 3NF tables.

The answer:

a.

Branch Code → Branch Name, Supervisor ID, Supervisor Name-

Car Plate No → Car Type` (Car plate number determines the car type)

Bill No → Bill Date, Penalty, Final Bill Amount

Supervisor ID → Supervisor Name

b. Normalization Process: 1NF, 2NF, and 3NF

1NF

The data already meets the 1NF criteria, as each field contains atomic values and there are no repeating groups.

2NF:

1. Branch Table:

-Branch Code → Branch Name, Supervisor ID, Supervisor Name

2. Car Table

Car Plate No → Car Type`

3. Bill Table:

-Bill No → Bill Date, Penalty, Final Bill Amount

4. Branch-Car-Bill Table:

-Branch Code, Car Plate No, Bill No

3NF:

1. Branch Table:

Branch Code → Branch Name, Supervisor ID

2. Supervisor Table:

Supervisor ID → Supervisor Name

9. CHALLENGE YOURSLEF!

Question:

9. **CHALLENGE YOURSLEF!**

Scenario. M70 is a small marine service company that carries out the boat maintenance. The company employs 7 engineers and maintains a list of contractors for specific tasks (when a specialist skills or expertise knowledge required). The company records the type of the marine equipment, servicing time and related maintenance tasks for each boat. The company records the list of engineers who carry specific tasks on a boat, servicing time, man-hours of the engineers, the time and the date of the service. The list of the tasks is as follows:

- Service
- Software upgrade
- Repair
- Safety inspection
- Other (customer specific requests)

a. Follow the life cycle of Database development and produce both Logical and Physical models of the Database for the M70 scenario.
b. Generate the corresponding the ER diagram(s)
c. Identify functional dependencies
d. Populate the developed relations with some test (fictitious) information
e. Complete the normalisation in order minimise redundancy and dependency in the database relations.

The answer:

I. Logical Model Design

- Entities:
    - Boat: Attributes include boat_id and equipment_type.
    - Engineer: Attributes are engineer_id and name.
    - Task: Attributes are task_id and task_type.
    - ServiceRecord: Attributes are service_record_id, service_time, man_hours, date, engineer_id (foreign key referencing Engineer), boat_id (foreign key referencing Boat), and task_id (foreign key referencing Task).
- Relationships:
    - Boat and Task: Many-to-many relationship.

- Engineer and ServiceRecord: One-to-many relationship.
- ServiceRecord and Task: One-to-many relationship.
- ServiceRecord and Boat: One-to-many relationship.

II. Physical Model Design

- Table Structures:
    - boats: boat_id (primary key), equipment_type.
    - engineers: engineer_id (primary key), name.
    - tasks: task_id (primary key), task_type.

  - service_records: service_record_id (primary key), engineer_id (foreign key), boat_id (foreign key), task_id (foreign key), service_time, man_hours, date.

IV. Functional Dependencies

- In service_records table, service_record_id determines all other attributes, i.e., service_record_id → engineer_id, service_record_id → boat_id, service_record_id → task_id, service_record_id → service_time, service_record_id → man_hours, service_record_id → date.
- In engineers table, engineer_id determines name, i.e., engineer_id → name.
- In tasks table, task_id determines task_type, i.e., task_id → task_type.
- In boats table, boat_id determines equipment_type, i.e., boat_id → equipment_type.

V. Populating Test Information

- boats table:

    - boat_id: 1, equipment_type: Fishing boat equipment.
    - boat_id: 2, equipment_type: Yacht equipment.
- engineers table:
    - engineer_id: 1, name: John.
    - engineer_id: 2, name: Mike.
- tasks table:
    - task_id: 1, task_type: Service.
    - task_id: 2, task_type: Software upgrade.
    - task_id: 3, task_type: Repair.
    - task_id: 4, task_type: Safety inspection.
    - task_id: 5, task_type: Customer specific request.
- service_records table:
    - service_record_id: 1, engineer_id: 1, boat_id: 1, task_id: 1, service_time: October 10, 2024, 8 am, man_hours: 4, date: October 10, 2024.
    - service_record_id: 2, engineer_id: 2, boat_id: 2, task_id: 2, service_time: October 11, 2024, 9 am, man_hours: 3, date: October 11, 2024.

VI. Normalization

- First Normal Form (1NF): Each attribute is atomic and indivisible. The design already meets 1NF.

- Second Normal Form (2NF): Eliminate partial functional dependencies of non-key attributes on candidate keys. In service_records table, service_record_id is the candidate key and all other attributes fully depend on it. In engineers table, engineer_id is the candidate key and name fully depends on it. In tasks table, task_id is the candidate key and task_type fully depends on it. In boats table, boat_id is the candidate key and equipment_type fully depends on it. So, it meets 2NF.
- Third Normal Form (3NF): Eliminate transitive functional dependencies. There are no transitive functional dependencies in the design, so it meets 3NF.