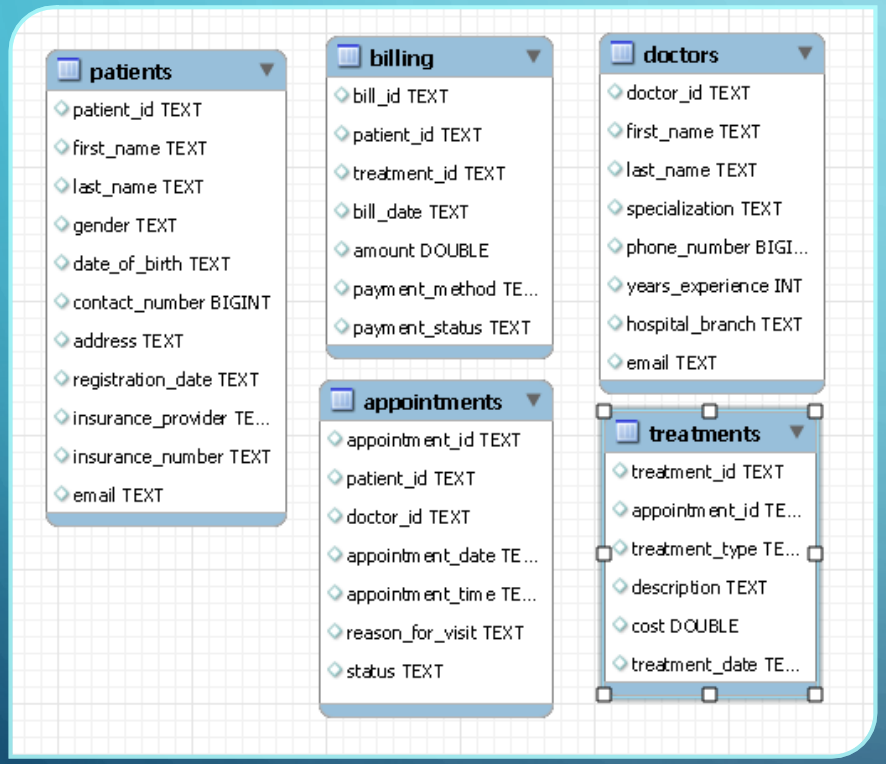
HOSPITAL MANAGEMENT SQL PROJECT SCHEMA BUILD AND ANALYTICS

DESIGNING DATABASES AND ANALYSING HOSPITAL DATA EFFICIENTLY

DATA DEFINITION & SCHEMA SETUP

INITIAL RAW SCHEMA



Baseline Raw Schema

The raw schema contains tables with TEXT columns limiting indexing and constraints.

Missing Keys and Constraints

No primary or foreign keys exist, preventing uniqueness and referential integrity enforcement.

Data Quality Risks

Lack of structure causes risks like duplicate records and orphaned rows affecting reliability.

Need for Schema Improvement

Applying enhancements is necessary to add constraints and standardise data types for consistency.

SQL SCRIPT: PRIMARY KEYS

Definition of Primary Keys

Primary keys uniquely identify each record and are essential for relational integrity in databases.

Role in Data Relationships

Primary keys enable foreign keys, allowing accurate joins and enforcing referential integrity.

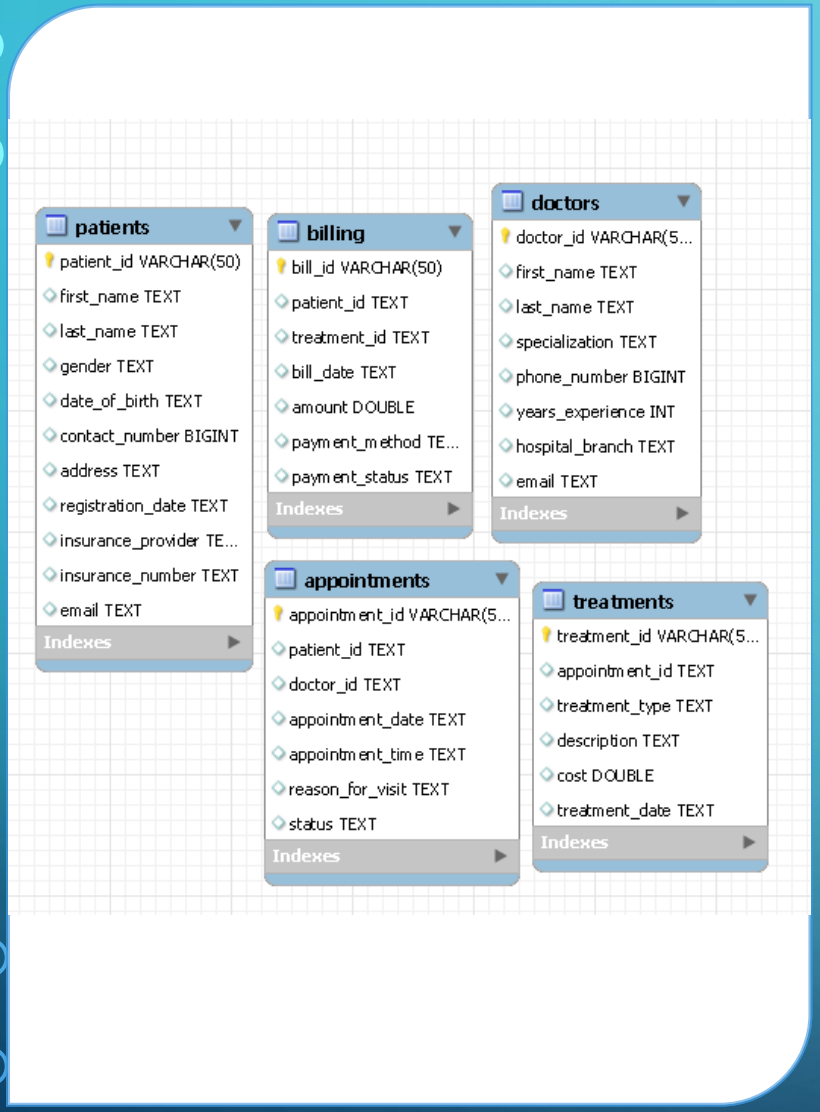
Performance Benefits

Indexed primary keys improve query performance and data retrieval efficiency in relational databases.

Foundation for Schema Design

Creating primary keys before foreign keys ensures a robust and well-structured relational schema.

```sql
USE hospital_management;

-- 1. Patients: patient_id is the PK
ALTER TABLE patients
MODIFY COLUMN patient_id VARCHAR(50) NOT NULL,
ADD PRIMARY KEY (patient_id);

-- 2. Doctors: doctor_id is the PK
ALTER TABLE doctors
MODIFY COLUMN doctor_id VARCHAR(50) NOT NULL,
ADD PRIMARY KEY (doctor_id);

-- 3. Appointments: appointment_id is the PK
ALTER TABLE appointments
MODIFY COLUMN appointment_id VARCHAR(50) NOT NULL
ADD PRIMARY KEY (appointment_id);

-- 4. Treatments: treatment_id is the PK
ALTER TABLE treatments
MODIFY COLUMN treatment_id VARCHAR(50) NOT NULL,
ADD PRIMARY KEY (treatment_id);

-- 5. Billing: bill_id is the PK
ALTER TABLE billing
MODIFY COLUMN bill_id VARCHAR(50) NOT NULL,
ADD PRIMARY KEY (bill_id);
```



SCHEMA PROGRESS AFTER PKS

Primary Key Assignment

Assigning primary keys structures the schema and reduces duplicate records, enhancing database reliability.

Logical Table Linking

Primary keys enable logical linking of tables reflecting real-world relationships like patients, doctors, and appointments.

Improved Data Integrity

Structural improvements from primary keys prepare the schema for referential constraints and stronger data integrity.

DATA STANDARDISATION & CLEANING

DATA TYPING & FOREIGN KEYS

Standardising Data Types

Converting ID columns from TEXT to VARCHAR(50) ensures type compatibility for relational integrity.

Establishing Foreign Keys

Foreign key constraints link related tables, such as appointments to patients and doctors, ensuring data consistency.

Maintaining Data Integrity

Constraints prevent orphan records and enable automatic validation during data insert and update operations.

```sql
-- DATA STANDARDIZATION & CLEANING
-- CSV imports often default to 'TEXT'. We convert IDs to 'VARCHAR(50)'
ALTER TABLE patients MODIFY patient_id VARCHAR(50);
ALTER TABLE doctors MODIFY doctor_id VARCHAR(50);
ALTER TABLE appointments MODIFY appointment_id VARCHAR(50), MODIFY patient_id VARCHAR(50), MODIFY doctor_id VARCHAR(50);
ALTER TABLE treatments MODIFY treatment_id VARCHAR(50), MODIFY appointment_id VARCHAR(50);
ALTER TABLE billing MODIFY bill_id VARCHAR(50), MODIFY patient_id VARCHAR(50), MODIFY treatment_id VARCHAR(50);

-- ESTABLISHING PRIMARY KEYS
ALTER TABLE patients ADD PRIMARY KEY (patient_id);
ALTER TABLE doctors ADD PRIMARY KEY (doctor_id);
ALTER TABLE appointments ADD PRIMARY KEY (appointment_id);
ALTER TABLE treatments ADD PRIMARY KEY (treatment_id);
ALTER TABLE billing ADD PRIMARY KEY (bill_id);

-- CREATING REFERENTIAL INTEGRITY (FOREIGN KEYS)
-- This ensures that every appointment, treatment, and bill is tied to a valid patient and doctor record.

-- Linking Appointments to Patients and Doctors
ALTER TABLE appointments
ADD CONSTRAINT fk_appt_patient FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
ADD CONSTRAINT fk_appt_doctor FOREIGN KEY (doctor_id) REFERENCES doctors(doctor_id);

-- Linking Treatments to Appointments
ALTER TABLE treatments
ADD CONSTRAINT fk_treat_appt FOREIGN KEY (appointment_id) REFERENCES appointments(appointment_id);

-- Linking Billing to Patients and Treatments
ALTER TABLE billing
ADD CONSTRAINT fk_bill_patient FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
ADD CONSTRAINT fk_bill_treatment FOREIGN KEY (treatment_id) REFERENCES treatments(treatment_id);
```

FINAL ERD DIAGRAM

Relational Schema Representation

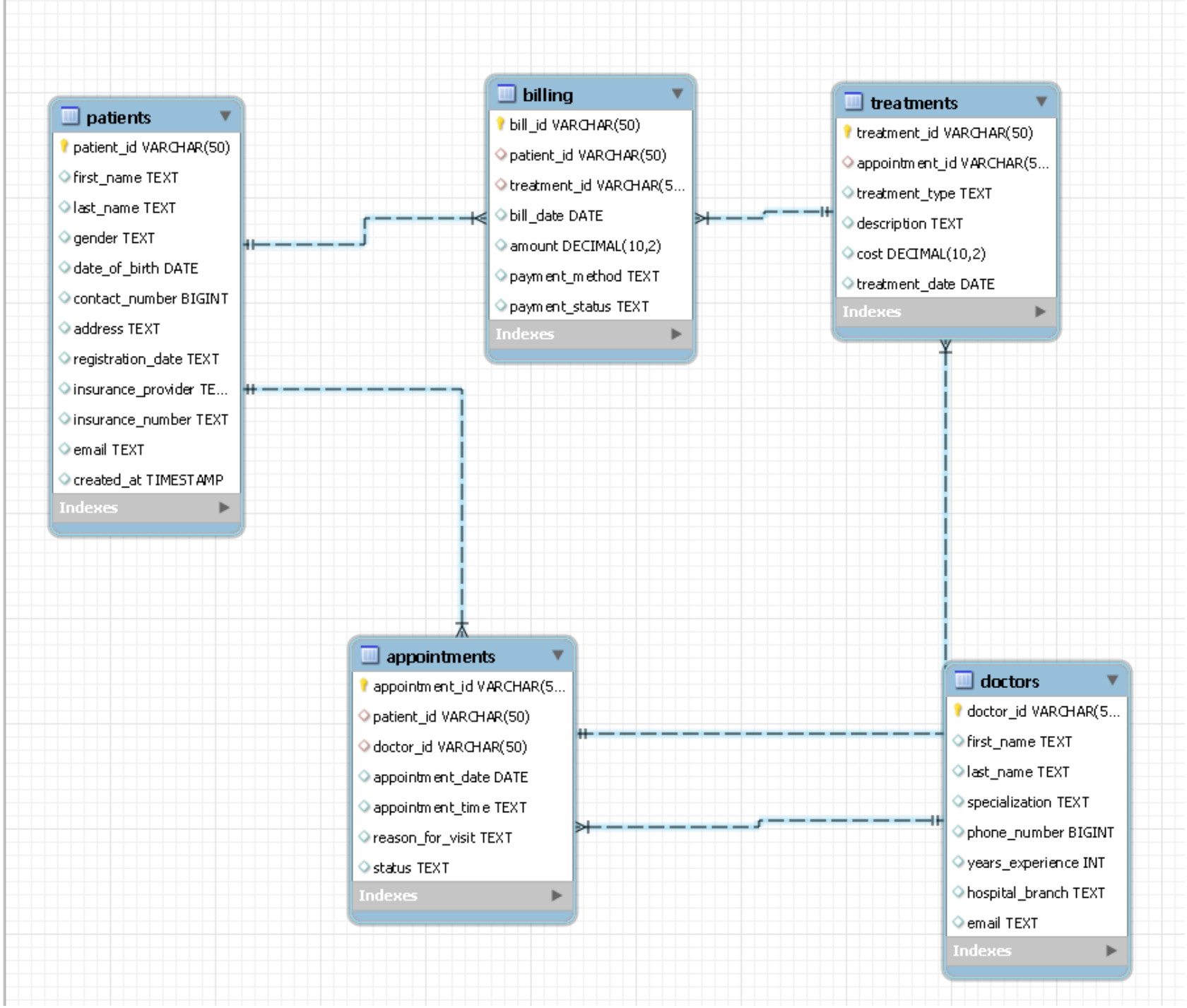
The Entity Relationship Diagram visually illustrates interconnected tables via primary and foreign keys modeling hospital workflows.

Workflow Integration

Patients, doctors, appointments, treatments, and billing entities are logically connected representing real hospital processes.

Database Robustness and Analytics

The structured schema supports reliable joins, accurate reports, and advanced analytics for clinical and financial insights.



DATA TRANSFORMATION

```sql
-- DATA TRANSFORMATION
-- Converting text-based date columns into proper SQL DATE types.

UPDATE patients SET date_of_birth = STR_TO_DATE(date_of_birth, '%Y-%m-%d');
ALTER TABLE patients MODIFY COLUMN date_of_birth DATE;

UPDATE appointments SET appointment_date = STR_TO_DATE(appointment_date, '%Y-%m-%d');
ALTER TABLE appointments MODIFY COLUMN appointment_date DATE;

UPDATE billing SET bill_date = STR_TO_DATE(bill_date, '%Y-%m-%d');
ALTER TABLE billing MODIFY COLUMN bill_date DATE;

UPDATE treatments SET treatment_date = STR_TO_DATE(treatment_date, '%Y-%m-%d');
ALTER TABLE treatments MODIFY COLUMN treatment_date DATE;

-- Converting currency columns from generic types to DECIMAL(10,2).

-- Update Billing table
ALTER TABLE billing
MODIFY COLUMN amount DECIMAL(10,2);

-- Update Treatments table
ALTER TABLE treatments
MODIFY COLUMN cost DECIMAL(10,2);

-- Adding a timestamp to the patients table that automatically records when a new record is created.

ALTER TABLE patients
ADD COLUMN created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP;

```

SQL SCRIPT: TYPE CONVERSION

- **Date Type Conversion**

Converting string dates to DATE type enables accurate filtering and time-based analysis.

- **Currency Precision**

Currency fields use DECIMAL(10,2) to avoid rounding errors and ensure financial accuracy.

- **Audit Timestamps**

Added timestamps like created at support traceability and enable trend analysis in data.

OPERATIONAL EFFICIENCY & HOSPITAL PERFORMANCE

```sql
92      -- OPERATIONAL EFFICIENCY & HOSPITAL PERFORMANCE
93
94      -- 1. PEAK BUSY DAYS
95
96  •   SELECT DAYNAME(appointment_date) AS day_of_week,
97      COUNT(*) AS total_appointments
98      FROM appointments
99      GROUP BY day_of_week
00      ORDER BY FIELD(day_of_week, 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday');
01
```

esult Grid | Filter Rows: | Export: | Wrap Cell Content:

day_of_week	total_appointments
Monday	26
Tuesday	37
Wednesday	37
Thursday	28
Friday	23
Saturday	23
Sunday	26

ANALYSIS PEAK BUSY DAYS

- **Identifying Peak Days**
- Analyse appointment data to pinpoint the busiest days of the week for patient visits and service demand.

- **Optimising Staffing and Rooms**
- Use peak day insights to allocate staff and rooms efficiently, reducing patient wait times and improving care quality.

- **Supporting Forecasting and Planning**
- Leverage analysis for seasonal planning and forecasting to ensure operational capacity meets patient demand effectively.

ANALYSIS DOCTOR WORKLOAD

Workload Distribution

Analysing appointment distribution helps balance workloads and prevent physician burnout for better healthcare delivery.

Capacity and Recruitment

Identifying high-demand doctors supports addressing capacity issues and guides recruitment planning effectively.

Optimising Resource Use

Monitoring workload patterns allows hospitals to optimise resources and maintain high-quality patient care.

```
102     -- 2. DOCTOR WORKLOAD
103     -- Measures total appointments per doctor to prevent burnout.
104 •   SELECT CONCAT('Dr. ', d.last_name) AS doctor_name, d.specialization,
105     COUNT(a.appointment_id) AS total_patients_seen
106     FROM doctors d
107     JOIN appointments a ON d.doctor_id = a.doctor_id
108     GROUP BY d.doctor_id
109     ORDER BY total_patients_seen DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

doctor_name	specialization	total_patients_seen
Dr. Taylor	Dermatology	29
Dr. Taylor	Dermatology	25
Dr. Davis	Pediatrics	24
Dr. Smith	Pediatrics	22
Dr. Davis	Pediatrics	21
Dr. Wilson	Oncology	19
Dr. Smith	Pediatrics	17
Dr. Brown	Dermatology	16
Dr. Jones	Pediatrics	14
Dr. Davis	Oncology	13

ANALYSIS NO-SHOW IMPACT

Operational Efficiency Impact

No-shows and cancellations disrupt hospital operations and reduce efficiency in patient scheduling and resource allocation.

Quantifying Missed Appointments

Analyzing missed appointments per physician helps identify problem areas for targeted intervention and improvement.

Improving Financial Performance

Reducing no-shows through reminders or overbooking optimizes resource use and boosts hospital revenue.

```sql
-- 3. NO-SHOW IMPACT ANALYSIS
-- Calculates missed appointment percentages and potential revenue loss.
SELECT CONCAT('Dr. ', d.last_name) AS doctor_name,
COUNT(a.appointment_id) AS total_scheduled,
SUM(CASE WHEN a.status = 'No-show' THEN 1 ELSE 0 END) AS total_no_shows,
ROUND((SUM(CASE WHEN a.status = 'No-show' THEN 1 ELSE 0 END) / COUNT(a.appointment_id)) * 100, 2) AS no_show_rate_percent
FROM doctors d
JOIN appointments a ON d.doctor_id = a.doctor_id
GROUP BY d.doctor_id
ORDER BY no_show_rate_percent DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

doctor_name	total_scheduled	total_no_shows	no_show_rate_percent
Dr. Jones	14	5	35.71
Dr. Smith	17	6	35.29
Dr. Smith	22	7	31.82
Dr. Taylor	29	9	31.03
Dr. Taylor	25	7	28.00
Dr. Wilson	19	5	26.32
Dr. Davis	24	6	25.00
Dr. Brown	16	4	25.00
Dr. Davis	13	2	15.38
Dr. Davis	21	1	4.76

ANALYSIS DEPARTMENTAL SUCCESS

```
122     -- 4. DEPARTMENTAL SUCCESS RATES
123     -- Compares completed visits versus cancellations by specialization.
124 •   SELECT d.specialization,
125     SUM(CASE WHEN a.status = 'Completed' THEN 1 ELSE 0 END) AS completed_visits,
126     SUM(CASE WHEN a.status = 'Cancelled' THEN 1 ELSE 0 END) AS cancellations,
127     COUNT(a.appointment_id) AS total_volume
128     FROM doctors d
129     JOIN appointments a ON d.doctor_id = a.doctor_id
130     GROUP BY d.specialization
131     ORDER BY completed_visits DESC;
132
133
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

specialization	completed_visits	cancellations	total_volume
Pediatrics	22	25	98
Dermatology	14	18	70
Oncology	10	8	32

Visit Completion Comparison

Comparing completed versus cancelled visits by specialisation highlights departmental performance variations.

Causes of High Cancellation

High cancellation rates suggest inefficiencies or communication gaps impacting patient scheduling.

Improvement Opportunities

Analysis enables targeted improvements, resource prioritization, and service redesign to optimize operations.

FINANCIAL & INSURANCE PERFORMANCE

```sql
-- FINANCIAL & INSURANCE PERFORMANCE
-- These queries analyze the hospital's cash flow, insurance pa

-- INSURANCE REVENUE STREAMS

SELECT p.insurance_provider,
COUNT(b.bill_id) AS total_invoices,
SUM(b.amount) AS total_paid_revenue
FROM patients p
JOIN billing b ON p.patient_id = b.patient_id
WHERE b.payment_status = 'Paid'
GROUP BY p.insurance_provider
ORDER BY total_paid_revenue DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

insurance_provider	total_invoices	total_paid_revenue
MedCare Plus	29	81109.48
WellnessCorp	19	45882.28
PulseSecure	13	38486.63
HealthIndia	3	7946.51

INSURANCE REVENUE STREAMS

Ranking Insurance Providers

The analysis ranks insurers by total payments and claim counts to identify major revenue sources.

Revenue Contribution Insights

Understanding payer mix helps hospitals manage contracts and reimbursement strategies effectively.

Financial Planning Support

Insights from insurer data assist in financial planning and risk management for hospitals.

AGING ACCOUNTS RECEIVABLE

Importance of Tracking Overdue Bills

Monitoring bills older than 30 days helps manage outstanding debt effectively and improve collection rates.

Prioritising Follow-Up Actions

The report helps prioritise accounts for follow-up to reduce aging receivables and enhance cash flow.

Mitigating Financial Risk

Hospitals can mitigate financial risks by closely monitoring overdue accounts and maintaining strong revenue cycles.

```
154    -- AGING ACCOUNTS RECEIVABLE
155
156 •  SELECT p.first_name, p.last_name, p.contact_number, b.amount AS pending_amount,
157    DATE_FORMAT(b.bill_date, '%d-%m-%Y') AS bill_date
158    FROM billing b
159    JOIN patients p ON b.patient_id = p.patient_id
160    WHERE b.payment_status = 'Pending'
161      AND b.bill_date < DATE_SUB(CURDATE(), INTERVAL 30 DAY)
162    ORDER BY b.bill_date ASC;
163
164    -- 3. TREATMENT PROFITABILITY
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

first_name	last_name	contact_number	pending_amount	bill_date
Alex	Moore	7028910482	894.39	05-01-2023
Michael	Taylor	8019925828	3729.19	06-01-2023
Michael	Wilson	7923214041	4833.17	08-01-2023
Emily	Miller	8720989381	2593.43	11-01-2023
John	Taylor	9900972256	956.39	15-01-2023
Jane	Jones	6158428240	4652.41	17-01-2023
Emily	Smith	8228188767	3615.96	20-01-2023
Michael	Wilson	7765390555	3428.95	24-01-2023
Michael	Taylor	7223380592	2512.41	28-01-2023
Emily	Williams	7587653815	2761.55	03-02-2023
Laura	Davis	8135666049	4019.13	05-02-2023
David	Smith	8923607677	1882.80	06-02-2023
Sarah	Brown	7196777444	1565.92	12-02-2023
Linda	Moore	8724518272	806.78	13-02-2023
David	Smith	8923607677	857.39	14-02-2023
Jane	Smith	7040069008	2293.98	26-02-2023
Linda	Moore	8724518272	929.91	09-03-2023
Laura	Johnson	9059178882	3492.10	16-03-2023
Michael	Wilson	8545613046	4781.32	21-03-2023
Laura	Davis	7060324619	1158.68	21-03-2023
David	Williams	6939585183	975.49	01-04-2023

TREATMENT PROFITABILITY

```sql
-- TREATMENT PROFITABILITY

SELECT t.treatment_type,
COUNT(t.treatment_id) AS frequency,
SUM(b.amount) AS total_revenue,
ROUND(AVG(b.amount), 2) AS avg_revenue_per_treatment
FROM treatments t
JOIN billing b ON t.treatment_id = b.treatment_id
WHERE b.payment_status = 'Paid'
GROUP BY t.treatment_type
ORDER BY total_revenue DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

treatment_type	frequency	total_revenue	avg_revenue_per_treatment
X-Ray	17	47978.78	2822.28
MRI	14	43064.42	3076.03
Chemotherapy	13	32607.26	2508.25
Physiotherapy	11	32251.38	2931.94
ECG	9	17523.06	1947.01

Revenue Analysis by Treatment

Reviewing income streams from different treatments highlights the most profitable medical services offered.

Strategic Resource Allocation

Data-driven insights support better pricing, staff deployment, and equipment investment decisions.

Optimising Service Mix

Identifying high-margin procedures helps hospitals refine services to maximize overall profitability.

PATIENT & CLINICAL INSIGHTS

PATIENT RETENTION (LOYALTY)

Targeted Care Management

Frequent-visit patients with chronic conditions provide opportunities for personalized, targeted care management.

Resource Allocation

Identifying frequent patients helps hospitals allocate resources more effectively to improve care quality.

Proactive Engagement

Proactive patient engagement and continuity of care enhance long-term health outcomes for loyal patients.

```
188    -- PATIENT & CLINICAL INSIGHTS
189    -- Focuses on patient demographics and long-term care needs to improve clinical outcomes and marketing.
190
191    -- PATIENT RETENTION
192
193    SELECT p.patient_id,
194    CONCAT(p.first_name, ' ', p.last_name) AS patient_name,
195    COUNT(a.appointment_id) AS visit_count, p.email
196    FROM patients p
197    JOIN appointments a ON p.patient_id = a.patient_id
198    GROUP BY p.patient_id
199    HAVING visit_count > 3
200    ORDER BY visit_count DESC;
201
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

patient_id	patient_name	visit_count	email
P012	Laura Davis	10	laura.davis@mail.com
P005	David Wilson	8	david.wilson@mail.com
P016	Michael Taylor	7	michael.taylor@mail.com
P029	David Smith	7	david.smith@mail.com
P035	David Wilson	7	david.wilson@mail.com
P036	Michael Wilson	7	michael.wilson@mail.com
P049	David Moore	7	david.moore@mail.com
P010	Michael Taylor	6	michael.taylor@mail.com
P023	Linda Johnson	6	linda.johnson@mail.com
P026	John Taylor	6	john.taylor@mail.com
P037	Robert Williams	6	robert.williams@mail.com
P019	Sarah Miller	5	sarah.miller@mail.com
P025	Robert Wilson	5	robert.wilson@mail.com
P032	Alex Moore	5	alex.moore@mail.com

AGE GROUP DEMOGRAPHICS

Patient Age Segmentation

Patients are categorised into Pediatric, Adult, and Senior groups to better understand demographic needs and trends.

Resource and Staff Allocation

Segmented demographics guide effective allocation of staff and resources tailored to each age group's needs.

Strategic Clinical Planning

Age demographics analysis supports forecasting future healthcare demand and informs strategic clinical decisions.

```
202     -- AGE GROUP DEMOGRAPHICS
203
204 •   SELECT CASE
205     WHEN (YEAR(CURDATE()) - YEAR(date_of_birth)) < 18 THEN 'Pedi
206     WHEN (YEAR(CURDATE()) - YEAR(date_of_birth)) BETWEEN 18 AND
207     ELSE 'Senior (60+)' END AS age_category,
208     COUNT(*) AS patient_count
209     FROM patients
210     GROUP BY age_category;
211
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

age_category	patient_count
Senior (60+)	10
Adult (18-60)	40

GENDER DISTRIBUTION

Equitable Care and Staffing

Analysing gender distribution ensures fair staffing and equitable care in various medical specializations.

Aligning Services with Demographics

Gender insights help hospitals tailor services to meet diverse patient needs effectively.

Supporting Diversity and Inclusion

Gender-based analysis supports compliance with diversity and inclusion goals in healthcare.

```
223     -- GENDER DISTRIBUTION BY SPECIALISATION
224     -- Helps determine if certain departments (like OBGYN or Urology) if need more gender-specific medical staff.
225
226 •   SELECT d.specialization, p.gender,
227     COUNT(*) AS patient_count
228     FROM patients p
229     JOIN appointments a ON p.patient_id = a.patient_id
230     JOIN doctors d ON a.doctor_id = d.doctor_id
231     GROUP BY d.specialization, p.gender
232     ORDER BY d.specialization;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

specialization	gender	patient_count
Dermatology	F	23
Dermatology	M	47
Oncology	F	12
Oncology	M	20
Pediatrics	F	35
Pediatrics	M	63

STRATEGIC PLANNING & QUALITY CONTROL

DOCTOR–PATIENT RATIO

```
234     -- STRATEGIC PLANNING & QUALITY CONTROL
235     -- These queries focus on hospital growth, doctor-to-patient ratios, and potential revenue leakages.
236
237     -- DOCTOR-PATIENT RATIO BY SPECIALIZATION
238     -- Helps determine if you need to hire more doctors for a specific wing.
239     SELECT d.specialization,
240     COUNT(DISTINCT a.patient_id) AS unique_patients,
241     COUNT(DISTINCT d.doctor_id) AS total_doctors,
242     ROUND(COUNT(DISTINCT a.patient_id) / COUNT(DISTINCT d.doctor_id), 1) AS patients_per_doctor
243     FROM doctors d
244     JOIN appointments a ON d.doctor_id = a.doctor_id
245     GROUP BY d.specialization
246     ORDER BY patients_per_doctor DESC;
247
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

specialization	unique_patients	total_doctors	patients_per_doctor
Dermatology	37	3	12.3
Oncology	22	2	11.0
Pediatrics	47	5	9.4

Measuring Patient Load

Tracking unique patients per doctor by specialisation reveals workload and capacity pressures in healthcare settings.

Staffing Implications

High doctor–patient ratios may require additional hires or schedule adjustments to maintain care quality.

Quality Care and Burnout Prevention

Managing doctor–patient ratios helps prevent physician burnout and ensures consistent quality care.

UNBILLED TREATMENTS (REVENUE LEAKAGE)

Identifying Unbilled Treatments

Detecting treatments not billed is crucial for protecting hospital revenue and closing financial gaps.

Process Gaps in Billing Workflow

The audit query helps uncover inefficiencies in the treatment-to-billing workflow to prevent revenue loss.

Improving Profitability and Compliance

Addressing revenue leakage enhances hospital profitability and ensures regulatory compliance.

```
248     --  UNBILLED TREATMENTS
249     -- Finds treatments that were performed but do not have a corresponding bill.
250     -- This is critical for preventing revenue loss.
251     SELECT t.treatment_id, t.treatment_type, t.treatment_date, p.first_name, p.last_name
252     FROM treatments t
253     LEFT JOIN billing b ON t.treatment_id = b.treatment_id
254     JOIN appointments a ON t.appointment_id = a.appointment_id
255     JOIN patients p ON a.patient_id = p.patient_id
256     WHERE b.bill_id IS NULL;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

treatment_id	treatment_type	treatment_date	first_name	last_name

```sql
-- TREATMENT RECURRENCE
-- Tracks the average number of visits per patient for specific treatments.
SELECT t.treatment_type,
COUNT(t.treatment_id) AS total_procedures,
COUNT(DISTINCT a.patient_id) AS unique_patients,
ROUND(COUNT(t.treatment_id) / COUNT(DISTINCT a.patient_id), 1) AS visits_per_patient
FROM treatments t
JOIN appointments a ON t.appointment_id = a.appointment_id
GROUP BY t.treatment_type
ORDER BY visits_per_patient DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

treatment_type	total_procedures	unique_patients	visits_per_patient
MRI	36	22	1.6
Physiotherapy	36	23	1.6
Chemotherapy	49	32	1.5
X-Ray	41	28	1.5
ECG	38	31	1.2

TREATMENT RECURRENCE

Measuring Treatment Recurrence

Tracking repeated treatments per patient reveals patterns in clinical effectiveness and helps identify overuse or inefficiencies.

Implications of High Recurrence

High recurrence rates may indicate the need to review treatment protocols or consider alternative therapies for better outcomes.

Supporting Quality Improvement

Analyzing treatment recurrence data aids in quality improvement initiatives and supports long-term healthcare planning.