

## **Assignment #4 Part 2: Demo of Designing Views/ Simple Queries**

Arman Grewal(501100160), Gurpreet Singh Bhatti(501103368), Jasdeep Singh(501089933)

Faculty of Engineering, Toronto Metropolitan University

CPS 510: Database Systems I (Section 7, Group 8, Topic 33)

Jorge Lopez

October 16th, 2023

<b>Creating Views</b>	<b>3</b>
Comprehensive Appointment Information View with Patient, Dentist, and Medical Records Data	3
Invoice Summary View: Patient Information, Treatment Details, and Payment Status	4
Medical Record Summary View: Patient Information and Contact Details	5
Inventory and Treatment Information View: Item Details, Associated Treatments, and Costs	6
<b>Advanced Queries with Join Functions</b>	<b>7</b>
Invoice and Treatment Details: Patient Information, Treatment Costs, and Payment Status	7
Patient Unpaid Invoices Summary: Tracking Pending Payments by Patient and Insurance ID	8
Dentist Appointments by Day of Week: Counting Weekly Appointments and Dentist Assignments	9

# Creating Views

## Comprehensive Appointment Information View with Patient, Dentist, and Medical Records Data

```
CREATE VIEW appointment_info AS
SELECT
    a.appointment_id,
    a.description AS appointment_description,
    a.datetime AS appointment_datetime,
    p.name AS patient_name,
    d.name AS dentist_name,
    m.medical_file_id
FROM appointments a
JOIN patients p ON a.patient_id = p.patient_id
JOIN dentists d ON a.dentist_id = d.dentist_id
LEFT JOIN medical_records m ON p.patient_id = m.patient_id;
```

	APPOINTMENT_ID	APPOINTMENT_DESCRIPTION	APPOINTMENT_DATETIME	PATIENT_NAME	DENTIST_NAME	MEDICAL_FILE_ID
1	1	Routine Checkup	23-10-05	John Doe	Dr. John Smith	916577
2	2	Orthodontic Consultation	23-10-07	Jane Smith	Dr. Sarah Jones	117588
3	3	Periodontal Treatment	23-10-10	Bob Johnson	Dr. Michael Wilson	985116
4	4	Pediatric Checkup	23-10-12	Lisa Miller	Dr. Jennifer Davis	872205
5	5	Oral Surgery Consultation	23-10-15	Michael Wilson	Dr. David Brown	828194
6	6	Root Canal Procedure	23-10-17	Susan Brown	Dr. Laura Evans	403188
7	7	Denture Fitting	23-10-20	David Jones	Dr. Joseph Martin	485955
8	8	Orthodontic Adjustment	23-10-22	Sarah Jackson	Dr. Emily Thompson	291069
9	9	Routine Checkup	23-10-25	Chris White	Dr. Mark Harris	615852
10	10	Periodontal Surgery	23-10-27	Ella Davis	Dr. Linda Wilson	423866
11	50	Periodontal Surgery	24-02-21	Peter Thompson	Dr. Linda Wilson	404663
12	49	Routine Checkup	24-02-18	Laura Martin	Dr. Mark Harris	242850
13	48	Orthodontic Adjustment	24-02-15	James Wilson	Dr. Emily Thompson	256632
14	47	Denture Fitting	24-02-12	Mary Brown	Dr. Joseph Martin	910842
15	46	Root Canal Procedure	24-02-09	Matthew Evans	Dr. Laura Evans	390435
16	45	Oral Surgery Consultation	24-02-06	Linda Johnson	Dr. David Brown	652301
17	44	Pediatric Checkup	24-02-03	Andrew Smith	Dr. Jennifer Davis	683400
18	43	Periodontal Treatment	24-01-31	Jennifer Davis	Dr. Michael Wilson	106240
19	42	Orthodontic Consultation	24-01-28	Mark Jones	Dr. Sarah Jones	882194
20	41	Routine Checkup	24-01-25	Susan Wilson	Dr. John Smith	524486
21	40	Root Canal Procedure	24-01-22	Daniel Miller	Dr. Linda Wilson	145140

The SQL code above creates a view called "appointment\_info." This view combines information from several tables, this includes appointments, patients, dentists, and medical records. It will select and rename specific columns, such as the appointment description, patient name, datetime, dentist name, and medical file ID, and then performs joins and left joins to gather important data into a single view. This view is designed to provide a thorough overview of appointment details along with associated patient and dentist information, as well as medical records where available.

## Invoice Summary View: Patient Information, Treatment Details, and Payment Status

```
CREATE VIEW invoice_summary AS
SELECT
    i.invoice_id,
    p.name AS patient_name,
    t.name_of_treatment,
    i.payment_status,
    i.payment_type,
    i.cost
FROM invoices i
JOIN patients p ON i.patient_id = p.patient_id
JOIN treatments t ON i.treatment_id = t.treatment_id;
```

	INVOICE_ID	PATIENT_NAME	NAME_OF_TREATMENT	PAYMENT_STATUS	PAYMENT_TYPE	COST
1	10	Michael Wilson	Oral Surgery Consultation	Paid	Credit Card	450
2	3	Michael Wilson	Orthodontic Consultation	Pending	Insurance	350
3	4	David Jones	Root Canal	Paid	Credit Card	600
4	11	Peter Thompson	Root Canal	Paid	Credit Card	600
5	19	Mary Brown	Dental Cleaning	Pending	Insurance	200
6	15	Matthew Evans	Orthodontic Consultation	Paid	Cash	350
7	9	Jennifer Davis	Dental Implant	Pending	Insurance	550
8	17	Mark Jones	Periodontal Treatment	Paid	Credit Card	400
9	13	Nancy Johnson	Oral Surgery Consultation	Pending	Insurance	450
10	2	Emily Johnson	Tooth Extraction	Paid	Cash	500
11	7	Ella Davis	Periodontal Treatment	Paid	Credit Card	400
12	18	Michael Davis	Dental Implant	Paid	Cash	550
13	14	Sarah Wilson	Dental Cleaning	Paid	Credit Card	200
14	1	David Brown	Dental Cleaning	Paid	Credit Card	200
15	12	Joseph Garcia	Dental Filling	Paid	Cash	300
16	20	Nancy Allen	Dental Cleaning	Paid	Credit Card	200
17	16	Christopher King	Dental Crown	Pending	Insurance	250
18	8	Robertttt Davis	Dental Filling	Paid	Cash	300
19	6	Robertttt Davis	Teeth Whitening	Pending	Insurance	700
20	5	Lauraaaa Johnson	Dental Crown	Paid	Cash	250

The SQL code above creates a view named "invoice\_summary." This view gathers information from the "invoices," "patients," and "treatments" tables. It selects specific columns such as invoice ID, treatment name, patient name, payment status, payment type, and cost. By using joins, it connects invoices to the respective patients and treatments, offering a simple summary of invoice details, including the associated patient, treatment, cost, and payment information.

## Medical Record Summary View: Patient Information and Contact Details

```
CREATE VIEW medical_record_summary AS
SELECT
  m.medical_file_id,
  p.name AS patient_name,
  p.date_of_birth,
  p.email_address AS patient_email,
  p.phone_number AS patient_phone
FROM medical_records m
JOIN patients p ON m.patient_id = p.patient_id;
```

	⚡ MEDICAL_FILE_ID	⚡ PATIENT_NAME	⚡ DATE_OF_BIRTH	⚡ PATIENT_EMAIL	⚡ PATIENT_PHONE	
1	916577	John Doe	90-01-15	john.doe@example.com	555-555-5555	
2	117588	Jane Smith	85-03-20	jane.smith@example.com	555-555-5556	
3	985116	Bob Johnson	95-07-10	bob.johnson@example.com	555-555-5557	
4	872205	Lisa Miller	82-12-05	lisa.miller@example.com	555-555-5558	
5	828194	Michael Wilson	78-08-25	michael.wilson@example.com	555-555-5559	
6	403188	Susan Brown	92-04-30	susan.brown@example.com	555-555-5560	
7	485955	David Jones	87-06-18	david.jones@example.com	555-555-5561	
8	291069	Sarah Jackson	76-11-12	sarah.jackson@example.com	555-555-5562	
9	615852	Chris White	98-03-02	chris.white@example.com	555-555-5563	
10	423866	Ella Davis	95-12-10	patient50@example.com	555-555-5610	
11	404663	Peter Thompson	75-05-22	peter.thompson@example.com	555-555-5565	
12	242850	Laura Martin	94-01-19	laura.martin@example.com	555-555-5566	
13	256632	James Wilson	83-07-14	james.wilson@example.com	555-555-5567	
14	910842	Mary Brown	97-11-29	mary.brown@example.com	555-555-5568	
15	390435	Matthew Evans	79-04-01	matthew.evans@example.com	555-555-5569	
16	652301	Linda Johnson	91-08-11	linda.johnson@example.com	555-555-5570	
17	683400	Andrew Smith	86-12-26	andrew.smith@example.com	555-555-5571	
18	106240	Jennifer Davis	80-02-03	jennifer.davis@example.com	555-555-5572	
19	882194	Mark Jones	96-06-29	mark.jones@example.com	555-555-5573	
20	524486	Susan Wilson	84-10-16	susan.wilson@example.com	555-555-5574	
21	145140	Daniel Miller	77-03-07	daniel.miller@example.com	555-555-5575	
22	299465	Karen Brown	93-09-03	karen.brown@example.com	555-555-5576	
23	500955	Robert Taylor	81-01-10	robert.taylor@example.com	555-555-5577	
24	922901	Nancy Johnson	99-05-14	nancy.johnson@example.com	555-555-5578	
25	878980	William Davis	88-07-23	william.davis@example.com	555-555-5579	

The SQL code above creates a view called "medical\_record\_summary" that compiles important patient information from the "medical\_records" and "patients" tables. It selects columns including the medical file ID, patient name, date of birth, email address, and phone number. By employing a join, it links medical records to the corresponding patient records, providing a concise summary of key patient details, which can be valuable for medical record management and analysis.

### Inventory and Treatment Information View: Item Details, Associated Treatments, and Costs

```
CREATE VIEW inventory_treatment_info AS
SELECT
    i.inventory_id,
    i.name_of_item,
    t.name_of_treatment,
    t.cost AS treatment_cost
FROM inventory i
LEFT JOIN treatments t ON i.inventory_id = t.inventory_id;
```

	INVENTORY_ID	NAME_OF_ITEM	NAME_OF_TREATMENT	TREATMENT_COST
1	1	Dental Chair	Dental Implant	550
2	2	Dental Instruments Set	Orthodontic Consultation	350
3	3	X-Ray Machine	Periodontal Treatment	400
4	4	Dental Supplies Kit	Dental Cleaning	200
5	5	Dental Chair Upholstery	(null)	(null)
6	6	Sterilization Equipment	Dental Crown	250
7	7	Dental Imaging Software	(null)	(null)
8	8	Dental Masks	Dental Filling	300
9	9	Dental Handpieces	Tooth Extraction	500
10	10	Dental Chairside Table	(null)	(null)
11	11	Dental X-Ray Film	(null)	(null)
12	12	Dental Curing Light	(null)	(null)
13	13	Dental Loupes	Oral Surgery Consultation	450
14	14	Dental Autoclave	Teeth Whitening	700
15	15	Dental Impression Trays	(null)	(null)
16	16	Dental Nitrous Oxide System	(null)	(null)
17	17	Dental Amalgamator	(null)	(null)
18	18	Dental Prophylaxis Angles	(null)	(null)
19	19	Dental Apex Locator	Root Canal	600
20	20	Dental Ultrasonic Scaler	(null)	(null)

This SQL code creates a view named "inventory\_treatment\_info" that combines information about inventory items and their associated treatments. It selects columns like inventory ID, item name, treatment name, and treatment cost. A left join is utilized, it combines data from the "inventory" and "treatments" tables, providing a concise summary of inventory items and their linked treatments, including cost information, which can be valuable for managing inventory and pricing treatment options.

# Advanced Queries with Join Functions

## Invoice and Treatment Details: Patient Information, Treatment Costs, and Payment Status

```
SELECT
  i.invoice_id,
  p.name AS patient_name,
  p.insurance_id,
  t.name_of_treatment,
  i.cost AS treatment_cost,
  i.payment_status,
  i.payment_type
FROM invoices i
JOIN patients p ON i.patient_id = p.patient_id
JOIN treatments t ON i.treatment_id = t.treatment_id;
```

	INVOICE_ID	PATIENT_NAME	INSURANCE_ID	NAME_OF_TREATMENT	TREATMENT_COST	PAYMENT_STATUS	PAYMENT_TYPE
1	10	Michael Wilson	5432109874	Oral Surgery Consultation	450	Paid	Credit Card
2	3	Michael Wilson	5432109874	Orthodontic Consultation	350	Pending	Insurance
3	4	David Jones	3210987656	Root Canal	600	Paid	Credit Card
4	11	Peter Thompson	9988776655	Root Canal	600	Paid	Credit Card
5	19	Mary Brown	6655443322	Dental Cleaning	200	Pending	Insurance
6	15	Matthew Evans	5544332211	Orthodontic Consultation	350	Paid	Cash
7	9	Jennifer Davis	3333444555	Dental Implant	550	Pending	Insurance
8	17	Mark Jones	2222333444	Periodontal Treatment	400	Paid	Credit Card
9	13	Nancy Johnson	4567456745	Oral Surgery Consultation	450	Pending	Insurance
10	2	Emily Johnson	260027	Tooth Extraction	500	Paid	Cash
11	7	Ella Davis	5674548567856	Periodontal Treatment	400	Paid	Credit Card
12	18	Michael Davis	260028	Dental Implant	550	Paid	Cash
13	14	Sarah Wilson	260029	Dental Cleaning	200	Paid	Credit Card
14	1	David Brown	260030	Dental Cleaning	200	Paid	Credit Card
15	12	Joseph Garcia	260036	Dental Filling	300	Paid	Cash
16	20	Nancy Allen	260039	Dental Cleaning	200	Paid	Credit Card
17	16	Christopher King	260040	Dental Crown	250	Pending	Insurance
18	8	Roberttttt Davis	234523314523	Dental Filling	300	Paid	Cash
19	6	Roberttttt Davis	234523314523	Teeth Whitening	700	Pending	Insurance

The SQL query above selects specific columns from the "invoices" table and joins them with relevant data from the "patients" and "treatments" tables. It will retrieve information such as invoice ID, insurance ID, patient name, treatment name, treatment cost, payment status, and payment type. This will also generate a summary of invoices, including details about the associated patient, treatment, and billing information.

### Patient Unpaid Invoices Summary: Tracking Pending Payments by Patient and Insurance ID

```
SELECT
  p.name AS patient_name,
  p.insurance_id,
  SUM(CASE WHEN i.payment_status = 'Pending' THEN i.cost ELSE 0 END) AS
total_unpaid_invoices
FROM patients p
LEFT JOIN invoices i ON p.patient_id = i.patient_id
LEFT JOIN treatments t ON i.treatment_id = t.treatment_id
GROUP BY p.name, p.insurance_id;
```

	PATIENT_NAME	INSURANCE_ID	TOTAL_UNPAID_INVOICES
1	Bob Johnson	7654321092	0
2	Susan Brown	4321098765	0
3	Susan Wilson	1111222333	0
4	Jennifer Lee	260031	0
5	Margaret Rodriguez	260041	0
6	Jenniferrrr Martin	123417823412	0
7	Emilyyyyy Wilson	3456354645634	0
8	Michael Wilson	5432109874	350
9	Laura Martin	8877665544	0
10	Matthew Evans	5544332211	0
11	Roberttttt Davis	234523314523	700
12	John Doe	9876543210	0
13	Sarah Jackson	2109876547	0
14	Peter Thompson	9988776655	0
15	Karen Brown	2345234523	0
16	Robert Taylor	3456345634	0
17	Patricia Lewis	260043	0
18	Lisa Miller	6543210983	0
19	David Jones	3210987656	0

The SQL query above calculates the total unpaid invoices for each patient with details on their name and insurance ID. It achieves this by joining the "patients," "invoices," and "treatments" tables and then grouping the results by patient name and insurance ID. The "SUM" function is used to add up the costs of invoices with a 'Pending' payment status, providing a summary of payments for each patient.



## Dentist Appointments by Day of Week: Counting Weekly Appointments and Dentist

### Assignments

```
SELECT
  TO_CHAR(a.datetime, 'DY') AS day_of_week,
  COUNT(*) AS number_of_appointments,
  d.name AS dentist_name
FROM appointments a
JOIN patients p ON a.patient_id = p.patient_id
JOIN dentists d ON a.dentist_id = d.dentist_id
GROUP BY TO_CHAR(a.datetime, 'DY'), d.name
ORDER BY TO_CHAR(a.datetime, 'DY');
```

DAY_OF_WEEK	NUMBER_OF_APPOINTMENTS	DENTIST_NAME
1 FRI	2	Dr. David Brown
2 FRI	1	Dr. Joseph Martin
3 FRI	1	Dr. Laura Evans
4 FRI	1	Dr. Linda Wilson
5 FRI	1	Dr. Mark Harris
6 FRI	1	Dr. Sarah Jones
7 MON	1	Dr. John Smith
8 MON	1	Dr. Joseph Martin
9 MON	2	Dr. Laura Evans
10 MON	1	Dr. Linda Wilson
11 MON	1	Dr. Mark Harris
12 MON	1	Dr. Michael Wilson
13 SAT	1	Dr. Emily Thompson
14 SAT	1	Dr. Jennifer Davis
15 SAT	1	Dr. Joseph Martin
16 SAT	1	Dr. Linda Wilson
17 SAT	1	Dr. Michael Wilson
18 SAT	1	Dr. Sarah Jones
19 SUN	2	Dr. David Brown

The SQL query above retrieves data about the number of appointments scheduled for each day of the week, grouped by the day of the week and dentist's name. A function called "TO\_CHAR" is utilized to specifically extract the day of the week from the appointment datetime. The results are then sorted in ascending order based on the day of the week.