

DBST 652: Patient Management Database

Student 1

Student 2

Student 3

Date

## Table of Contents

Deliverable 1 – Plan of Work .....	2
Deliverable 2 – Entity Relationship Diagram .....	4
Deliverable 3 – Data Definition Language .....	6
Deliverable 4 – Data Manipulation Language .....	11
Deliverable 5 – SQL Queries .....	22
Deliverable 6 – Database Maintenance Scripts .....	27
Deliverable 7 – Stored Procedure .....	30
Deliverable 8 – Explain Tool .....	32
Deliverable 9 – Object-Relational Exercises .....	35
References .....	37

1.	POW Project Work Plan	<ul style="list-style-type: none"> <li>Clearly state project description, purpose and objective.</li> <li>Present a timeline for the project</li> </ul>
----	-----------------------------	---

Database: Doctor's Office Patient Management Database

Database Objective: The objective of this project is to create a database that will store information for a doctor's office, in order to help its leaders manage their patient population, their employee workload, and their billing. The database will contain specific patient information, including name, Social Security Number, date of birth, address, other biographical data, and insurance information. The database will also store the names of health care professionals and a history of the patients whom they see and the procedures they perform. Finally, the database will store billing and bill payment information, to facilitate reimbursement from patients and their insurance.

The patient, provider, and procedure information are currently managed entirely on paper. Every patient has a paper folder that contains his or her contact, biographical, and insurance information, as well as a history of his or her appointments, including who saw them and what procedures were performed. Each health care provider has a scheduling book into which the schedulers record appointments. Billing is outsourced to a third party, to whom billing data are transmitted via manual entry in a web-based interface—but the implementation of this database would permit the doctor's office to manage its billing in-house.

Benefits: This database will help office managers to automate many processes pertaining to patient management, including mailings, billings, and scheduling of both patients and health care professionals, such as doctors and nurses. It will also enable the health care professionals to electronically log the procedures they perform for a given patient, which will facilitate

automation of billing to insurance carriers and patients. It could also potentially enable patients to manage their own accounts and payments online.

Analyses of the data collected in such a system would permit the office management to see which procedures are performed most frequently by which providers, and in which combinations they are performed. These data could be used to identify waste from unnecessary procedures, or gaps in staff knowledge. The management could use biographical information of existing patients to refine their marketing toward potential new patients. They could use insurance and reimbursement information to negotiate better reimbursement rates from insurers. Finally, they could use visit information to balance the workload of the healthcare professionals.

Technical Details: The database to be used is Oracle 10g, residing on a Unix server and accessed via a PuTTY terminal emulator using a Secure Shell (SSH) network protocol. The Data Definition Language (DDL) and Data Manipulation Language (DML) subsets of controls belonging to Structured Query Language (SQL) for Oracle will be utilized to create and populate the tables, respectively. Stored procedures will be written using PL/SQL and executed using SQL\*Plus.

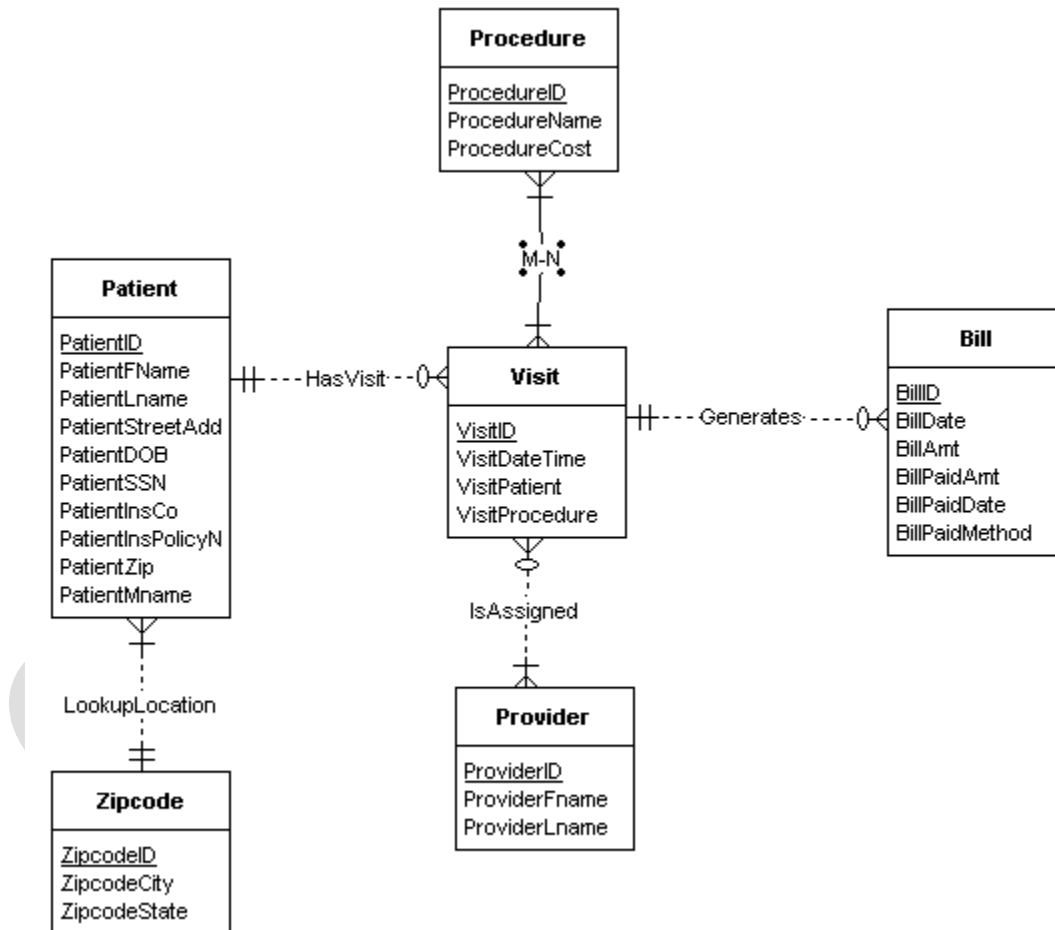
#### Project Timeline by Deliverable

1. Develop Plan of Work (POW): 9/8/10-9/14/10
2. Develop Entity Relationship Diagram (ERD): 9/15/10-9/21/10
3. Translate ERD into DDL, run DDL, and start loading data: 9/22/10-9/28/10
4. Finish data load and control files, and document query results: 9/29/10-10/5/10
5. Run 5 simple and 7 advanced SQL queries and document results: 10/6/10-10/12/10
6. Develop scripts for performing basic maintenance tasks: 10/13/10-10/26/10 (2 weeks)
7. Create a stored procedure and test performance using the explain tool: 10/27/10-11/2/10
8. Continue testing of explain tool and show CPU execution times: 11/3/10-11/9/10
9. Conduct object relational exercises and complete explain tool testing: 11/10/10-11/16/10
10. Complete and submit lab project report: 11/17/10-11/23/10

2.	ERD Entity Relationship Diagram	<ul style="list-style-type: none"> <li>• Use ER-Assistant tool.</li> <li>• Include a M-N relationship</li> <li>• Obtain clean compilation</li> <li>• Show PKs and attributes</li> </ul>	<ul style="list-style-type: none"> <li>• Limit to 5-6 entities</li> <li>• Resolve M-N relationships (resulting tables count toward your 5-6 entities)</li> </ul>
----	--	---	--

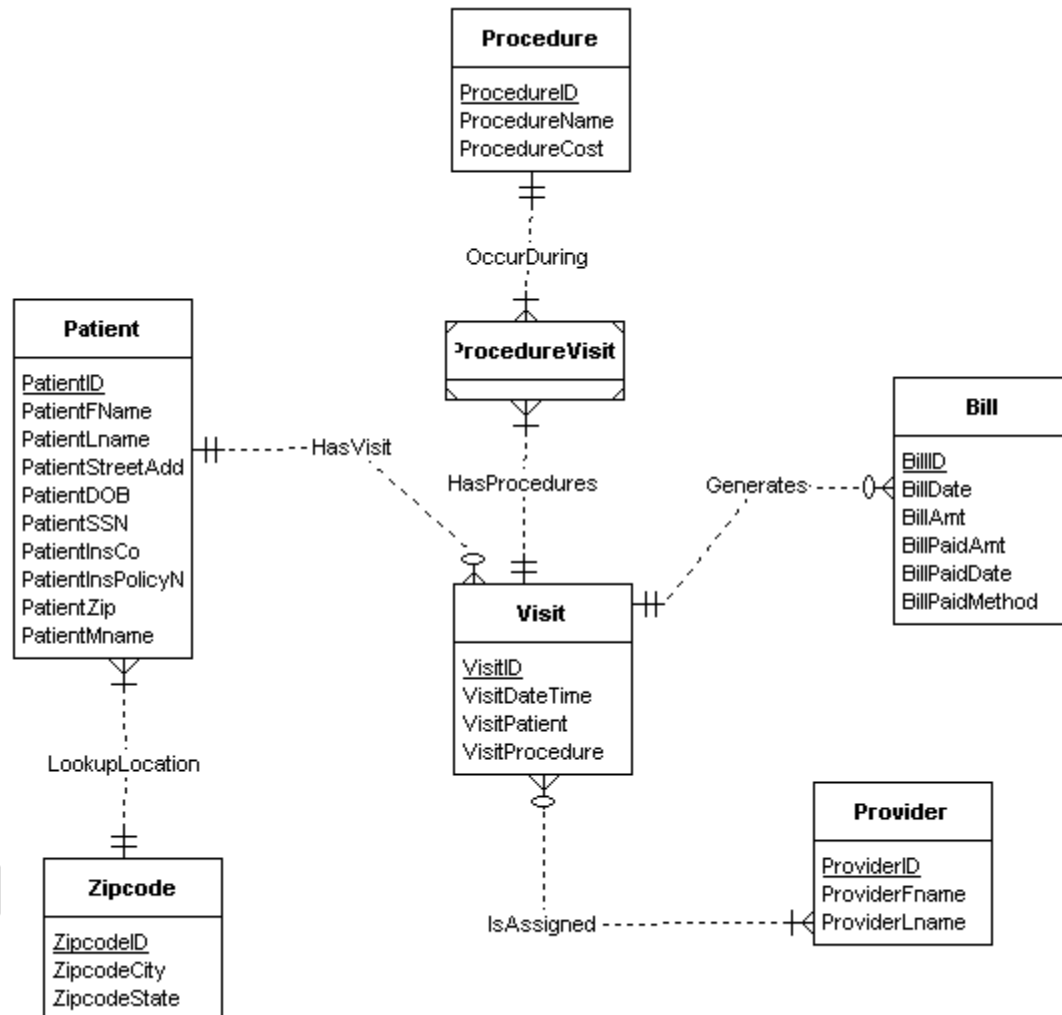
The first ERD shows an unresolved M-N relationship between the Visit and Procedure entities.

Each visit has multiple procedures, and each procedure is performed at different visits.



## Resolved M-N relationships

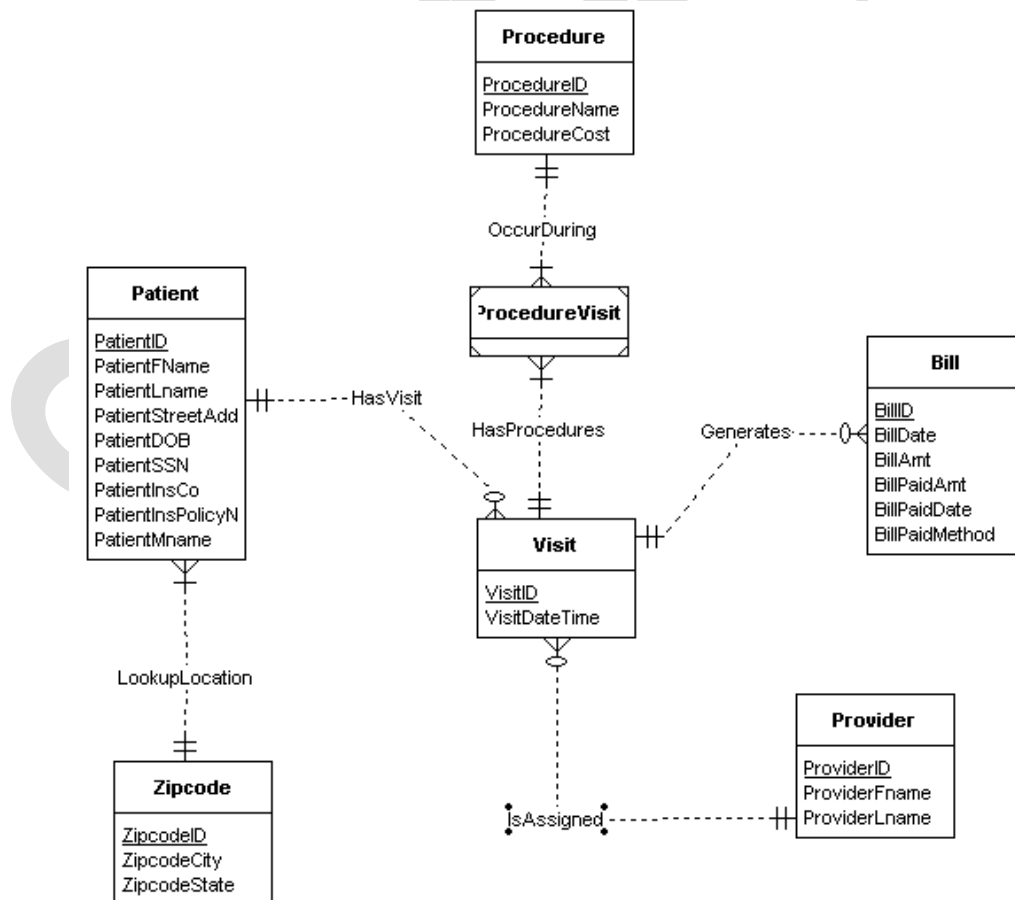
The second ERD shows a resolved M-N relationship between Procedures and Visits, transformed into 1-M relationships with the resulting entity ProcedureVisit.



3.	DDL Data Definition Language	<ul style="list-style-type: none"> <li>• Translate ERD into DDL</li> <li>• Make changes to POW, ERD and DDL as your project changes/develops.</li> <li>• Include query results from catalog in order to verify your db objects.</li> <li>• Include code and output/log</li> </ul>
----	---------------------------------------	---

### Changes to ERD

- VisitPatient and VisitProcedure were removed from the Visit type and PatientZip was removed from the Patient type, since all of those attributes are foreign keys and hence do not belong in the ERD.
- The relationship between visit and provider was modified so that there must be only one provider per visit. This was to simplify the structure.



**Translate ERD into DDL - Code**

```
drop table bill;
drop table procedurevisit;
drop table visit;
drop table procedure;
drop table provider;
drop table patient;
drop table zipcode2;
```

```
Create table Zipcode2 (
ZipcodeID    number(5,0) not null,
ZipcodeCity  Varchar2(35),
ZipcodeState varchar2(30),
constraint pk_zipcode primary key (zipcodeid)
);
```

```
Create table Patient (
PatientID number(9,0) not null,
PatientFname varchar2(20),
PatientLname varchar2(20),
PatientMname varchar2(20),
PatientSSN number(9,0),
PatientDob Date,
Patientstreetaddr varchar2(20),
PatientZipID number(5,0),
Patientinsco varchar2(25),
Patientinspolycynum number(10,0),
constraint pk_patient primary key (patientid),
constraint fk_zipcode foreign key (patientzipid) references
zipcode2(zipcodeid)
);
```

```
Create table Provider (
ProviderID Number(9,0) not null,
ProviderFname Varchar2(20),
ProviderLname varchar2(20),
constraint pk_provider primary key (providerid)
);
```

```
Create table Procedure (
ProcedureID number(9,0) not null,
Procedurename varchar2(35),
ProcedureCost number(8,2),
constraint pk_procedure2 primary key (procedureid)
);
```

```
Create table Visit (
VisitID number(5,0) not null,
VisitDateTime Date,
PatientID number(9,0),
ProviderID Number(9,0),
constraint pk_visit primary key (visitid),
constraint fk_patient foreign key (patientid) references patient (patientid),
constraint fk_provider foreign key (patientid) references provider
(providerid)
```



```

);

create table ProcedureVisit (
VisitID number (5,0) not null,
ProcedureID number (9,0),
constraint pk_procedure_visit primary key (visitid, procedureid),
constraint fk_visitid foreign key (visitid) references visit (visitid),
constraint fk_procedureid foreign key (procedureid) references procedure
(procedureid)
);

Create table Bill (
BillID number(9,0) not null,
BillDate Date,
BillAmt Decimal(8,2),
BillpaidAmt Decimal(8,2),
BillpaidDate Date,
BillpaidMethod varchar2(8),
VisitID number (5,0),
constraint pk_bill primary key (billid),
constraint fk_visit foreign key (visitid) references visit (visitid)
);

```

## Output/Log

Objects created on Oracle 10.2 instance other than dbcourse2 because dbcourse2 response time was too slow (as were many UMUC servers, including WebTycho) at the time this was completed.

```

drop table bill;
WARNING: 22:23:37 [DROP - 0 row(s), 0.031 secs] Command processed. No rows
were affected
drop table procedurevisit;
WARNING: 22:23:37 [DROP - 0 row(s), 0.032 secs] Command processed. No rows
were affected
drop table visit;
WARNING: 22:23:37 [DROP - 0 row(s), 0.031 secs] Command processed. No rows
were affected
drop table procedure;
WARNING: 22:23:37 [DROP - 0 row(s), 0.031 secs] Command processed. No rows
were affected
drop table provider;
WARNING: 22:23:37 [DROP - 0 row(s), 0.032 secs] Command processed. No rows
were affected
drop table patient;
WARNING: 22:23:37 [DROP - 0 row(s), 0.031 secs] Command processed. No rows
were affected
drop table zipcode2;
WARNING: 22:23:37 [DROP - 0 row(s), 0.062 secs] Command processed. No rows
were affected
Create table Zipcode2 (
ZipcodeID number(5,0) not null,

```

```

ZipcodeCity Varchar2(35),
ZipcodeState varchar2(30),
constraint pk_zipcode primary key (zipcodeid)
);
WARNING: 22:23:37 [CREATE - 0 row(s), 0.047 secs] Command processed. No rows
were affected
Create table Patient (
PatientID number(9,0) not null,
PatientFname varchar2(20),
PatientLname varchar2(20),
PatientMname varchar2(20),
PatientSSN number(9,0),
PatientDob Date,
Patientstreetaddr varchar2(20),
PatientZipID number(5,0),
Patientinsco varchar2(25),
Patientinspolicynum number(10,0),
constraint pk_patient primary key (patientid),
constraint fk_zipcode foreign key (patientzipid) references
zipcode2(zipcodeid)
);
WARNING: 22:23:37 [CREATE - 0 row(s), 0.032 secs] Command processed. No rows
were affected
Create table Provider (
ProviderID Number(9,0) not null,
ProviderFname Varchar2(20),
ProviderLname varchar2(20),
constraint pk_provider primary key (providerid)
);
WARNING: 22:23:37 [CREATE - 0 row(s), 0.031 secs] Command processed. No rows
were affected
Create table Procedure (
ProcedureID number(9,0) not null,
Procedurename varchar2(35),
ProcedureCost number(8,2),
constraint pk_procedure2 primary key (procedureid)
);
WARNING: 22:23:37 [CREATE - 0 row(s), 0.031 secs] Command processed. No rows
were affected
Create table Visit (
VisitID number(5,0) not null,
VisitDateTime Date,
PatientID number(9,0),
ProviderID Number(9,0),
constraint pk_visit primary key (visitid),
constraint fk_patient foreign key (patientid) references patient (patientid),
constraint fk_provider foreign key (patientid) references provider
(providerid)
);
WARNING: 22:23:37 [CREATE - 0 row(s), 0.015 secs] Command processed. No rows
were affected
create table ProcedureVisit (
VisitID number (5,0) not null,
ProcedureID number (9,0),
constraint pk_procedure_visit primary key (visitid, procedureid),
constraint fk_visitid foreign key (visitid) references visit (visitid),

```

```

constraint fk_procedureid foreign key (procedureid) references procedure
(procedureid)
);
WARNING: 22:23:37 [CREATE - 0 row(s), 0.016 secs] Command processed. No rows
were affected
Create table Bill (
BillID number(9,0) not null,
BillDate Date,
BillAmt Decimal(8,2),
BillpaidAmt Decimal(8,2),
BillpaidDate Date,
BillpaidMethod varchar2(8),
VisitID number (5,0),
constraint pk_bill primary key (billid),
constraint fk_visit foreign key (visitid) references visit (visitid)
);
WARNING: 22:23:37 [CREATE - 0 row(s), 0.016 secs] Command processed. No rows
were affected
SUMMARY: ... 14 statement(s) executed, 0 row(s) affected, exec/fetch time:
0.438/0.000 sec
[0 successful, 14 warnings, 0 errors]

```

### Query results from Catalog

OBJECT_NAME	OBJECT_TYPE	CREATED	STATUS
PK_BILL	INDEX	2010-09-28 21:57:30	VALID
PK_PATIENT	INDEX	2010-09-28 21:57:03	VALID
PK_PROCEDURE2	INDEX	2010-09-28 21:57:26	VALID
PK_PROCEDURE_VISIT	INDEX	2010-09-28 21:57:28	VALID
PK_PROVIDER	INDEX	2010-09-28 21:57:12	VALID
PK_VISIT	INDEX	2010-09-28 21:57:27	VALID
PK_ZIPCODE	INDEX	2010-09-28 21:56:40	VALID
BILL	TABLE	2010-09-28 21:57:29	VALID
PATIENT	TABLE	2010-09-28 21:56:42	VALID
PROCEDURE	TABLE	2010-09-28 21:57:14	VALID
PROCEDUREVISIT	TABLE	2010-09-28 21:57:28	VALID
PROVIDER	TABLE	2010-09-28 21:57:04	VALID
VISIT	TABLE	2010-09-28 21:57:27	VALID
ZIPCODE2	TABLE	2010-09-28 21:56:40	VALID

4.	DML Data Manipulation Language	<ul style="list-style-type: none"> <li>• Use INSERT command to load some data</li> <li>• Total number of records should be 100-150 (all tables combined)</li> <li>• Include code/control files and output/log</li> </ul>
----	---	--

### DML – Insert Command code

```

delete from bill;
delete from procedurevisit;
delete from visit;
delete from procedure;
delete from provider;
delete from patient;
delete from zipcode2;

insert into zipcode2 values (20790,'Capitol Heights','MD');
insert into zipcode2 values (20783,'Hyattsville','MD');
insert into zipcode2 values (20792,'Upper Marlboro','MD');
insert into zipcode2 values (20794,'Jessup','MD');
insert into zipcode2 values (20742,'College Park','MD');

insert into patient values
(1,'Robert','Smith','H',123456789,to_date('19731001','YYYYMMDD'),'123 Main
Street',20790,'Aetna',123);
insert into patient values
(2,'John','Ortiz','A',142378123,to_date('19560312','YYYYMMDD'),'2342 Wayne
Ave',20790,'Aetna',534);
insert into patient values
(3,'Mike','Valencia','B',284756246,to_date('19600515','YYYYMMDD'),'1333
Maple',20790,'Blue Cross',106);
insert into patient values
(4,'Tim','Cruz','C',569512492,to_date('19650801','YYYYMMDD'),'2343
Pine',20790,'United',212);
insert into patient values
(5,'Shawn','Grageda','D',113902498,to_date('19821230','YYYYMMDD'),'983
Oak',20790,'Aetna',424);
insert into patient values
(6,'Dave','Brazoban','E',227804996,to_date('19760707','YYYYMMDD'),'1328
Willow',20783,'Blue Cross',848);
insert into patient values
(7,'Chris','Rivas','F',455609992,to_date('19840202','YYYYMMDD'),'2433
Walnut',20783,'United',169);
insert into patient values
(8,'Steve','Wilson','G',911219984,to_date('19430101','YYYYMMDD'),'111
Redwood',20783,'Aetna',338);
insert into patient values
(9,'Harry','Henderson','H',182243996,to_date('19670205','YYYYMMDD'),'9873
Cherry',20783,'Blue Cross',676);
insert into patient values
(10,'Doug','Feith','I',364487992,to_date('19681109','YYYYMMDD'),'8585
Holly',20783,'United',135);

```

```

insert into patient values
(11,'Dan','Snyder','J',728975984,to_date('19690911','YYYYMMDD'),'2433
Almond',20792,'Aetna',270);
insert into patient values
(12,'Fred','Ortiz','K',145795196,to_date('19701012','YYYYMMDD'),'1724
Pomegranate',20792,'Blue Cross',540);
insert into patient values
(13,'Larry','Valencia','L',291590392,to_date('19711113','YYYYMMDD'),'973 1st
St',20792,'United',108);
insert into patient values
(14,'Chuck','Cruz','M',583180784,to_date('19721214','YYYYMMDD'),'234 2nd
Ave',20792,'Aetna',216);
insert into patient values
(15,'Pat','Grageda','N',116636156,to_date('19730115','YYYYMMDD'),'2432 3rd
St',20794,'Blue Cross',432);
insert into patient values
(16,'Bob','Brazoban','O',233272312,to_date('19740216','YYYYMMDD'),'6534 4th
Ave',20794,'United',864);
insert into patient values
(17,'Bill','Rivas','P',466544624,to_date('19750317','YYYYMMDD'),'9083 5th
St',20794,'Aetna',172);
insert into patient values
(18,'Lenny','Wilson','Q',933089248,to_date('19760418','YYYYMMDD'),'23 7th
Ave',20794,'Blue Cross',344);
insert into patient values
(19,'Carl','Henderson','R',186617849,to_date('19770519','YYYYMMDD'),'9087 8th
St',20742,'United',688);
insert into patient values
(20,'Julio','Feith','S',373235698,to_date('19780629','YYYYMMDD'),'232 9th
St',20742,'Aetna',137);

```

```

INSERT INTO PROVIDER VALUES (301,'Laurie','Blain');
INSERT INTO provider VALUES (302,'Joseph','Sinyan');
INSERT INTO provider VALUES (300,'Darren','Schneider');

```

```

INSERT INTO PROCEDURE VALUES (6,'New Patient History',100);
INSERT INTO PROCEDURE VALUES (7,'Rapid Strep Test',50);
INSERT INTO PROCEDURE VALUES (8,'Injury Evaluation',100);
INSERT INTO PROCEDURE VALUES (9,'Well Check',100);
INSERT INTO PROCEDURE VALUES (10,'Follow Up',50);
insert into procedure values (1,'Colonoscopy',213);
insert into procedure values (2,'Chest x-ray',243);
insert into procedure values (3,'Vaccination',95);
insert into procedure values (4,'Blood test',143);
insert into procedure values (5,'Stress test',233);

```

```

insert into visit values (1,to_date('20100101','YYYYMMDD'),2,300);
insert into visit values (2,to_date('20100202','YYYYMMDD'),3,301);
insert into visit values (3,to_date('20100303','YYYYMMDD'),4,302);
insert into visit values (4,to_date('20100404','YYYYMMDD'),5,300);
insert into visit values (5,to_date('20100505','YYYYMMDD'),6,301);
insert into visit values (6,to_date('20100606','YYYYMMDD'),7,302);
insert into visit values (7,to_date('20100707','YYYYMMDD'),8,300);
insert into visit values (8,to_date('20100808','YYYYMMDD'),9,301);
insert into visit values (9,to_date('20100909','YYYYMMDD'),10,302);
insert into visit values (10,to_date('20101010','YYYYMMDD'),11,300);
insert into visit values (11,to_date('20101111','YYYYMMDD'),12,301);

```

```
insert into visit values (12,to_date('20101212','YYYYMMDD'),13,302);
insert into visit values (13,to_date('20100101','YYYYMMDD'),14,300);
insert into visit values (14,to_date('20100202','YYYYMMDD'),15,301);
insert into visit values (15,to_date('20100303','YYYYMMDD'),16,302);
insert into visit values (16,to_date('20100404','YYYYMMDD'),17,300);
insert into visit values (17,to_date('20100505','YYYYMMDD'),18,301);
insert into visit values (18,to_date('20100606','YYYYMMDD'),19,302);
insert into visit values (19,to_date('20100707','YYYYMMDD'),20,300);
insert into visit values (20,to_date('20100808','YYYYMMDD'),1,301);
insert into visit values (21,to_date('20100909','YYYYMMDD'),2,302);
insert into visit values (22,to_date('20101010','YYYYMMDD'),3,300);
insert into visit values (23,to_date('20101111','YYYYMMDD'),4,301);
insert into visit values (24,to_date('20101212','YYYYMMDD'),5,302);
insert into visit values (25,to_date('20100101','YYYYMMDD'),6,300);
insert into visit values (26,to_date('20100202','YYYYMMDD'),7,301);
insert into visit values (27,to_date('20100303','YYYYMMDD'),8,302);
insert into visit values (28,to_date('20100404','YYYYMMDD'),9,300);
insert into visit values (29,to_date('20100505','YYYYMMDD'),10,301);
insert into visit values (30,to_date('20100606','YYYYMMDD'),11,302);
```

```
insert into procedurevisit values (1,1);
insert into procedurevisit values (2,2);
insert into procedurevisit values (3,3);
insert into procedurevisit values (4,4);
insert into procedurevisit values (5,5);
insert into procedurevisit values (6,6);
insert into procedurevisit values (7,7);
insert into procedurevisit values (8,8);
insert into procedurevisit values (9,9);
insert into procedurevisit values (10,10);
insert into procedurevisit values (11,1);
insert into procedurevisit values (12,2);
insert into procedurevisit values (13,3);
insert into procedurevisit values (14,4);
insert into procedurevisit values (15,5);
insert into procedurevisit values (16,6);
insert into procedurevisit values (17,7);
insert into procedurevisit values (18,8);
insert into procedurevisit values (19,9);
insert into procedurevisit values (20,10);
insert into procedurevisit values (21,1);
insert into procedurevisit values (22,2);
insert into procedurevisit values (23,3);
insert into procedurevisit values (24,4);
insert into procedurevisit values (25,5);
insert into procedurevisit values (26,6);
insert into procedurevisit values (27,7);
insert into procedurevisit values (28,8);
insert into procedurevisit values (29,9);
insert into procedurevisit values (30,10);
insert into procedurevisit values (1,10);
insert into procedurevisit values (2,9);
insert into procedurevisit values (3,8);
insert into procedurevisit values (4,7);
insert into procedurevisit values (5,6);
insert into procedurevisit values (6,4);
insert into procedurevisit values (7,3);
```

```

insert into procedurevisit values (8,3);
insert into procedurevisit values (9,2);
insert into procedurevisit values (10,1);
insert into procedurevisit values (11,10);
insert into procedurevisit values (12,9);
insert into procedurevisit values (13,8);
insert into procedurevisit values (14,7);
insert into procedurevisit values (15,6);

insert into bill values
(1,to_date('20100120','YYYYMMDD'),333,333,to_date('20100215','YYYYMMDD'),'Cash',1);
insert into bill values
(2,to_date('20100220','YYYYMMDD'),746,746,to_date('20100315','YYYYMMDD'),'Check',2);
insert into bill values
(3,to_date('20100320','YYYYMMDD'),167,167,to_date('20100415','YYYYMMDD'),'CC',3);
insert into bill values
(4,to_date('20100420','YYYYMMDD'),374,374,to_date('20100515','YYYYMMDD'),'Cash',4);
insert into bill values
(5,to_date('20100520','YYYYMMDD'),838,838,to_date('20100615','YYYYMMDD'),'Check',5);
insert into bill values (6,to_date('20100620','YYYYMMDD'),187,0,'','',6);
insert into bill values (7,to_date('20100720','YYYYMMDD'),419,0,'','',7);
insert into bill values (8,to_date('20100820','YYYYMMDD'),939,0,'','',8);
insert into bill values (9,to_date('20100920','YYYYMMDD'),210,0,'','',9);
insert into bill values
(10,to_date('20101020','YYYYMMDD'),471,471,to_date('20101115','YYYYMMDD'),'Cash',10);
insert into bill values
(11,to_date('20101120','YYYYMMDD'),105,95,to_date('20101215','YYYYMMDD'),'Check',11);
insert into bill values
(12,to_date('20101220','YYYYMMDD'),235,235,to_date('20100115','YYYYMMDD'),'CC',12);
insert into bill values
(13,to_date('20100120','YYYYMMDD'),527,527,to_date('20100215','YYYYMMDD'),'Cash',13);
insert into bill values
(14,to_date('20100220','YYYYMMDD'),118,118,to_date('20100315','YYYYMMDD'),'Check',14);
insert into bill values
(15,to_date('20100320','YYYYMMDD'),264,150,to_date('20100415','YYYYMMDD'),'CC',15);
insert into bill values
(16,to_date('20100420','YYYYMMDD'),592,592,to_date('20100515','YYYYMMDD'),'Cash',16);
insert into bill values
(17,to_date('20100520','YYYYMMDD'),132,132,to_date('20100615','YYYYMMDD'),'Check',17);
insert into bill values
(18,to_date('20100620','YYYYMMDD'),296,200,to_date('20100715','YYYYMMDD'),'CC',18);

```

```

insert into bill values
(19,to_date('20100720','YYYYMMDD'),663,663,to_date('20100815','YYYYMMDD'),'Ca
sh',19);
insert into bill values
(20,to_date('20100820','YYYYMMDD'),148,148,to_date('20100915','YYYYMMDD'),'Ch
eck',20);

```

## Output Log

```

delete from bill;
00:50:42 [DELETE - 20 row(s), 0.188 secs] Command processed
delete from procedurevisit;
00:50:42 [DELETE - 45 row(s), 0.062 secs] Command processed
delete from visit;
00:50:42 [DELETE - 30 row(s), 0.063 secs] Command processed
delete from procedure;
00:50:42 [DELETE - 10 row(s), 0.047 secs] Command processed
delete from provider;
00:50:42 [DELETE - 3 row(s), 0.031 secs] Command processed
delete from patient;
00:50:42 [DELETE - 20 row(s), 0.047 secs] Command processed
delete from zipcode2;
00:50:42 [DELETE - 5 row(s), 0.062 secs] Command processed
insert into zipcode2 values (20790,'Capitol Heights','MD');
00:50:42 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into zipcode2 values (20783,'Hyattsville','MD');
00:50:42 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into zipcode2 values (20792,'Upper Marlboro','MD');
00:50:42 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into zipcode2 values (20794,'Jessup','MD');
00:50:42 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into zipcode2 values (20742,'College Park','MD');
00:50:42 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into patient values
(1,'Robert','Smith','H',123456789,to_date('19731001','YYYYMMDD'),'123 Main
Street',20790,'Aetna',123);
00:50:42 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(2,'John','Ortiz','A',142378123,to_date('19560312','YYYYMMDD'),'2342 Wayne
Ave',20790,'Aetna',534);
00:50:42 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into patient values
(3,'Mike','Valencia','B',284756246,to_date('19600515','YYYYMMDD'),'1333
Maple',20790,'Blue Cross',106);
00:50:42 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(4,'Tim','Cruz','C',569512492,to_date('19650801','YYYYMMDD'),'2343
Pine',20790,'United',212);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(5,'Shawn','Grageda','D',113902498,to_date('19821230','YYYYMMDD'),'983
Oak',20790,'Aetna',424);
00:50:43 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into patient values
(6,'Dave','Brazoban','E',227804996,to_date('19760707','YYYYMMDD'),'1328
Willow',20783,'Blue Cross',848);

```



```

00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(7,'Chris','Rivas','F',455609992,to_date('19840202','YYYYMMDD'),'2433
Walnut',20783,'United',169);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(8,'Steve','Wilson','G',911219984,to_date('19430101','YYYYMMDD'),'111
Redwood',20783,'Aetna',338);
00:50:43 [INSERT - 1 row(s), 0.063 secs] Command processed
insert into patient values
(9,'Harry','Henderson','H',182243996,to_date('19670205','YYYYMMDD'),'9873
Cherry',20783,'Blue Cross',676);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(10,'Doug','Feith','I',364487992,to_date('19681109','YYYYMMDD'),'8585
Holly',20783,'United',135);
00:50:43 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into patient values
(11,'Dan','Snyder','J',728975984,to_date('19690911','YYYYMMDD'),'2433
Almond',20792,'Aetna',270);
00:50:43 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into patient values
(12,'Fred','Ortiz','K',145795196,to_date('19701012','YYYYMMDD'),'1724
Pomegranate',20792,'Blue Cross',540);
00:50:43 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into patient values
(13,'Larry','Valencia','L',291590392,to_date('19711113','YYYYMMDD'),'973 1st
St',20792,'United',108);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(14,'Chuck','Cruz','M',583180784,to_date('19721214','YYYYMMDD'),'234 2nd
Ave',20792,'Aetna',216);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(15,'Pat','Grageda','N',116636156,to_date('19730115','YYYYMMDD'),'2432 3rd
St',20794,'Blue Cross',432);
00:50:43 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into patient values
(16,'Bob','Brazoban','O',233272312,to_date('19740216','YYYYMMDD'),'6534 4th
Ave',20794,'United',864);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(17,'Bill','Rivas','P',466544624,to_date('19750317','YYYYMMDD'),'9083 5th
St',20794,'Aetna',172);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(18,'Lenny','Wilson','Q',933089248,to_date('19760418','YYYYMMDD'),'23 7th
Ave',20794,'Blue Cross',344);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(19,'Carl','Henderson','R',186617849,to_date('19770519','YYYYMMDD'),'9087 8th
St',20742,'United',688);
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into patient values
(20,'Julio','Feith','S',373235698,to_date('19780629','YYYYMMDD'),'232 9th
St',20742,'Aetna',137);
00:50:43 [INSERT - 1 row(s), 0.062 secs] Command processed

```

```

INSERT INTO PROVIDER VALUES (301,'Laurie','Blain');
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
INSERT INTO provider VALUES (302,'Joseph','Sinyan');
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
INSERT INTO provider VALUES (300,'Darren','Schneider');
00:50:43 [INSERT - 1 row(s), 0.047 secs] Command processed
INSERT INTO PROCEDURE VALUES (6,'New Patient History',100);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
INSERT INTO PROCEDURE VALUES (7,'Rapid Strep Test',50);
00:50:44 [INSERT - 1 row(s), 0.078 secs] Command processed
INSERT INTO PROCEDURE VALUES (8,'Injury Evaluation',100);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
INSERT INTO PROCEDURE VALUES (9,'Well Check',100);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
INSERT INTO PROCEDURE VALUES (10,'Follow Up',50);
00:50:44 [INSERT - 1 row(s), 0.062 secs] Command processed
insert into procedure values (1,'Colonoscopy',213);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedure values (2,'Chest x-ray',243);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedure values (3,'Vaccination',95);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedure values (4,'Blood test',143);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedure values (5,'Stress test',233);
00:50:44 [INSERT - 1 row(s), 0.032 secs] Command processed
insert into visit values (1,to_date('20100101','YYYYMMDD'),2,300);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (2,to_date('20100202','YYYYMMDD'),3,301);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (3,to_date('20100303','YYYYMMDD'),4,302);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (4,to_date('20100404','YYYYMMDD'),5,300);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (5,to_date('20100505','YYYYMMDD'),6,301);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (6,to_date('20100606','YYYYMMDD'),7,302);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (7,to_date('20100707','YYYYMMDD'),8,300);
00:50:44 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into visit values (8,to_date('20100808','YYYYMMDD'),9,301);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (9,to_date('20100909','YYYYMMDD'),10,302);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (10,to_date('20101010','YYYYMMDD'),11,300);
00:50:44 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (11,to_date('20101111','YYYYMMDD'),12,301);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (12,to_date('20101212','YYYYMMDD'),13,302);
00:50:45 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into visit values (13,to_date('20100101','YYYYMMDD'),14,300);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (14,to_date('20100202','YYYYMMDD'),15,301);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (15,to_date('20100303','YYYYMMDD'),16,302);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (16,to_date('20100404','YYYYMMDD'),17,300);

```

```

00:50:45 [INSERT - 1 row(s), 0.063 secs] Command processed
insert into visit values (17,to_date('20100505','YYYYMMDD'),18,301);
00:50:45 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into visit values (18,to_date('20100606','YYYYMMDD'),19,302);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (19,to_date('20100707','YYYYMMDD'),20,300);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (20,to_date('20100808','YYYYMMDD'),1,301);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (21,to_date('20100909','YYYYMMDD'),2,302);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (22,to_date('20101010','YYYYMMDD'),3,300);
00:50:45 [INSERT - 1 row(s), 0.062 secs] Command processed
insert into visit values (23,to_date('20101111','YYYYMMDD'),4,301);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (24,to_date('20101212','YYYYMMDD'),5,302);
00:50:45 [INSERT - 1 row(s), 0.078 secs] Command processed
insert into visit values (25,to_date('20100101','YYYYMMDD'),6,300);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (26,to_date('20100202','YYYYMMDD'),7,301);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (27,to_date('20100303','YYYYMMDD'),8,302);
00:50:45 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into visit values (28,to_date('20100404','YYYYMMDD'),9,300);
00:50:45 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (29,to_date('20100505','YYYYMMDD'),10,301);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into visit values (30,to_date('20100606','YYYYMMDD'),11,302);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (1,1);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (2,2);
00:50:46 [INSERT - 1 row(s), 0.234 secs] Command processed
insert into procedurevisit values (3,3);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (4,4);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (5,5);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (6,6);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (7,7);
00:50:46 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into procedurevisit values (8,8);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (9,9);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (10,10);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (11,1);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (12,2);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (13,3);
00:50:46 [INSERT - 1 row(s), 0.032 secs] Command processed
insert into procedurevisit values (14,4);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed

```

```
insert into procedurevisit values (15,5);
00:50:46 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (16,6);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (17,7);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (18,8);
00:50:47 [INSERT - 1 row(s), 0.062 secs] Command processed
insert into procedurevisit values (19,9);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (20,10);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (21,1);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (22,2);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (23,3);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (24,4);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (25,5);
00:50:47 [INSERT - 1 row(s), 0.062 secs] Command processed
insert into procedurevisit values (26,6);
00:50:47 [INSERT - 1 row(s), 0.063 secs] Command processed
insert into procedurevisit values (27,7);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (28,8);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (29,9);
00:50:47 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into procedurevisit values (30,10);
00:50:47 [INSERT - 1 row(s), 0.032 secs] Command processed
insert into procedurevisit values (1,10);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (2,9);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (3,8);
00:50:47 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (4,7);
00:50:47 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into procedurevisit values (5,6);
00:50:48 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (6,4);
00:50:48 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (7,3);
00:50:48 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (8,3);
00:50:48 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (9,2);
00:50:48 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (10,1);
00:50:48 [INSERT - 1 row(s), 0.156 secs] Command processed
insert into procedurevisit values (11,10);
00:50:48 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (12,9);
00:50:48 [INSERT - 1 row(s), 0.109 secs] Command processed
insert into procedurevisit values (13,8);
```

```

00:50:48 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (14,7);
00:50:48 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into procedurevisit values (15,6);
00:50:48 [INSERT - 1 row(s), 0.172 secs] Command processed
insert into bill values
(1,to_date('20100120','YYYYMMDD'),333,333,to_date('20100215','YYYYMMDD'),'Cas
h',1);
00:50:48 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into bill values
(2,to_date('20100220','YYYYMMDD'),746,746,to_date('20100315','YYYYMMDD'),'Che
ck',2);
00:50:48 [INSERT - 1 row(s), 0.141 secs] Command processed
insert into bill values
(3,to_date('20100320','YYYYMMDD'),167,167,to_date('20100415','YYYYMMDD'),'CC'
,3);
00:50:49 [INSERT - 1 row(s), 0.125 secs] Command processed
insert into bill values
(4,to_date('20100420','YYYYMMDD'),374,374,to_date('20100515','YYYYMMDD'),'Cas
h',4);
00:50:49 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into bill values
(5,to_date('20100520','YYYYMMDD'),838,838,to_date('20100615','YYYYMMDD'),'Che
ck',5);
00:50:49 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into bill values (6,to_date('20100620','YYYYMMDD'),187,0,'',' ',6);
00:50:49 [INSERT - 1 row(s), 0.157 secs] Command processed
insert into bill values (7,to_date('20100720','YYYYMMDD'),419,0,'',' ',7);
00:50:49 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into bill values (8,to_date('20100820','YYYYMMDD'),939,0,'',' ',8);
00:50:49 [INSERT - 1 row(s), 0.110 secs] Command processed
insert into bill values (9,to_date('20100920','YYYYMMDD'),210,0,'',' ',9);
00:50:49 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into bill values
(10,to_date('20101020','YYYYMMDD'),471,471,to_date('20101115','YYYYMMDD'),'Ca
sh',10);
00:50:49 [INSERT - 1 row(s), 0.156 secs] Command processed
insert into bill values
(11,to_date('20101120','YYYYMMDD'),105,95,to_date('20101215','YYYYMMDD'),'Che
ck',11);
00:50:49 [INSERT - 1 row(s), 0.063 secs] Command processed
insert into bill values
(12,to_date('20101220','YYYYMMDD'),235,235,to_date('20100115','YYYYMMDD'),'CC'
,12);
00:50:49 [INSERT - 1 row(s), 0.156 secs] Command processed
insert into bill values
(13,to_date('20100120','YYYYMMDD'),527,527,to_date('20100215','YYYYMMDD'),'Ca
sh',13);
00:50:50 [INSERT - 1 row(s), 0.031 secs] Command processed
insert into bill values
(14,to_date('20100220','YYYYMMDD'),118,118,to_date('20100315','YYYYMMDD'),'Ch
eck',14);
00:50:50 [INSERT - 1 row(s), 0.046 secs] Command processed
insert into bill values
(15,to_date('20100320','YYYYMMDD'),264,150,to_date('20100415','YYYYMMDD'),'CC'
,15);
00:50:50 [INSERT - 1 row(s), 0.110 secs] Command processed

```

```
insert into bill values
(16,to_date('20100420','YYYYMMDD'),592,592,to_date('20100515','YYYYMMDD'),'Ca
sh',16);
00:50:50 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into bill values
(17,to_date('20100520','YYYYMMDD'),132,132,to_date('20100615','YYYYMMDD'),'Ch
eck',17);
00:50:50 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into bill values
(18,to_date('20100620','YYYYMMDD'),296,200,to_date('20100715','YYYYMMDD'),'CC
',18);
00:50:50 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into bill values
(19,to_date('20100720','YYYYMMDD'),663,663,to_date('20100815','YYYYMMDD'),'Ca
sh',19);
00:50:50 [INSERT - 1 row(s), 0.047 secs] Command processed
insert into bill values
(20,to_date('20100820','YYYYMMDD'),148,148,to_date('20100915','YYYYMMDD'),'Ch
eck',20);
00:50:50 [INSERT - 1 row(s), 0.141 secs] Command processed
... 140 statement(s) executed, 266 row(s) affected, exec/fetch time:
7.956/0.000 sec [140 successful, 0 warnings, 0 errors]
```

5.	Queries SQL queries	<ul style="list-style-type: none"> <li>• Write 5 simple queries</li> <li>• Write 7 advanced queries</li> <li>• Document the purpose of the queries</li> <li>• Include code and output/log</li> </ul>
----	------------------------	--

### Simple Queries (Code & Output)

1. Help user to locate a patient's insurance information:

```
Select patientfname, patientlname, patientinsco, patientinspolicynum
FROM Patient;
```

PATIENTFNAME	PATIENTLNAME	PATIENTINSOCO	PATIENTINSPOLICYNUM
Mike	Valencia	Blue Cross	106
Tim	Cruz	United	212
Shawn	Grageda	Aetna	424
Dave	Brazoban	Blue Cross	848
Chris	Rivas	United	169
Steve	Wilson	Aetna	338
Harry	Henderson	Blue Cross	676
Doug	Feith	United	135
Dan	Snyder	Aetna	270
Fred	Ortiz	Blue Cross	540
Larry	Valencia	United	108
Chuck	Cruz	Aetna	216
Pat	Grageda	Blue Cross	432
Bob	Brazoban	United	864
Bill	Rivas	Aetna	172
Lenny	Wilson	Blue Cross	344
Carl	Henderson	United	688
Julio	Feith	Aetna	137
Robert	Smith	Aetna	123
John	Ortiz	Aetna	534

2. See how many patients belong to the practice:

```
select count(patientid) patients from patient;
```

```
PATIENTS
-----
20
```

3. Tells a user which patients have insurance from United:

```
Select PatientFname, PatientLname FROM Patient WHERE Patientinsco =
'United';
```

```
PATIENTFNAME  PATIENTLNAME
-----
```

Tim	Cruz
Chris	Rivas
Doug	Feith
Larry	Valencia
Bob	Brazoban
Carl	Henderson

#### 4. Displays amount billed by date:

```
Select billdate, sum(billamt) from bill group by billdate order by
billdate;
```

BILLDATE	SUM(BILLAMT)
-----	-----
2010-01-20 00:00:00	860
2010-02-20 00:00:00	864
2010-03-20 00:00:00	431
2010-04-20 00:00:00	966
2010-05-20 00:00:00	970
2010-06-20 00:00:00	483
2010-07-20 00:00:00	1082
2010-08-20 00:00:00	1087
2010-09-20 00:00:00	210
2010-10-20 00:00:00	471
2010-11-20 00:00:00	105
2010-12-20 00:00:00	235

#### 5. Shows the costs of procedures:

```
Select procedurename, procedurecost FROM procedure order by
procedurename;
```

PROCEDURENAME	PROCEDURECOST
-----	-----
Blood test	143
Chest x-ray	243
Colonoscopy	213
Follow Up	50
Injury Evaluation	100
New Patient History	100
Rapid Strep Test	50
Stress test	233
Vaccination	95
Well Check	100

### Advanced Queries (Code & Output)

#### 1. Finds patients who have not had an appointment in at least six months:

```
Select PatientID, PatientFname, PatientLname FROM Patient p WHERE
```



```
EXISTS (SELECT 'X' FROM Visit WHERE p.patientID = patientID and
visitdatetime < (select sysdate - 6*30 from dual));
```

PATIENTID	PATIENTFNAME	PATIENTLNAME
6	Dave	Brazoban
7	Chris	Rivas
17	Bill	Rivas
5	Shawn	Grageda
16	Bob	Brazoban
8	Steve	Wilson
2	John	Ortiz
3	Mike	Valencia
18	Lenny	Wilson
14	Chuck	Cruz
4	Tim	Cruz
15	Pat	Grageda
9	Harry	Henderson
10	Doug	Feith

## 2. Finds patients who have a balance due:

```
select p.patientlname, p.patientfname, b.billdate, b.billamt -
b.billpaidamt balance_due from patient p join visit v on p.patientid =
v.patientid join bill b on b.visitid = v.visitid where b.billpaidamt <
b.billamt;
```

PATIENTLNAME	PATIENTFNAME	BILLDATE	BALANCE_DUE
Cruz	Tim	2010-03-20 00:00:00	127
Rivas	Chris	2010-06-20 00:00:00	187
Wilson	Steve	2010-07-20 00:00:00	419
Henderson	Harry	2010-08-20 00:00:00	939
Feith	Doug	2010-09-20 00:00:00	210
Ortiz	Fred	2010-11-20 00:00:00	10
Brazoban	Bob	2010-03-20 00:00:00	114
Henderson	Carl	2010-06-20 00:00:00	96
Ortiz	John	2010-01-20 00:00:00	313

## 3. Retrieves information necessary to invoice a customer:

```
Select p.patientid, p.patientfname, p.patientlname, v.visitdatetime,
v.visitid, b.billamt FROM patient p JOIN visit v ON (p.patientid
=v.patientid) JOIN bill b ON (b.visitid = v.visitid);
```

PATIENTID	PATIENTFNAME	PATIENTLNAME	VISITDATETIME	VISITID	BILLAMT
3	Mike	Valencia	2010-02-02 00:00:00	2	746
4	Tim	Cruz	2010-03-03 00:00:00	3	167
5	Shawn	Grageda	2010-04-04 00:00:00	4	374
6	Dave	Brazoban	2010-05-05 00:00:00	5	838
7	Chris	Rivas	2010-06-06 00:00:00	6	187
8	Steve	Wilson	2010-07-07 00:00:00	7	419
9	Harry	Henderson	2010-08-08 00:00:00	8	939

10	Doug	Feith	2010-09-09 00:00:00	9	210
11	Dan	Snyder	2010-10-10 00:00:00	10	471
12	Fred	Ortiz	2010-11-11 00:00:00	11	105
13	Larry	Valencia	2010-12-12 00:00:00	12	235
14	Chuck	Cruz	2010-01-01 00:00:00	13	527
15	Pat	Grageda	2010-02-02 00:00:00	14	118
16	Bob	Brazoban	2010-03-03 00:00:00	15	264
17	Bill	Rivas	2010-04-04 00:00:00	16	592
18	Lenny	Wilson	2010-05-05 00:00:00	17	132
19	Carl	Henderson	2010-06-06 00:00:00	18	296
20	Julio	Feith	2010-07-07 00:00:00	19	663
1	Robert	Smith	2010-08-08 00:00:00	20	148
2	John	Ortiz	2010-01-01 00:00:00	1	333

4. Shows the amount billed by provider (to see who is making more money for the practice):

```
select p.providerlname provider, sum(b.billamt) amt from bill b join
visit v on b.visitid = v.visitid join provider p on v.providerid =
p.providerid group by p.providerlname;
```

PROVIDER	AMT
-----	----
Schneider	3379
Blain	3026
Sinyan	1359

5. Shows how many patients each provider saw each month, for workload balancing purposes:

```
select providerlname p, x.period, x.visits from provider p join (select
to_char(visitdatetime, 'YYYY-MM') period, providerid, count(visitid)
visits from visit group by to_char(visitdatetime, 'YYYY-MM'),
providerid order by to_char(visitdatetime, 'YYYY-MM'))x on x.providerid
= p.providerid;
```

PROVIDER	PERIOD	VISITS
-----	-----	-----
Schneider	2010-10	2
Schneider	2010-01	3
Schneider	2010-04	3
Schneider	2010-07	2
Blain	2010-11	2
Blain	2010-05	3
Blain	2010-02	3
Blain	2010-08	2
Sinyan	2010-09	2
Sinyan	2010-12	2
Sinyan	2010-03	3
Sinyan	2010-06	3

6. Shows procedures performed, in order of frequency:

```
select p.procedurename, p.procedurecost, count(pv.procedureid) count
from procedure p join procedurevisit pv on p.procedureid =
pv.procedureid group by p.procedurename, p.procedurecost order by count
desc;
```

PROCEDURENAME	PROCEDURECOST	COUNT
-----	-----	-----
Vaccination	95	5
Well Check	100	5
Follow Up	50	5
New Patient History	100	5
Rapid Strep Test	50	5
Injury Evaluation	100	5
Chest x-ray	243	4
Blood test	143	4
Colonoscopy	213	4
Stress test	233	3

7. Retrieves the average and standard deviation of the age of all patients:

```
select round(avg((sysdate - patientdob)/365.25),1) avg_age,
round(stddev((sysdate - patientdob)/365.25),1) stddev_age from patient;
```

AVG_AGE	STDDEV_AGE
-----	-----
40	9.4

6.	Database Maintenance Scripts	<ul style="list-style-type: none"> <li>• Submit 1 or 2 scripts</li> <li>• Process multiple rows</li> <li>• Use ALTER command</li> </ul>	<ul style="list-style-type: none"> <li>• Include code and output/log</li> </ul>
----	------------------------------	---	---

The purpose of this script is to create an appointment table and to automate the assignment of workload to staff members.

### Code, including alter command

```
drop table appointments;
drop sequence appointment_id_seq_new;

CREATE TABLE APPOINTMENTS
(
  APPID NUMBER(5,0) NOT NULL ENABLE,
  APPTDATE DATE,
  APPTIME VARCHAR2(5),
  APPPATIENT NUMBER(9,0),
  APPPROVIDER NUMBER(9,0),
  APPPROCEDURE NUMBER(9,0),
  APPCOMPLETED NUMBER(1)
);

ALTER TABLE APPOINTMENTS
ADD CONSTRAINT APPPATIENT_FK FOREIGN KEY (APPPATIENT) REFERENCES
PATIENT(PATIENTID);
ALTER TABLE APPOINTMENTS
ADD CONSTRAINT APPPROVIDER_FK FOREIGN KEY (APPPROVIDER) REFERENCES
PROVIDER(PROVIDERID);
ALTER TABLE APPOINTMENTS
ADD CONSTRAINT APPPROCEDURE_FK FOREIGN KEY (APPPROCEDURE) REFERENCES
PROCEDURE(PROCEDUREID);

CREATE SEQUENCE Appointment_id_seq_new START WITH 1 NOCACHE;

INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '9:00', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '9:30', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '10:00', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '10:30', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '11:00', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '11:30', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '12:00', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '12:30', 0);
```

```

INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '2:00', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '2:30', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '3:00', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '3:30', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '4:00', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '4:30', 0);
INSERT INTO APPOINTMENTS (APPID, APPTDATE, APPTIME,APPCOMPLETED)
VALUES (Appointment_id_seq_new.NEXTVAL, CURRENT_DATE, '5:00', 0);

--*** Dividing the appointments per provider each 1/3 ***

UPDATE APPOINTMENTS SET APPPROVIDER = '302' WHERE MOD(APPID,3) = 0;

UPDATE APPOINTMENTS SET APPPROVIDER = '301' WHERE MOD(APPID,2) = 0 AND
APPPROVIDER IS NULL;

UPDATE APPOINTMENTS SET APPPROVIDER = '300' WHERE APPPROVIDER IS NULL;

-- PROVIDER 302 HAD THE DAY OFF - now divide 1/2 **

UPDATE APPOINTMENTS SET APPPROVIDER = NULL;

UPDATE APPOINTMENTS SET APPPROVIDER = '301' WHERE MOD(APPID,2) = 0 AND
APPPROVIDER IS NULL;

UPDATE APPOINTMENTS SET APPPROVIDER = '300' WHERE APPPROVIDER IS NULL;

```

### Output Log, showing multiple rows processed

```

Table dropped.
Sequence dropped.
Table created.
Table altered.
Table altered.
Table altered.
Sequence created.
1 row created.
1 row created.
1 row created.

```

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

5 rows updated.

5 rows updated.

5 rows updated.

15 rows updated.

7 rows updated.

8 rows updated.

7.	Create a Stored Procedure	<ul style="list-style-type: none"> <li>• Include code and output</li> </ul>
----	---------------------------	---

The purpose of this stored procedure is to update information about bills that have been paid. Presumably, the bookkeeper in the doctor's office would use an Internet-based application to input data about bills that have been paid. Submitting the web-based form would execute the stored procedure, to which the entered data would be passed.

### Code

```
create or replace procedure bill_paid
(
  ar_billid          in  number
,ar_billpaidamt      in  number
,ar_billpaidddate    in  date
,ar_billpaidmethod   in  varchar2
)
is
begin
update bill
set  billpaidamt = ar_billpaidamt
    ,billpaidddate = ar_billpaidddate
    ,billpaidmethod = ar_billpaidmethod
where billid = ar_billid;
if sql%NOTFOUND
then raise_application_error(-20000,'Bill ID ' || ar_billid || ' not
found.');
```

### Output

```
SQL> create or replace procedure bill_paid
2  (
3    ar_billid          in  number
,ar_billpaidamt      in  number
4    5 ,ar_billpaidddate    in  date
,ar_billpaidmethod   in  varchar2
6    7 )
8  is
9  begin
10 update bill
11 set  billpaidamt = ar_billpaidamt
    ,billpaidddate = ar_billpaidddate
    ,billpaidmethod = ar_billpaidmethod
12 13 14 where billid = ar_billid;
15 if sql%NOTFOUND
16 then raise_application_error(-20000,'Bill ID ' || ar_billid || ' not
found.');
```

19 /

Procedure created.

**The following demonstrates the functionality of the stored procedure.**

SQL> select \* from bill where billid = 3;

BILLID	BILLDATE	BILLAMT	BILLPAIDAMT	BILLPAIDD	BILLPAID	VISITID
3	20-MAR-10	167	167	15-APR-10	CC	3

SQL> execute bill\_paid (3,40,'10-MAR-2010','Check');

PL/SQL procedure successfully completed.

SQL> commit;

Commit complete.

SQL> select \*from bill where billid = 3;

BILLID	BILLDATE	BILLAMT	BILLPAIDAMT	BILLPAIDD	BILLPAID	VISITID
3	20-MAR-10	167	40	10-MAR-10	Check	3

**The following shows the error message when an invalid parameter is passed to the procedure.**

SQL> execute bill\_paid (300,46,'10-APR-2010','Cash');  
BEGIN bill\_paid (300,46,'10-APR-2010','Cash'); END;

\*

ERROR at line 1:  
ORA-20000: Bill ID 300 not found.  
ORA-06512: at "DB652A17.BILL\_PAID", line 16  
ORA-06512: at line 1



8.	Usage of Oracle's EXPLAIN tool	<ul style="list-style-type: none"> <li>Investigate usage of Oracle's Explain Plan tool</li> <li>Follow examples on Ch 18 of Workbook and article in course Webliography</li> <li>Perform test based on Projection, Selection, Union and Join queries</li> <li>Use Oracle TIMING command</li> <li>Provide the explanation/interpretation of your results</li> <li>Include SQL queries, optimizer output, and times.</li> </ul>
----	--------------------------------	---

### Query 1 – Projection

```
set timing on;
delete from plan_table; explain plan for
select * from procedure;
select * from table(dbms_xplan.display)
```

PLAN\_TABLE\_OUTPUT

Plan hash value: 3217754044

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		10	190	3 (0)	00:00:01
1	TABLE ACCESS FULL	PROCEDURE	10	190	3 (0)	00:00:01

Elapsed: 00:00:00.00

Explanation: This query performs a full table scan, which means that the entire table is accessed.

### Query 2 – Selection

```
set timing on;
delete from plan_table; explain plan for
select PatientFname, PatientLname FROM Patient WHERE Patientinsco =
'United';
select * from table(dbms_xplan.display)
```

PLAN\_TABLE\_OUTPUT

Plan hash value: 3370287156

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		7	154	3 (0)	00:00:01
* 1	TABLE ACCESS FULL	PATIENT	7	154	3 (0)	00:00:01

Predicate Information (identified by operation id):

1 - filter("PATIENTINSCO"='United')

Elapsed: 00:00:00.01

Explanation: This query also performs a full table scan, since none of the selected attributes are indexed.

### Query 3 – Union

```
set timing on;
delete from plan_table; explain plan for
select patientlname lname, patientfname fname from patient union all
select providerlname, providerfname from provider;
select * from table(dbms_xplan.display)
```

PLAN\_TABLE\_OUTPUT

Plan hash value: 2135091174

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		23	325	6 (50)	00:00:01
1	UNION-ALL					
2	TABLE ACCESS FULL	PATIENT	20	280	3 (0)	00:00:01
3	TABLE ACCESS FULL	PROVIDER	3	45	3 (0)	00:00:01

Elapsed: 00:00:00.01

Explanation: This query performs two full table scans, since none of the selected attributes are indexed. The results are passed back to the select statement, via the union-all function.

### Query 4 – Join

```
set timing on;
delete from plan_table; explain plan for
select v.visitid, p.procedurename from procedurevisit v join procedure
p on v.procedureid = p.procedureid;
select * from table(dbms_xplan.display)
```

PLAN\_TABLE\_OUTPUT

Plan hash value: 2768965875

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		45	1035	4 (25)	00:00:01
1	MERGE JOIN		45	1035	4 (25)	00:00:01
2	TABLE ACCESS BY INDEX ROWID	PROCEDURE	10	170	2 (0)	00:00:01
3	INDEX FULL SCAN	PK_PROCEDURE2	10		1 (0)	00:00:01
* 4	SORT JOIN		45	270	2 (50)	00:00:01
5	INDEX FULL SCAN	PK_PROCEDURE_VISIT	45	270	1 (0)	00:00:01

Predicate Information (identified by operation id):

```
-----  
4 - access("V"."PROCEDUREID"="P"."PROCEDUREID")  
    filter("V"."PROCEDUREID"="P"."PROCEDUREID")
```

Elapsed: 00:00:00.01

Explanation: A scan of the indexed column `procedure.procedureid` is performed to return the desired row IDs, which are then passed to the table access by index row ID operation, which in turn passes the results to the merge join operation. A scan of the indexed primary key of `procedurevisit` is likewise performed to return the desired rows (that agree with the predicate information), which are sorted and passed to the merge join. The merge join operation merges the two sorted lists together.

Sources: Rischert (2003), Oracle (2008)

9.	Object-Relational exercises	<ul style="list-style-type: none"> <li>• Create a type</li> <li>• Create variable array</li> <li>• Include code and output/log</li> </ul>
----	-----------------------------	---

The purpose of the objects created in the scripts below is to automate the population of merge fields in written correspondence to patients.

### Create a type:

```
CREATE TYPE LetterBody_typ AS OBJECT (LetterLines VARCHAR2(100));
```

### Create variable array:

```
CREATE TYPE LetterBody_varray_typ AS VARRAY(5) OF LetterBody_typ;
```

### Code and output/log:

```
SQL> CREATE TYPE LetterBody_typ AS OBJECT (LetterLines VARCHAR2(100));
```

```
2 /
```

Type created.

```
SQL> CREATE TYPE LetterBody_varray_typ AS VARRAY(5) OF LetterBody_typ;
```

```
2 /
```

Type created.

```
SQL> CREATE TABLE LetterBody_list (
2   LetterType Varchar2(10),
3   LetterBody_list LetterBody_varray_typ);
```

Table created.

```
SQL> INSERT INTO LetterBody_list VALUES (
2   'Template',
3   LetterBody_varray_typ( LetterBody_typ ('Line1'),
4   LetterBody_typ ('Line2'),
5   LetterBody_typ ('Line3')));
```

1 row created.

```
SQL> INSERT INTO LetterBody_list VALUES (
2   'Past Due',
3   LetterBody_varray_typ( LetterBody_typ ('Your account is now Past
Due!'),
4   LetterBody_typ ('Please contact our office with
payment information.'),
5   LetterBody_typ ('If payment or arrangements have
been made please disregard this notice.')));
```

1 row created.

Desc LetterBody\_list;

Name	Null?	Type
LETTERTYPE		VARCHAR2(10)
LETTERBODY_LIST		LETTERBODY_VARRAY_TYP

Select \* from LetterBody\_list;

LETTERTYPE	LETTERBODY_LIST(LETTERLINES)
Template	LETTERBODY_VARRAY_TYP(LETTERBODY_TYP('Line1'), LETTERBODY_TYP('Line2'), LETTERBODY_TYP('Line3'))
Past Due	LETTERBODY_VARRAY_TYP(LETTERBODY_TYP('Your account is now Past Due!'), LETTERBODY_TYP('Please contact our office with payment information.'), LETTERBODY_TYP('If payment or arrangements have been made please disregard this notice.'))

## References

Oracle. (2008). The Query Optimizer. Retrieved from

[http://download.oracle.com/docs/cd/B19306\\_01/server.102/b14211/optimops.htm](http://download.oracle.com/docs/cd/B19306_01/server.102/b14211/optimops.htm)

Rischert, A. (2004). Oracle SQL by Example. Upper Saddle River, NJ: Prentice Hall.

UMUC. (2011). DBST 652 Lab Description/Example. Retrieved from

<http://tychousa11.umuc.edu/DBST652/1009/9040/class.nsf/Course+Content/08EF78F22128620852577850052C5E8?OpenDocument>