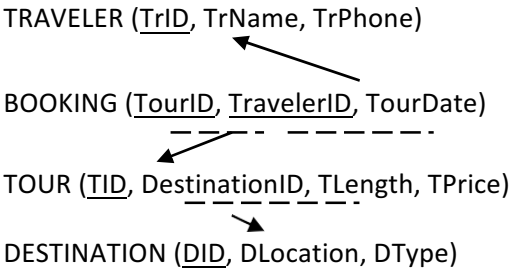
CSC 355 Database Systems 501
Assignment 1 (1/8)

ERICA FILGUERAS

Database Schema:



Database Instance:

TRAVELER

TrID (P)	TrName	TrPhone
0001	White	5551234
0002	Goodman	5559876
0003	Pinkman	5551609
0004	Fring	5552112

BOOKING

TourID (P& F)	TravelerID (P&F)	TourDate
44890	0001	03-JUN-19
44890	0002	03-JUN-19
86428	0004	01-AUG-19
12345	0001	15-AUG-19

TOUR

TID (P)	DestinationID (F)	TLength	TPrice
12345	002	5	4000
44890	016	14	8000
70291	002	3	2500

DESTINATION

DID (P)	DLocation	DType
001	Cleveland	City
002	New York	City
003	Cozumel	Resort
004	Bermuda	Cruise

a. List the attribute(s) that make up the primary key (if one exists) in DESTINATION.
Answer: DID

b. List the attribute(s) that make up the primary key (if one exists) in TOUR.
Answer: TID

c. List the attribute(s) that make up the primary key (if one exists) in BOOKING.
Answer: TourID, TravelerID

d. List the attribute(s) that make up the foreign key(s) (if any exist) in DESTINATION.
Answer: None

e. List the attribute(s) that make up the foreign key(s) (if any exist) in TOUR.
Answer: DestinationID

f. List the attribute(s) that make up the foreign key(s) (if any exist) in BOOKING.
Answer: TourID, TravelerID

g. Construct a new tuple that can be inserted into BOOKING **without violating any constraints.**
Answer: (44890,0001,'04-23-2020')

h. Construct a new tuple that cannot be inserted into BOOKING because doing so **would violate referential integrity** (but would not violate any other constraints)
Answer: (90201,0008,'02-14-2017')

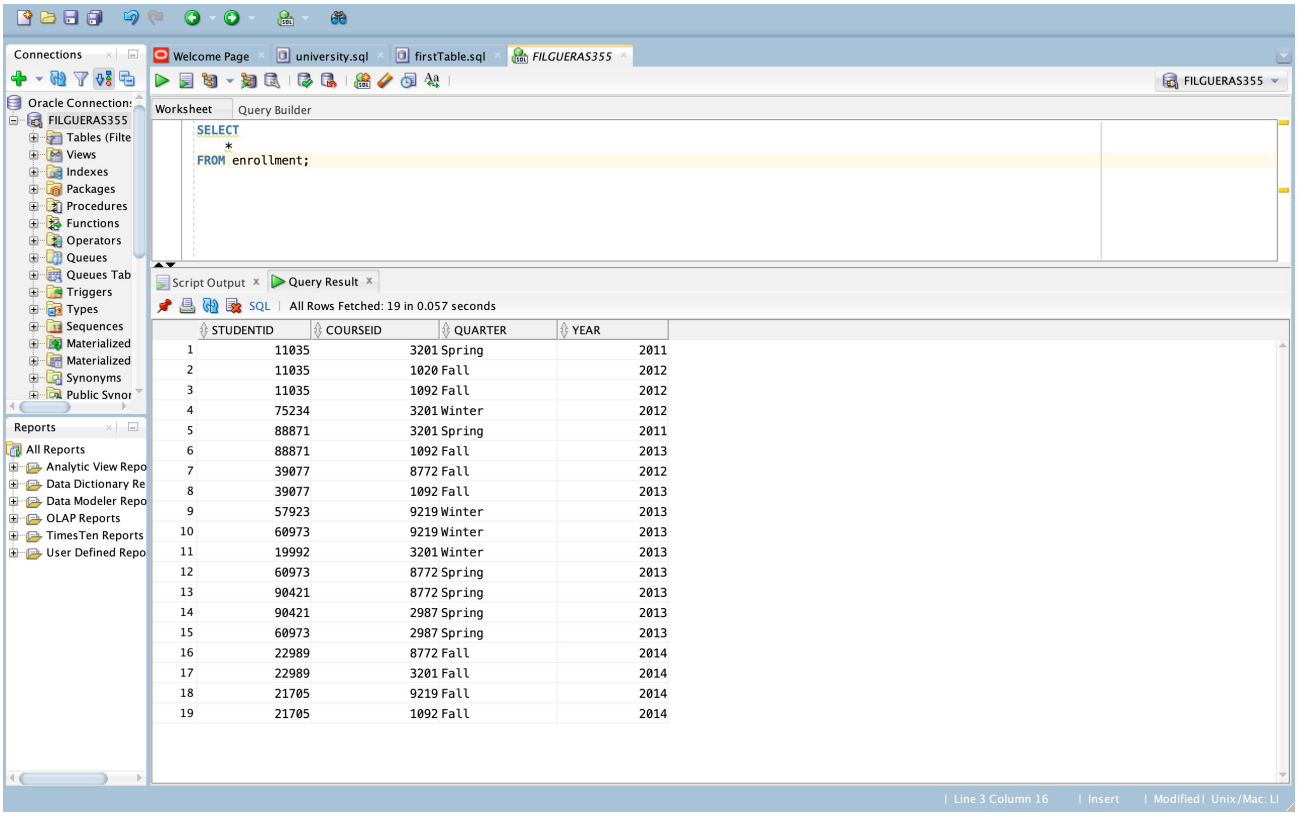
i. Construct a new tuple that can be inserted into TOUR without violating any constraints.
Answer: (54378,003,'01-01-2020')

j. Construct a new tuple that cannot be inserted into TOUR because doing so would violate a key constraint (and thus would also violate entity integrity), but would not violate any other constraints.
Answer: (12345, 002,'05-14-1981')

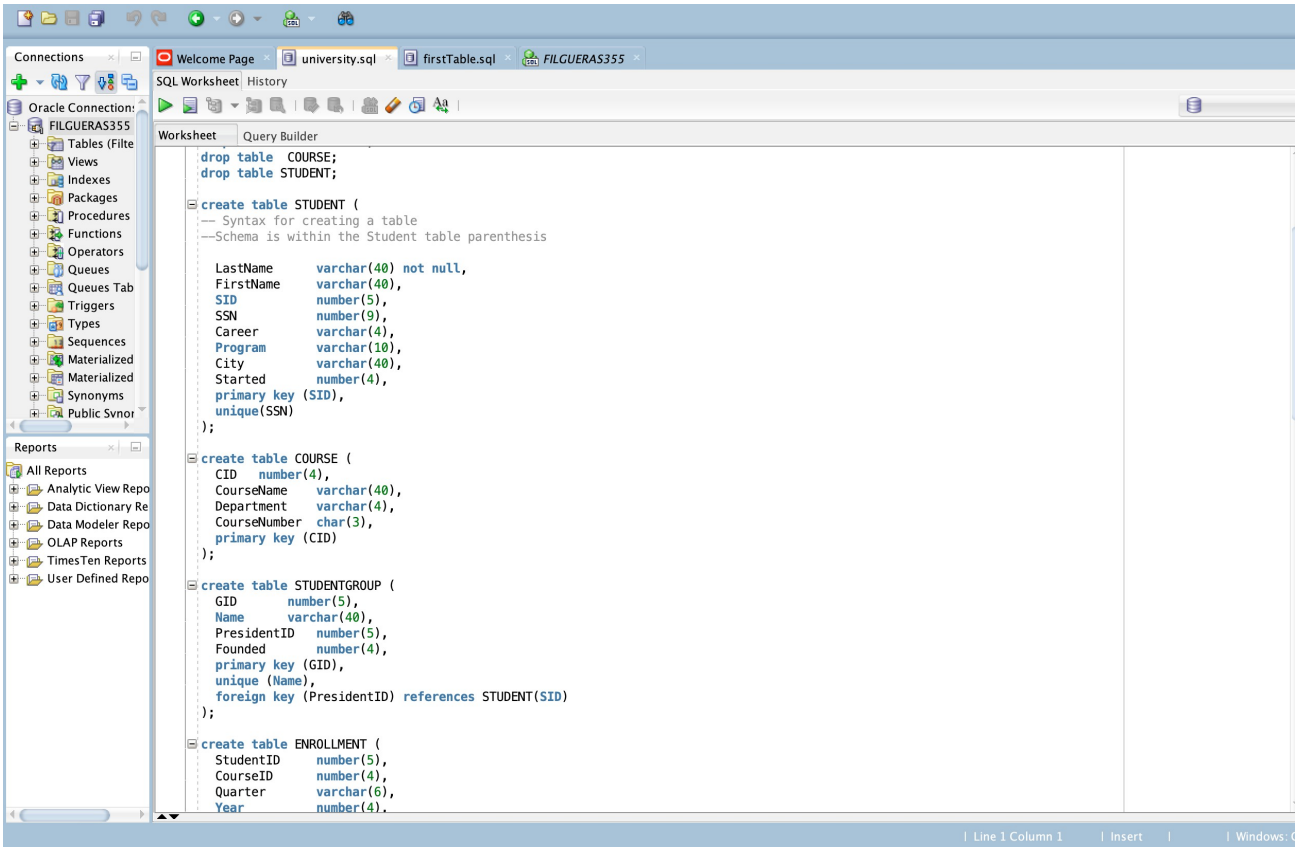
k. Which of the tuples in DESTINATION could be removed without violating referential integrity? Explain why.

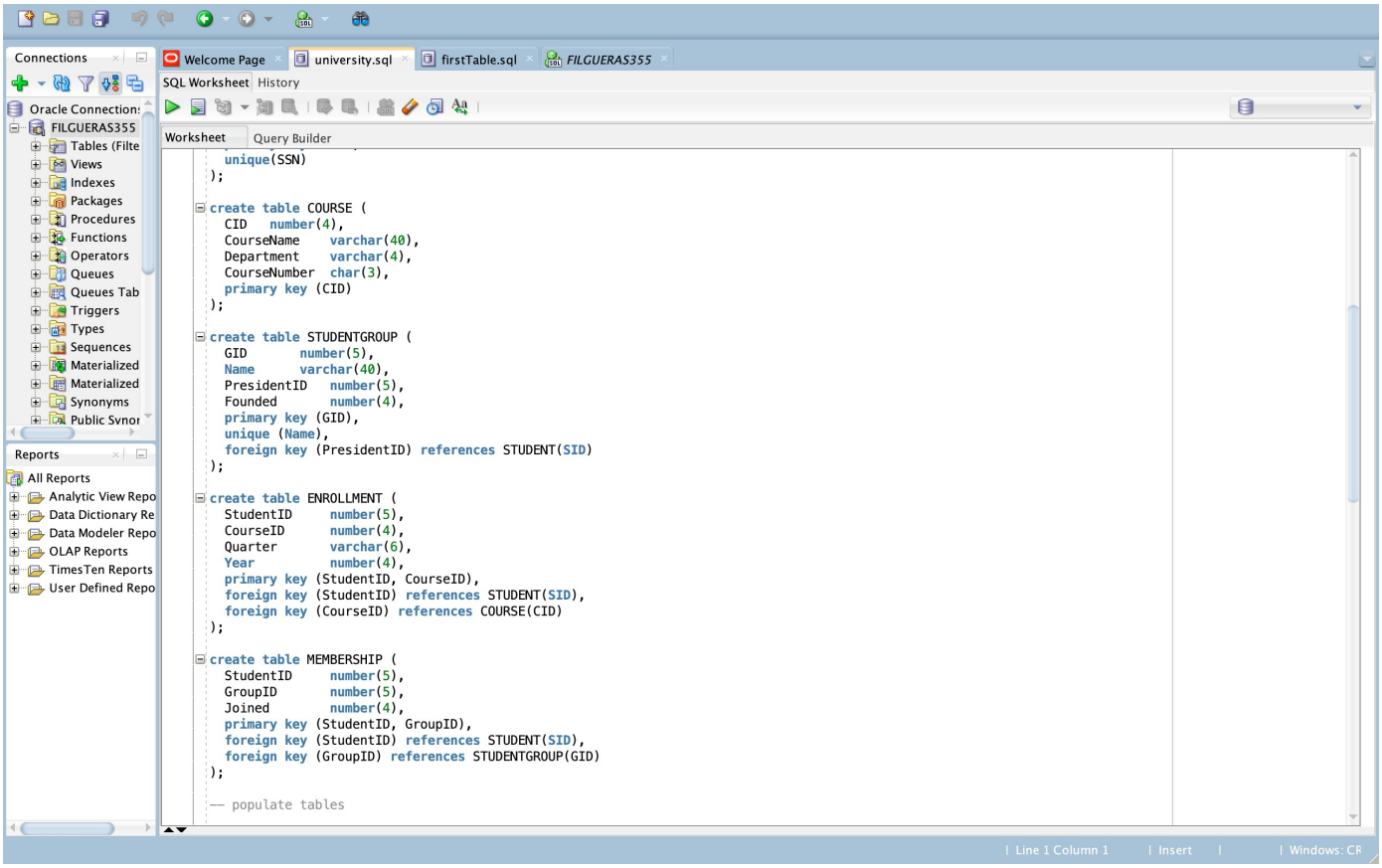
Answer:
(003,'Conzumel','Resort')
(004, 'Bermuda', 'Cruise')
(001,'Cleveland',' City')
- Removing the following tuples wouldn't violate referential integrity since the primary ID of DESTINATION (DID), where TOUR's foreign key (Destination ID) points to, is not referenced by TOUR. In other words, the foreign key of DESTINATION has to coincide with the primary keys in TOUR. Removing these tuples therefore, wouldn't disrupt the foreign keys in TOUR because the primary keys in DESTINATION isn't referenced anywhere.

Screenshots



	STUDENTID	COURSEID	QUARTER	YEAR
1	11035	3201	Spring	2011
2	11035	1020	Fall	2012
3	11035	1092	Fall	2012
4	75234	3201	Winter	2012
5	88871	3201	Spring	2011
6	88871	1092	Fall	2013
7	39077	8772	Fall	2012
8	39077	1092	Fall	2013
9	57923	9219	Winter	2013
10	60973	9219	Winter	2013
11	19992	3201	Winter	2013
12	60973	8772	Spring	2013
13	90421	8772	Spring	2013
14	90421	2987	Spring	2013
15	60973	2987	Spring	2013
16	22989	8772	Fall	2014
17	22989	3201	Fall	2014
18	21705	9219	Fall	2014
19	21705	1092	Fall	2014




Connections
Oracle Connections
FILGUERAS355
Tables (Filte
Views
Indexes
Packages
Procedures
Functions
Operators
Queues
Queues Tab
Triggers
Types
Sequences
Materialized
Materialized
Synonyms
Public Synor
Reports
All Reports
Analytic View Repo
Data Dictionary Re
Data Modeler Repo
OLAP Reports
TimesTen Reports
User Defined Repo
Welcome Page
university.sql
firstTable.sql
FILGUERAS355
SQL Worksheet
History
Worksheet
Query Builder
unique(SSN)
);
create table COURSE (
CID number(4),
CourseName varchar(40),
Department varchar(4),
CourseNumber char(3),
primary key (CID)
);
create table STUDENTGROUP (
GID number(5),
Name varchar(40),
PresidentID number(5),
Founded number(4),
primary key (GID),
unique (Name),
foreign key (PresidentID) references STUDENT(SID)
);
create table ENROLLMENT (
StudentID number(5),
CourseID number(4),
Quarter varchar(6),
Year number(4),
primary key (StudentID, CourseID),
foreign key (StudentID) references STUDENT(SID),
foreign key (CourseID) references COURSE(CID)
);
create table MEMBERSHIP (
StudentID number(5),
GroupID number(5),
Joined number(4),
primary key (StudentID, GroupID),
foreign key (StudentID) references STUDENT(SID),
foreign key (GroupID) references STUDENTGROUP(GID)
);
-- populate tables
Line 1 Column 1
Insert
Windows: CR