

Question 1

The screenshot shows the Oracle SQL Developer interface. The title bar reads "Oracle SQL Developer : Table WJONES17.COURSE@Jones355". The menu bar includes File, Edit, View, Navigate, Run, Team, Tools, Window, and Help. The left pane shows the "Connections" tree with "Jones355" selected, and the "Tables (Filtered)" list containing COURSE, ENROLLMENT, MEMBERSHIP, STUDENTID, GROUPID, JOINED, STUDENT, and STUDENTGROUP. The right pane displays the "COURSE" table data in a grid view. The grid has columns: CID, COURSENAME, DEPARTMENT, and COURSENUMBER. The data is as follows:

	CID	COURSENAME	DEPARTMENT	COURSENUMBER
1	1020	Theory of Computation	CSC	489
2	1092	Cryptography	CSC	440
3	3201	Data Analysis	IT	223
4	9219	Database Systems	CSC	355
5	3111	Theory of Computation	CSC	389
6	8772	History of Games	GAM	206
7	2987	Topics in Digital Cinema DC		270

Question 2

Oracle SQL Developer : Jones355

File Edit View Navigate Run Source Team Tools Window Help

Connections - Jones355

Oracle Connections

- Jones355
 - Tables (Filtered)
 - COURSE
 - ENROLLMENT
 - MEMBERSHIP
 - STUDENTID
 - GROUPID
 - JOINED
 - STUDENT
 - STUDENTGROUP
 - Views
 - Indexes
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Sequences
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - Database Links
 - Public Database Links
 - Directories
 - Editions
 - Java
 - XML Schemas
 - XML DB Repository
 - OLAP Option
 - Analytic Views
 - Scheduler
 - Recycle Bin
 - Other Users
- Oracle NoSQL Connections
- Database Schema Service Connections

Worksheet

Query Builder

SELECT * FROM STUDENT

Script Output

SQL All Rows Fetched: 15 in 0.037 seconds

	LASTNAME	FIRSTNAME	SID	SSN	CAREER	PROGRAM	CITY	STARTED
1	Brennigan	Marcus	90421	987654321	UGRD	COMP-GAM	Evanston	2010
2	Patel	Deepa	14662	(null)	GRAD	COMP-SCI	Evanston	2013
3	Snowdon	Jonathan	88871	123123123	GRAD	INFO-SYS	Springfield	2009
4	Starck	Jason	16992	789789789	UGRD	INFO-SYS	Springfield	2009
5	Johnson	Peter	32105	123456789	UGRD	COMP-SCI	Chicago	2010
6	Winter	Abigail	11035	111111111	GRAD	PHD	Chicago	2009
7	Patel	Prakash	75234	(null)	UGRD	COMP-SCI	Chicago	2011
8	Snowdon	Jennifer	93321	321321321	GRAD	COMP-SCI	Springfield	2012
9	Degroff	Jarvis	14998	113311331	GRAD	COMP-GAM	Evanston	2012
10	Rubik	Dwayne	57923	979797979	UGRD	COMP-GAM	Springfield	2013
11	Skelly	Trinity	58992	555222555	GRAD	PHD	Springfield	2012
12	Krol	Angelo	60973	(null)	UGRD	COMP-SCI	Springfield	2011
13	Pollard	Joya	39077	(null)	GRAD	INFO-SYS	Springfield	2010
14	Peoples	Brandon	22969	555443333	UGRD	COMP-GAM	Milwaukee	2014
15	Peoples	Bridget	21705	571441234	UGRD	COMP-SCI	Milwaukee	2014

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Question 3

a. List the attribute(s) that make up the primary key (if one exists) in TENANT.

The primary key for the TENANT schema is the TID attribute.

b. List the attribute(s) that make up the primary key (if one exists) in LEASE.

The LEASE schema uses the BuildingID and TenantID attributes as a primary key.

c. List the attribute(s) that make up the foreign key(s) (if any exist) in TENANT.

The TENANT schema does not have attributes that function as foreign keys.

d. List the attribute(s) that make up the foreign key(s) (if any exist) in LEASE.

The LEASE schema has two foreign keys: BuildingID and TenantID. The BuildingID attribute acts as a foreign key that relates the LEASE schema to the BUILDING schema via the BID primary key. The TenantID attribute acts as a foreign key that relates the LEASE schema to the TENANT schema via the TID primary key.

e. Construct a new tuple that can be inserted into LEASE without violating any constraints.

One could safely add a tuple with the following values for each of the attributes (shown below in tabular format) to the LEASE instance. Each value is within the domain of its attribute and BuildingID and TenantID are non-null, form a unique set, and reference a primary key in the BUILDING and TENANT instances respectively.

BuildingID	TenantID	Unit	Rent
111	W209	1573	1500

f. Construct a new tuple that cannot be inserted into LEASE because doing so would violate referential integrity (but would not violate any other constraints)

Adding the below tuple (shown in tabular format) to the LEASE instance would break referential integrity. The BuildingID of “123” does not appear as a BID primary key in the BUILDING instance.

BuildingID	TenantID	Unit	Rent
123	W209	1573	1500

Weston Jones – CS 355 Database Systems – Section 602 – Assignment 1 – 4/8/20

g. Construct a new tuple that can be inserted into TENANT without violating any constraints.

The below tuple inserted into the TENANT instance would not violate any constraints. Each value is within the domain of its attribute, the value of TID is unique to the instance, and TID is not a foreign key that has to reference a key elsewhere.

TID	Name	Phone
X304	Jones	123-444-5678

h. Construct a new tuple that cannot be inserted into TENANT because doing so would violate a key constraint (and thus would also violate entity integrity), but would not violate any other constraints.

The below tuple inserted into the TENANT instance would violate a key constraint and thus entity integrity. W021 is already a value of the TID attribute elsewhere in the schema. Since TID is a candidate key and primary key, it must have unique values for each tuple in the instance. Domain integrity is upheld because each value is within the domain of its attribute. Referential integrity is also upheld because W021 does not appear a foreign key elsewhere in the model.

TID	Name	Phone
W021	Jones	123-444-5678

i. List the tuples in BUILDING that could be removed without violating referential integrity, and explain why it is safe to remove them.

Referential Integrity requires that each primary key referenced by a foreign key elsewhere in the model must not be deleted. Therefore, it is only safe to remove tuples whose BID is not referenced by the BuildingID foreign key in the LEASE instance (In plain English, this means we can discard the buildings not currently being leased). The single tuple that is safe to remove is shown below.

BID	Address	City
555	50 Main St	Oxnard