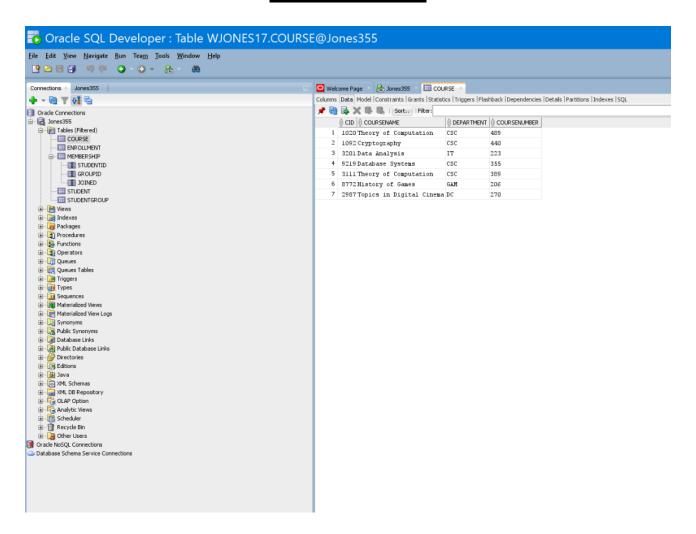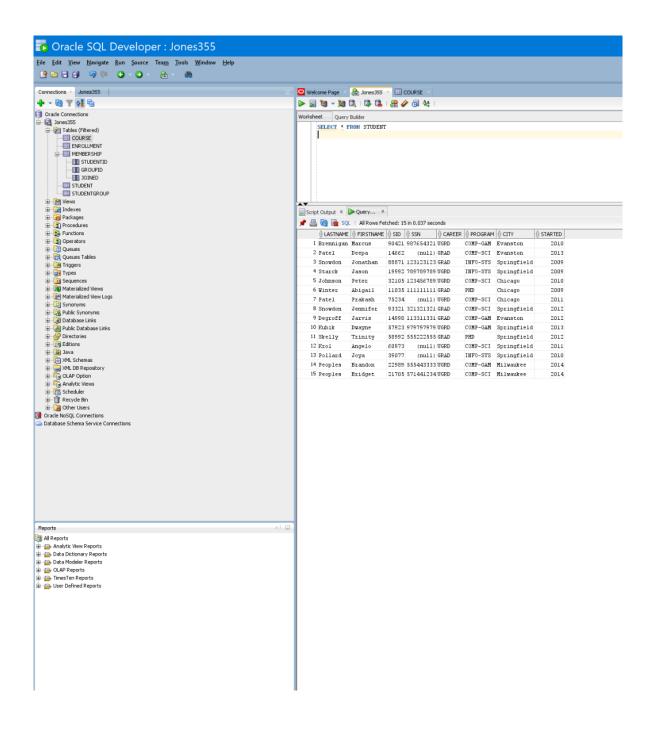# Question 1

Question 2

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

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