

**COURSE: CS/DSA- 4513 – DATABASE MANAGEMENT**  
**SECTION: 001**  
**SEMESTER: FALL 2019**  
**INSTRUCTOR: DR. LE GRUENWALD**

**LINCE RUMAINUM**

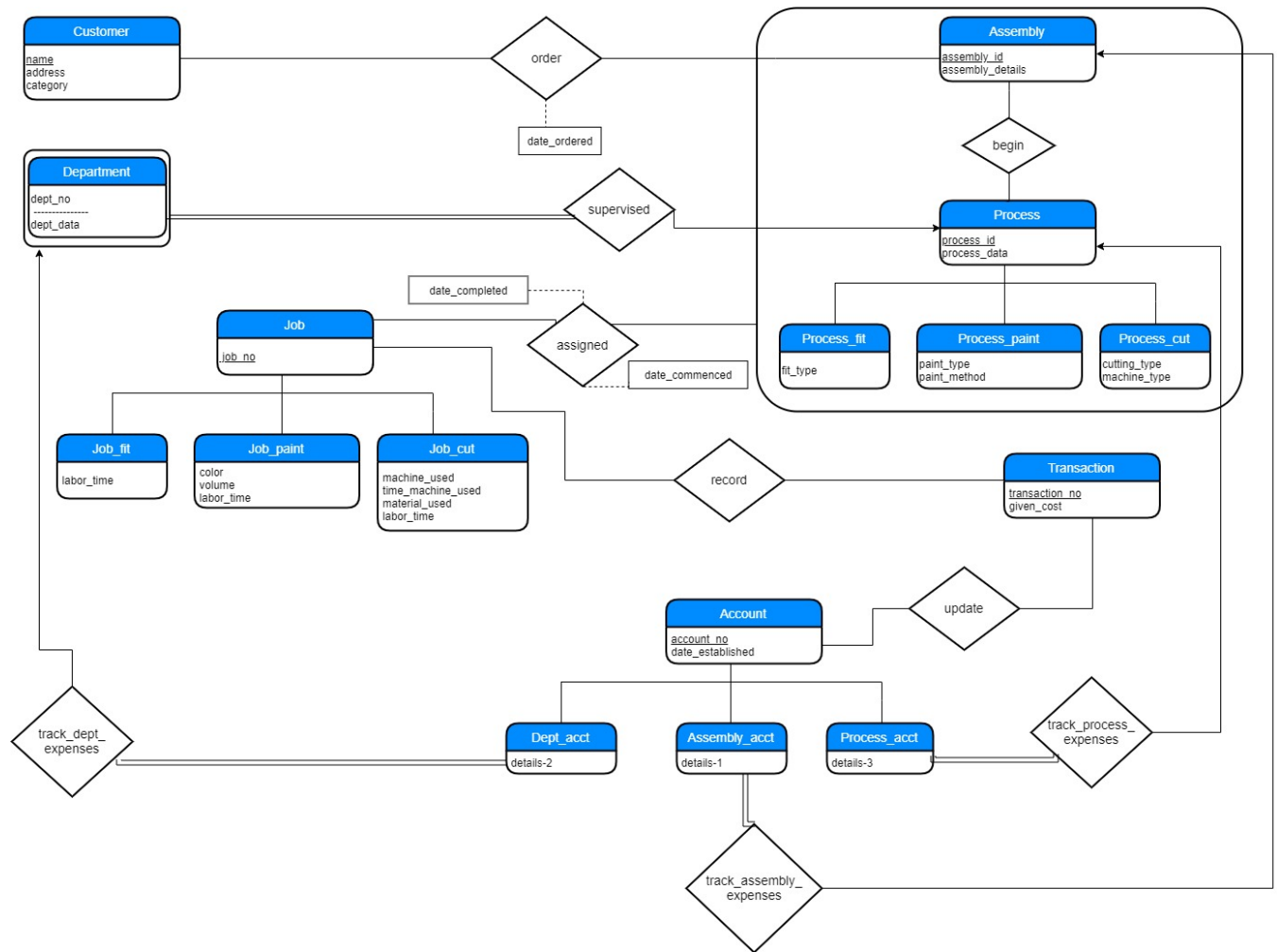
**SCORE:**

## TABLE OF CONTENT

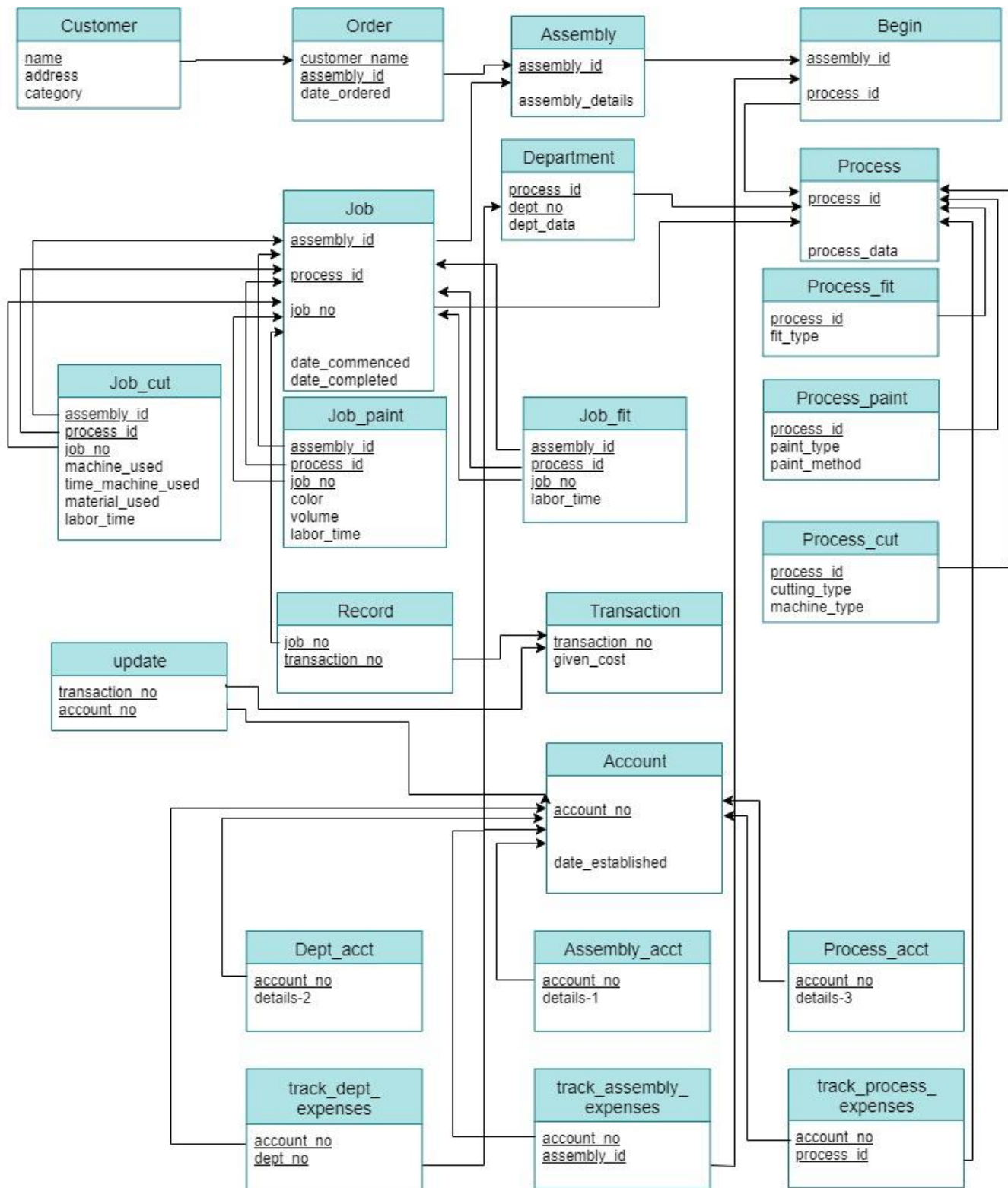
Tasks Performed	Page Number
Task 1.	3-4
1.1. ER Diagram	3-3
1.2. Relational Database Schema	4-4
Task 2. Data Dictionary	5-6
Task 3.	7-8
3.1. Discussion of storage structures for tables	7-8
3.2. Discussion of storage structures for tables (Azure SQL Database)	8-8
Task 4. SQL statements and screenshots showing the creation of tables in Azure SQL Database	9-11

## Task 1

### 1.1. ER Diagram



## 1.2. Relational Database Schema



## Task 2. Data Dictionary

Table	Attributes			
	Name	Type	Size (In Bytes)	CONSTRAINTS
Customer	name	VARCHAR	24	PRIMARY KEY, NOT NULL
	address	VARCHAR	50	
	category	INT	4	INT value between 1-10
Assembly	assembly_id	INT	4	PRIMARY KEY, NOT NULL
	assembly_details	VARCHAR	50	
	date_ordered	DATETIME	8	
Process	process_id	INT	4	PRIMARY KEY, NOT NULL
	process_data	VARCHAR	50	
Process_fit	process_id	INT	4	PRIMARY KEY, NOT NULL
	fit_type	VARCHAR	50	
Process_paint	process_id	INT	4	PRIMARY KEY, NOT NULL
	paint_type	VARCHAR	50	
	paint_method	VARCHAR	50	
Process_cut	process_id	INT	4	PRIMARY KEY, NOT NULL
	cutting_type	VARCHAR	50	
	machine_type	VARCHAR	50	
Department	dept_no	INT	4	
	dept_data	VARCHAR	50	
Job	job_no	INT	4	PRIMARY KEY, NOT NULL
	aseembly_id	INT	4	FOREIGN KEY
	process_id	INT	4	FOREIGN KEY
	date_commenced	DATETIME	8	
	date_completed	DATETIME	8	
Job_fit	job_no	INT	4	FOREIGN KEY
	aseembly_id	INT	4	FOREIGN KEY
	process_id	INT	4	FOREIGN KEY
	labor_time	DECIMAL	5	
Job_paint	job_no	INT	4	FOREIGN KEY
	aseembly_id	INT	4	FOREIGN KEY
	process_id	INT	4	FOREIGN KEY
	color	VARCHAR	20	
	volume	DECIMAL	5	
	labor_time	DECIMAL	5	
Job_cut	job_no	INT	4	FOREIGN KEY
	aseembly_id	INT	4	FOREIGN KEY
	process_id	INT	4	FOREIGN KEY
	machine_used	VARCHAR	50	
	time_machine_used	DECIMAL	5	
	material_used	VARCHAR	50	
	labor_time	DECIMAL	5	
Transaction	transaction_no	INT	4	PRIMARY KEY, NOT NULL
	job_no	INT	4	

	given_cost	DECIMAL	5	
<b>Account</b>	account_no	INT	4	PRIMARY KEY, NOT NULL
	date_established	DATETIME	8	
<b>Dept_acct</b>	account_no	INT	4	FOREIGN KEY
	dept_no	INT	4	FOREIGN KEY
	details-2	VARCHAR	50	
<b>Assembly_acct</b>	account_no	INT	4	FOREIGN KEY
	aseembly_id	INT	4	FOREIGN KEY
	details-1	VARCHAR	50	
<b>Process_acct</b>	acct_no	INT	4	FOREIGN KEY
	process_id	INT	4	FOREIGN KEY
	details-3	VARCHAR	50	
<b>order</b>	customer_name	VARCHAR	50	FOREIGN KEY
	assembly_id	INT	4	FOREIGN KEY
	date_ordered	DATETIME	8	
<b>update record</b>	transaction_no	INT	4	FOREIGN KEY
	account_no	INT	4	FOREIGN KEY
	job_no	INT	4	FOREIGN KEY
	transaction_no	INT	4	FOREIGN KEY

The reference for the bytes sizes of this table came from <https://www.connectionstrings.com/sql-server-data-types-reference/>.

**Task 3.****3.1. Discussion of storage structures for tables**

Table Name	Query # and Type	Search Key	Query Frequency	Selected File Organization	Justifications
<b>Customer</b>	#1 insertion #3 insertion #13 range search	category	30/day 40/day 100/day	index sequential file or B+ tree index with search key category	the range search is the most frequent queries
<b>Assembly</b>	#3 insertion #5 insertion #6 insertion #9 random search #11 range search #12 random search	assembly_id date_commenced date_completed	40/day 10/day 50/day 200/day 100/day 20/day	index sequential file or B+ tree index with search key assembly_id	the random search with assembly_id search key is the most frequent queries
<b>order</b>	#3 insertion		40/day	index sequential file	faster insertion
<b>Process</b>	#4 insertion #5 insertion #6 insertion #11 range search	date_ordered	infrequent 10/day 50/day 100/day	index sequential file or B+ tree index with search key date_ordered	the range search is the most frequent queries
<b>Process_fit</b>	#4 insertion		infrequent	index sequential file	faster insertion
<b>Process_paint</b>	#4 insertion		infrequent	index sequential file	faster insertion
<b>Process_cut</b>	#4 insertion		infrequent	index sequential file	faster insertion
<b>Department</b>	#2 insertion #4 insertion #5 insertion #10 range search #11 random search #12 range search	date_completed process_id date_completed	infrequent infrequent 10/day 20/day 100/day 20/day	index sequential file or B+ tree index with search key process_id	the random search with process_no search key is the most frequent queries
<b>Job</b>	#6 insertion #7 insertion #10 range search #12 range search	date_completed date_completed	50/day 50/day 20/day 20/day	index sequential file or B+ tree index with search key date_completed	although insertion is more frequent but by indexing it, it will be easier to search by the date_completed when doing the range search queries
<b>Job_fit</b>					
<b>Job_paint</b>	#15 random search	color	1/week	B+ tree index with search key color	faster color search when needed
<b>Job_cut</b>	#13 deletion	job_no	1/month	B+ tree index with search key job_no	faster deletion when needed
<b>Transaction</b>	#8 insertion		50/day	index sequential file	faster insertion
<b>Account</b>	#5 insertion		10/day	index sequential file	faster insertion
<b>Dept_acct</b>	#5 insertion #8 random search	account_no	10/day 50/day	index sequential file or B+ tree index with search key account_no	the random search with account_no search key is the

					most frequent queries
<b>Assembly_a cct</b>	#5 insertion		10/day	index sequential file or B+ tree index with search key account_no	the random search with account_no search key is the most frequent queries
	#8 insertion	account_no	50/day		
<b>Process_acc t</b>	#5 insertion		10/day	index sequential file or B+ tree index with search key account_no	the random search with account_no search key is the most frequent queries
	#8 random search	account_no	50/day		
<b>record</b>	#8 insertion		50/day	index sequential file	faster insertion
<b>update</b>				index sequential file or B+ tree index with search key account_no	the random search with account_no search key is the most frequent queries
	#8 random search	account_no	50/day		
<b>track_dept_ expenses</b>	#8 insertion		50/day	index sequential file	faster insertion
<b>track_asse mbly_expen ses</b>	#8 insertion		50/day	index sequential file	faster insertion
<b>track_proce ss_expenses</b>	#8 insertion		50/day	index sequential file	faster insertion

### 3.2. Discussion of storage structures for tables (Azure SQL Database)

The storage structures for tables in the Azure SQL Database for tables with the range search and random search queries will be using the indexed of the search keys that was mentioned in part 3.1 respectively while the other will be inserted in sequential manners for faster result. The exception on the Process table, since the date ordered is not linked to the Process table directly, it will not be able to be indexed by the Process table but by using the Order relation table.



## Task 4. SQL statements and screenshots showing the creation of tables in Azure SQL Database

```

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db Explain Enable SQLCMD
1  /*
2  | Setup new tables for our sets and relations
3  | */
4  -- Delete the table if it exists
5  /*
6  DROP TABLE Customer;
7  DROP TABLE tblAssembly;
8
9  DROP TABLE Process;
10 DROP TABLE Process_fit;
11 DROP TABLE Process_cut;
12 DROP TABLE Process_paint;
13
14 DROP TABLE Department;
15
16 DROP TABLE Job;
17 DROP TABLE Job_fit;
18 DROP TABLE Job_cut;
19 DROP TABLE Job_paint;
20
21 DROP TABLE tblTransaction;
22
23 DROP TABLE Account;
24 DROP TABLE dept_acct;
25 DROP TABLE assembly_acct;
26 DROP TABLE process_acct;
27
28 DROP TABLE orderRel;
29 DROP TABLE updateRel;
30 DROP TABLE recordRel;
31 */
32 -- Create Customer table
33 CREATE TABLE Customer
34 | (cust_name VARCHAR(24),
35 |  cust_address VARCHAR(50),
36 |  category INT,
37 |  CONSTRAINT category CHECK (category BETWEEN 1 AND 10),
38 |  PRIMARY KEY(cust_name));
39 CREATE INDEX cust_index ON Customer (category)
40
41 -- Create tblAssembly table
42 CREATE TABLE tblAssembly
43 | (assembly_id INT,
44 |  assembly_data VARCHAR(50)
45 |  PRIMARY KEY(assembly_id));
46 CREATE INDEX assembly_index ON tblAssembly (assembly_id)
47
48 -- Create Process table
49 CREATE TABLE Process
50 | (process_id INT,
51 |  process_data VARCHAR(50),
52 |  PRIMARY KEY(process_id));
53
54 -- Create Process fit table
55 CREATE TABLE Process_fit
56 | (process_id INT,
57 |  process_data VARCHAR(50),
58 |  FOREIGN KEY(process_id) REFERENCES Process (process_id));
59
60 -- Create Process cut table
61 CREATE TABLE Process_cut
62 | (process_id INT,
63 |  cutting_type VARCHAR(50),
64 |  machine_type VARCHAR(50),
65 |  FOREIGN KEY(process_id) REFERENCES Process (process_id));
66
67 -- Create Process paint table
68 CREATE TABLE Process_paint
69 | (process_id INT,
70 |  paint_type VARCHAR(50),
71 |  paint_method VARCHAR(50),
72 |  FOREIGN KEY(process_id) REFERENCES Process (process_id));
73
74
75 -- Create Department table
76 CREATE TABLE Department
77 | (dept_no INT,
78 |  dept_data VARCHAR(50),
79 |  process_id INT,
80 |  PRIMARY KEY(dept_no),
81 |  FOREIGN KEY(process_id) REFERENCES Process (process_id));
82 -- create index for dept table
83 CREATE INDEX dept_index ON Process (process_id)
84
85 -- Create Job table
86 CREATE TABLE Job
87 | (job_no INT,
88 |  assembly_id INT,
89 |  process_id INT,
90 |  date_commenced DATETIME,
91 |  date_completed DATETIME,
92 |  PRIMARY KEY(job_no),
93 |  FOREIGN KEY(assembly_id) REFERENCES tblAssembly (assembly_id),
94 |  FOREIGN KEY(process_id) REFERENCES Process (process_id));
95 -- create index for job table
96 CREATE INDEX job_index ON Job (date_completed)
97

```

```

98 -- Create Job table
99 CREATE TABLE Job_fit
100     (job_no INT,
101      assembly_id INT,
102      process_id INT,
103      labor_time DECIMAL,
104      date_completed DATETIME,
105      FOREIGN KEY(job_no) REFERENCES Job (job_no),
106      FOREIGN KEY(assembly_id) REFERENCES tblAssembly (assembly_id),
107      FOREIGN KEY(process_id) REFERENCES Process (process_id));
108
109 -- Create Job table
110 CREATE TABLE Job_cut
111     (job_no INT,
112      assembly_id INT,
113      process_id INT,
114      machine_used VARCHAR(50),
115      time_machine_used VARCHAR(50),
116      material_used VARCHAR(50),
117      labor_time DECIMAL,
118      FOREIGN KEY(job_no) REFERENCES Job (job_no),
119      FOREIGN KEY(assembly_id) REFERENCES tblAssembly (assembly_id),
120      FOREIGN KEY(process_id) REFERENCES Process (process_id));
121 CREATE INDEX jobCut_index ON Job_cut (job_no)
122
123 -- Create Job table
124 CREATE TABLE Job_paint
125     (job_no INT,
126      assembly_id INT,
127      process_id INT,
128      color VARCHAR(20),
129      volume DECIMAL,
130      labor_time DECIMAL,
131      FOREIGN KEY(job_no) REFERENCES Job (job_no),
132      FOREIGN KEY(assembly_id) REFERENCES tblAssembly (assembly_id),
133      FOREIGN KEY(process_id) REFERENCES Process (process_id));
134 CREATE INDEX jobPaint_index ON Job_paint (color)
135
136 -- Create Transaction table
137 CREATE TABLE tblTransaction
138     (transaction_no INT,
139      job_no INT,
140      given_cost DECIMAL,
141      PRIMARY KEY(transaction_no),
142      FOREIGN KEY(job_no) REFERENCES Job (job_no));
143
144 -- Create Account table
145 CREATE TABLE Account
146     (account_no INT,
147      date_established DATETIME,
148      PRIMARY KEY(account_no));
149
150 -- Create Department Account table
151 CREATE TABLE dept_acct
152     (account_no INT,
153      dept_no INT,
154      details_2 VARCHAR(50),
155      FOREIGN KEY(dept_no) REFERENCES Department (dept_no),
156      FOREIGN KEY(account_no) REFERENCES Account (account_no));
157
158 -- Create Assembly Account table
159 CREATE TABLE assembly_acct
160     (account_no INT,
161      assembly_id INT,
162      details_1 VARCHAR(50),
163      FOREIGN KEY(assembly_id) REFERENCES tblAssembly (assembly_id),
164      FOREIGN KEY(account_no) REFERENCES Account (account_no));
165
166 -- Create Process Account table
167 CREATE TABLE process_acct
168     (account_no INT,
169      process_id INT,
170      details_3 VARCHAR(50),
171      FOREIGN KEY(process_id) REFERENCES Process (process_id),
172      FOREIGN KEY(account_no) REFERENCES Account (account_no));
173
174 -- Create order relation table
175 CREATE TABLE orderRel
176     (cust_name VARCHAR(24),
177      assembly_id INT,
178      date_ordered DATETIME,
179      FOREIGN KEY(cust_name) REFERENCES Customer (cust_name),
180      FOREIGN KEY(assembly_id) REFERENCES tblAssembly (assembly_id));
181
182 -- Create update relation table
183 CREATE TABLE updateRel
184     (transaction_no INT,
185      account_no INT,
186      FOREIGN KEY(transaction_no) REFERENCES tblTransaction (transaction_no),
187      FOREIGN KEY(account_no) REFERENCES Account (account_no));
188
189 -- Create record relation table
190 CREATE TABLE recordRel
191     (job_no INT,

```

```
190 -- Create record relation table
191 CREATE TABLE recordRel
192     (job_no INT,
193      transaction_no INT,
194      FOREIGN KEY(transaction_no) REFERENCES tblTransaction (transaction_no),
195      FOREIGN KEY(job_no) REFERENCES Job (job_no));
196
197 -- create index for process range search on the date_ordered of order relations table
198 CREATE INDEX process_index ON orderRel (date_ordered)
199
200 /*
201  Add data from the given PDF into their respective tables
202 */
203 -- Insert data into Customer table
204 INSERT INTO Customer
205     (cust_name, cust_address, category)
206 VALUES
207     ('Abe', '101 N Blvd', 1),
208     ('John', '100 NW 63rd St', 2),
209     ('Jack', '1000 University Dr', 3);
210
211 -- View Customer table
212 SELECT * FROM Customer;
```

Results		Messages	
	cust_name	cust_address	category
1	Abe	101 N Blvd	1
2	Jack	1000 University Dr	3
3	John	100 NW 63rd St	2