

**CS 4513-001 Database Management**

**Fall 2019**

**Dr. Le Gruenwald**

**Treyson Martin**

**112963422**

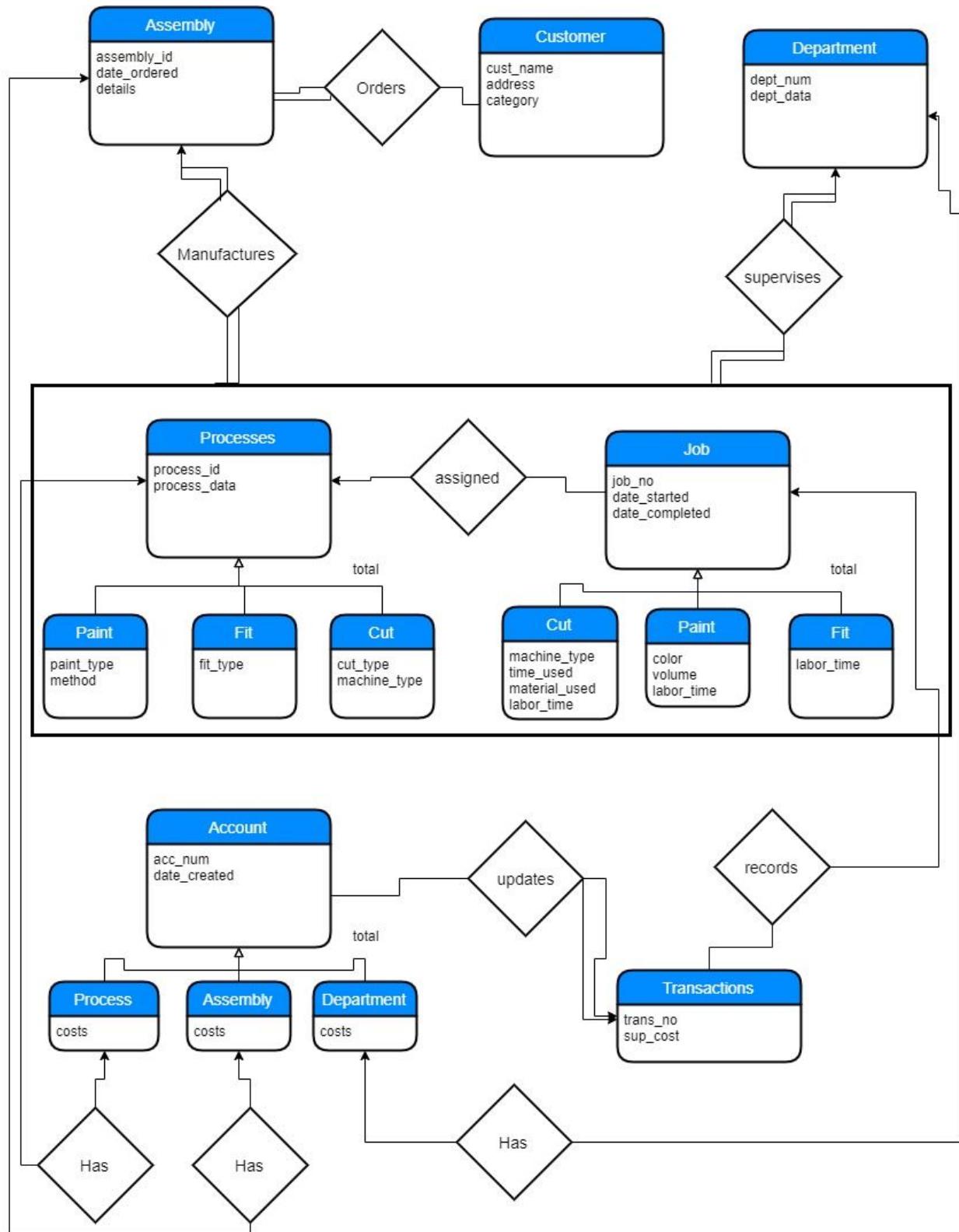
**[Treyson.M.Martin@ou.edu](mailto:Treyson.M.Martin@ou.edu)**

**Process DB Project**

# Table of Contents

<b>Tasks</b>	<b>Page Numbers</b>
1) 1.1 ER Diagram	2
1.2 Relational Schema	3
2) Data Element Dictionary	3 - 6
3) 3.1 File Organization Table	7 - 9
3.2 File Organization Justification	9
4) SQL Statements for Tables	10 - 14
5) Java Application	14 - 33
6) Java Execution	33 - 66
7) Web Application	66 - 77

## 1.1) ER Diagram



## 1.2) Relational Schema

Assembly(assembly\_id, date\_ordered, details)  
 Customer(cust\_name, address, category)  
 Department(dept\_num, dept\_data)  
 Processes(process\_id, process\_data)  
 Process\_Paint(process\_id, process\_data, paint\_type, method)  
 Process\_Fit(process\_id, process\_data, fit\_type)  
 Process\_Cut(process\_id, process\_data, cut\_type, method\_type)  
 Job(job\_no, date\_started, date\_completed)  
 Job\_Cut(job\_no, date\_started, date\_completed, machine\_type, times\_used, material\_used, labor\_time)  
 Job\_Paint(job\_no, date\_started, date\_completed, color, volume, labor\_time)  
 Job\_Fit(job\_no, date\_started, date\_completed, labor\_time)  
 Transaction(trans\_no, sup\_cost)  
 Account(acc\_num, date\_created)  
 Account\_Assembly(acc\_num, date\_created, costs)  
 Account\_Process(acc\_num, date\_created, costs)  
 Account\_Department(acc\_num, date\_created, costs)  
 Orders(cust\_name, assembly\_id)  
 Manufactures(assembly\_id, process\_id, job\_no)  
 Assigned(process\_id, job\_no)  
 Supervises(dept\_name, process\_id, job\_no)  
 Records(job\_no, trans\_no)  
 Updates(trans\_no, acc\_num)  
 Has1(assembly\_id, acc\_num)  
 Has2(process\_id, acc\_num)  
 Has3(dept\_num, acc\_num)

## 2) Data Element Dictionary

Table	Column	Type	Size(Bytes)	Constraints
Assembly	assembly_id	int	4	Primary Key
Assembly	date_ordered	date	3	
Assembly	details	varchar(128)	130	
Customer	cust_name	varchar(32)	34	Primary key
Customer	address	varchar(40)	42	

Customer	category	int	4	Values 1-10
Department	dept_num	int	4	Primary Key
Department	dept_data	varchar(128)	130	
Processes	process_id	int	4	Primary Key
Processes	process_data	varchar(128)	130	
Process_Paint	process_id	int	4	Primary Key
Process_Paint	process_data	varchar(128)	130	
Process_Paint	paint_type	varchar(32)	34	
Process_Paint	method	varchar(32)	34	
Process_Fit	process_id	int	4	Primary Key
Process_Fit	process_data	varchar(128)	130	
Process_Fit	fit_type	varchar(32)	34	
Process_Cut	process_id	int	4	Primary Key
Process_Cut	process_data	varchar(128)	130	
Process_Cut	cut_type	varchar(32)	34	
Process_Cut	machine_type	varchar(32)	34	
Job	job_no	int	4	Primary Key
Job	date_started	date	3	
Job	date_completed	date	3	Date_completed >= date_started
Job_Cut	job_no	int	4	Primary Key
Job_Cut	date_started	date	3	
Job_Cut	date_completed	date	3	Date_completed >= date_started
Job_Cut	machine_type	varchar(32)	34	
Job_Cut	time_used	int	4	Positive number
Job_Cut	material_used	varchar(32)	34	

Job_Cut	labor_time	int	4	Positive number
Job_Paint	job_no	int	4	Primary Key
Job_Paint	date_started	date	3	
Job_Paint	date_completed	date	3	Date_completed >= date_started
Job_Paint	color	varchar(32)	34	
Job_Paint	volume	int	4	Positive number
Job_Paint	labor_time	int	4	Positive number
Job_Fit	job_no	int	4	Primary Key
Job_Fit	date_started	date	3	
Job_Fit	date_completed	date	3	Date_completed >= date_started
Job_Fit	labor_time	int	4	Positive number
Transaction	trans_no	int	4	Primary Key
Transaction	sup_cost	float	8	Positive number
Account	acc_num	int	4	Primary Key
Account	date_created	date	3	
Account_Assembly	acc_num	int	4	Primary Key
Account_Assembly	date_created	date	3	
Account_Assembly	cost(Details1)	float	8	Positive number
Account_Department	acc_num	int	4	Primary Key
Account_Department	date_created	date	3	
Account_Department	cost(Details2)	float	8	Positive number

Account_Process	acc_num	int	4	Primary Key
Account_Process	date_created	date	3	
Account_Process	cost(Details3)	float	8	Positive number
Orders	cust_name	varchar(32)	34	Foreign Key
Orders	assembly_id	int	4	Primary Key
Manufactures	assembly_id	int	4	Foreign Key
Manufactures	process_id	int	4	Primary Key
Manufactures	job_no	int	4	
Assigned	process_id	int	4	Primary Key
Assigned	job_no	int	4	Foreign Key
Supervises	dept_num	int	4	Primary Key
Supervises	process_id	int	4	Foreign Key
Supervises	job_no	int	4	
Records	job_no	int	4	Primary Key
Records	trans_no	int	4	Foreign Key
Updates	trans_no	int	4	Primary Key
Updates	acc_no	int	4	Foreign Key
Has1	assembly_id	int	4	Primary Key
Has1	acc_num	int	4	Foreign Key
Has2	process_id	int	4	Primary Key
Has2	acc_num	int	4	Foreign Key
Has3	dept_num	int	4	Primary Key
Has3	acc_num	int	4	Foreign Key

### 3.1) File Organization

Table	Query # / Type	Search Key	Query Freq	File Organization	Justifications
Assembly	#3 Insertion		40/day	Extendible hashing	Tied to customer_name
Customer	#1 Insertion #13 Range Search	category	30/day 100/day	Heap Index with search key category	Faster cause of frequency Fastest range search.
Department	#2 Insertion #4 Insertion		Infrequent Infrequent	Heap B+ Index tree	Independent on insertion Because insertion is not often
Process_Paint	#4 Insertion		Infrequent	B+ index tree	Because insertion is not often
Process_Fit	#4 Insertion		Infrequent	B+ index tree	Because insertion is not often
Process_Cut	#4 Insertion		Infrequent	B+ index tree	Because insertion is not often
Job_Cut	#6 Insertion  #7 Update #10 Range Search  #14 Delete  #12 Random Search	Job_no Date_completed  Job_no date_completed	50/day  50/day 20/day  1/month  20/day	Extendible hashing  Heap Index with date_completed  Index with job_no Hashing with date_completed	Tied to assembly and process Fast Process. Fastest results.  Fastest way to delete records. Fastest Results
Job_Paint	#6 Insertion		50/day	Extendible	Tied to

	#7 Update #10 Range Search  #15 Update  #12 Random Search	Job_no Date_complet ed  Job_no  date_complet ed	50/day 20/day  1/week  20/day	hashing Heap Index with date_complet ed  Heap  Hashing with key date_complet ed	assembly and process Fast Process. Fastest Results.  Fastest Process  Fastest Results.
Job_Fit	#6 Insertion  #7 Update #10 Range Search  #12 Random Search	Job_no Date_complet ed  date_complet ed	50/day 20/day  20/day	Extendible hashing Heap Index with date_complet ed  Hashing with key date_complet ed	Tied to assembly and process Fast Process. Fastest Results.  Fastest Results.
Transaction	#8 Insertion		50/day	Heap	Needs to go fast and process FIFO
Account_Assembly	#5 Insertion  #9 Random Search  #8 Update		10/day 200/day  50/day	Extendible hashing With hash key assembly_id Heap	Tied to assembly. Fastest search results.  See Updates
Account_Department	#5 Insertion  #8 Update	acc_num	10/day 50/day	Extendible Hashing Heap	Tied to department See Updates
Account_Process	#5 Insertion  #Update	acc_num	10/day 50-day	Extendible Hashing Heap	Tied to Process See Updates
Orders	#3 Insertion		40/day	Extendible hashing	Tied to customer_name

Manufactures	#6 Insertion #12 Random Search	job_no	50/day 20/day	Extendible hashing Hashing with job_no	Tied to assembly and process. Fastest results.
Assigned	#6 Insertion		50/day	Extendible hashing	Tied to assembly and process
Supervises	#4 Insertion #10 Range Search #12 Random Search	Date_comple ted dept_num	Infrequent 20/day 20/day	B+ index tree Index with date_complet ed Hashing with key dept_num	Because insertion is not often Fastest results. Fastest results.
Records	#8 Insertion		50/day	Heap	Needs to go fast and process FIFO
Updates	#8 Insertion		50/day	Heap	Needs to go fast and process FIFO
Has1	#5 Insertion #9 Random Search	assembly_id	10/day 200/day	Extendible hashing Hashing with key assembly_id	Tied to assembly. Fastest search results.
Has2	#5 Insertion		10/day	Extendible Hashing	Tied to process
Has3	#5 Insertion		10/day	Extendible Hashing	Tied to department

### 3.2) File Organization.

This is a bit tricky. For the most part it looks like storage implementation will be within the relational table. The biggest issue is the relations ie(Manufactures) and making sure these get populated. Normally I would just implement foreign keys within the other tables like Customer and etc. Since then you don't have that middle table that can sometimes muddy stuff up. Some tables changed to match the 3.1 from the pictures in the create tables this is to allow proper insertion / updates because of the relationships between tables.

## 4) SQL Statements

**SQL Query 1 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio**

```

CREATE TABLE Assembly
(
    assembly_id int PRIMARY KEY,
    date_ordered date,
    details varchar(128)
);

CREATE TABLE Customers
(
    cust_name varchar(32) PRIMARY KEY,
    address varchar(40),
    category int,
    CHECK (category>0),
    CHECK (category<11)
);

CREATE TABLE Department
(
    dept_num int PRIMARY KEY,
    dept_data varchar(128)
);

CREATE TABLE Processes
(
    process_id int PRIMARY KEY,
    process_data varchar(128)
);

```

**Messages**

8:22:01 AM Started executing query at Line 1  
Commands completed successfully.  
Total execution time: 00:00:00.610

**SQL Query 1 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio**

```

CREATE TABLE Process_Paint
(
    process_id int PRIMARY KEY,
    process_data varchar(128),
    paint_type varchar(32),
    method varchar(32)
);

CREATE TABLE Process_Fit
(
    process_id int PRIMARY KEY,
    process_data varchar(128),
    fit_type varchar(32)
);

CREATE TABLE Process_Cut
(
    process_id int PRIMARY KEY,
    process_data varchar(128),
    cut_type varchar(32),
    machine_type varchar(32)
);

CREATE TABLE Job
(
    job_no int PRIMARY KEY,
    date_started date,
    date_completed date,
    duration int
);

```

**Messages**

8:23:23 AM Started executing query at Line 29  
Commands completed successfully.  
Total execution time: 00:00:00.056

File Edit View Help

• SQLQuery\_1 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio

CONNECTIONS 3\_Problem2\_Group39.sql - disconnected mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db SQLQuery\_1 - mart34...rt3422

**SERVERS**

- > mart3422-sql-server.database.windows.net
- > Tables
- > Views
- > Synonyms
- > Programmability
- > External Resources
- > Storage
- > Security

CREATE TABLE Job

```

52 (
53     job_no int PRIMARY KEY,
54     date_started date,
55     date_completed date,
56     CHECK (date_completed >= date_started)
57 );
58
59 CREATE TABLE Job_Cut
60 (
61     job_no int PRIMARY KEY,
62     date_started date,
63     date_completed date,
64     machine_type varchar(32),
65     time_used int,
66     material_used varchar(32),
67     labor_time int,
68     CHECK (date_completed >= date_started),
69     CHECK (labor_time > 0),
70     CHECK (time_used > 0)
71 );
72
73 CREATE TABLE Job_Paint
74 (
75     job_no int PRIMARY KEY,
76     date_started date,
77     date_completed date,
78     color varchar(32),
79
80     volume int,
81     labor_time int,
82     CHECK (date_completed >= date_started),
83     CHECK (volume >= 0),
84     CHECK (labor_time >= 0)
85 );
86
87 CREATE TABLE Job_Fit
88 (
89     job_no int PRIMARY KEY,
90     date_started date,
91     date_completed date,
92     labor_time int,
93     CHECK (date_completed >= date_started),
94     CHECK (labor_time >= 0)
95 );
96
97 CREATE TABLE [transaction]
98 (
99     trans_no int PRIMARY KEY,
100    sup_cost float,
101    CHECK (sup_cost >= 0)
102 );

```

Messages

8:23:49 AM Started executing\_query\_at Line 52  
Commands completed successfully.  
Total execution time: 00:00:00.040

LN 73, Col 1 (421 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 0 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 823 AM 11/5/2019

Type here to search

File Edit View Help

• SQLQuery\_1 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio

CONNECTIONS 3\_Problem2\_Group39.sql - disconnected mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db SQLQuery\_1 - mart34...rt3422

**SERVERS**

- > mart3422-sql-server.database.windows.net
- > Tables
- > Views
- > Synonyms
- > Programmability
- > External Resources
- > Storage
- > Security

CREATE TABLE Job\_Paint

```

74 (
75     job_no int PRIMARY KEY,
76     date_started date,
77     date_completed date,
78     color varchar(32),
79     volume int,
80     labor_time int,
81     CHECK (date_completed >= date_started),
82     CHECK (volume >= 0),
83     CHECK (labor_time >= 0)
84 );
85
86
87 CREATE TABLE Job_Fit
88 (
89     job_no int PRIMARY KEY,
90     date_started date,
91     date_completed date,
92     labor_time int,
93     CHECK (date_completed >= date_started),
94     CHECK (labor_time >= 0)
95 );
96
97 CREATE TABLE [transaction]
98 (
99     trans_no int PRIMARY KEY,
100    sup_cost float,
101    CHECK (sup_cost >= 0)
102 );

```

Messages

8:24:13 AM Started executing\_query\_at Line 74  
Commands completed successfully.  
Total execution time: 00:00:00.070

LN 97, Col 1 (434 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 0 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 824 AM 11/5/2019

Type here to search

File Edit View Help

• SQLQuery\_1 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio

CONNECTIONS 3\_Problem2\_Group39.sql - disconnected mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db SQLQuery\_1 - mart34...rt3422

**SERVERS**

- > mart3422-sql-server.database.windows.net
- > Tables
- > Views
- > Synonyms
- > Programmability
- > External Resources
- > Storage
- > Security

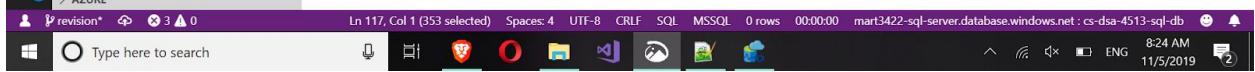
```

95 );
96
97 CREATE TABLE [Transaction]
98 (
99     trans_no int PRIMARY KEY,
100    sup_cost float,
101    CHECK (sup_cost >= 0)
102 );
103
104 CREATE TABLE Account
105 (
106     acc_num int PRIMARY KEY,
107     date_created date DEFAULT GETDATE()
108 );
109
110 CREATE TABLE Account_Assembly
111 (
112     acc_num int PRIMARY KEY,
113     date_created date DEFAULT GETDATE(),
114     cost float DEFAULT 0,
115     CHECK (cost >= 0),
116 );
117
118 CREATE TABLE Account_Department
119 (
120     acc_num int PRIMARY KEY,
121     date_created date DEFAULT GETDATE(),
122     cost float DEFAULT 0,
123     CHECK (cost >= 0),
124 );
125
126 CREATE TABLE Account_Process
127 (
128     acc_num int PRIMARY KEY,
129     date_created date DEFAULT GETDATE(),
130     cost float DEFAULT 0,
131     CHECK (cost >= 0),
132 );
133
134 CREATE TABLE Manufactures
135 (
136     assembly_id int PRIMARY KEY,
137     process_id int FOREIGN KEY REFERENCES Processes(process_id),
138     job_no int
139 );
140
141 CREATE TABLE Assigned
142 (
143     process_id int PRIMARY KEY,
144     job_no int FOREIGN KEY REFERENCES Job(job_no)
145 );

```

Messages

8:24:46 AM Started executing\_query\_at Line 97  
Commands completed successfully.  
Total execution time: 00:00:00.060



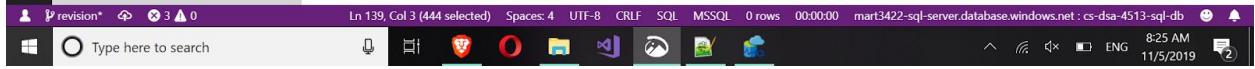
```

118 CREATE TABLE Account_Department
119 (
120     acc_num int PRIMARY KEY,
121     date_created date DEFAULT GETDATE(),
122     cost float DEFAULT 0,
123     CHECK (cost >= 0),
124 );
125
126 CREATE TABLE Account_Process
127 (
128     acc_num int PRIMARY KEY,
129     date_created date DEFAULT GETDATE(),
130     cost float DEFAULT 0,
131     CHECK (cost >= 0),
132 );
133
134 CREATE TABLE Manufactures
135 (
136     assembly_id int PRIMARY KEY,
137     process_id int FOREIGN KEY REFERENCES Processes(process_id),
138     job_no int
139 );
140
141 CREATE TABLE Assigned
142 (
143     process_id int PRIMARY KEY,
144     job_no int FOREIGN KEY REFERENCES Job(job_no)
145 );

```

Messages

8:25:16 AM Started executing\_query\_at Line 118  
Commands completed successfully.  
Total execution time: 00:00:00.048



A screenshot of the Azure Data Studio interface. The left sidebar shows a tree view of 'SERVERS' containing various database objects like tables (e.g., dbo.Account\_Process, dbo.Acted, dbo.Assembly, etc.) and system objects (e.g., Views, Synonyms, Programmability). The main pane displays a SQL script for creating tables. The script includes several CREATE TABLE statements:

```
140
141 CREATE TABLE Assigned
142 (
143     process_id int PRIMARY KEY,
144     job_no int FOREIGN KEY REFERENCES Job(job_no)
145 );
146
147 CREATE TABLE Supervises
148 (
149     dept_num int PRIMARY KEY,
150     process_id int FOREIGN KEY REFERENCES Processes(process_id),
151     job_no int
152 );
153
154 CREATE TABLE Records
155 (
156     job_no int PRIMARY KEY,
157     trans_no int FOREIGN KEY REFERENCES [Transaction](trans_no)
158 );
159
160 CREATE TABLE Updates
161 (
162     trans_no int PRIMARY KEY,
163     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
164 );
165
166 CREATE TABLE Has1
```

The 'Messages' section at the bottom shows the execution status:

8:30:47 AM Started executing query at Line 141  
Commands completed successfully.  
Total execution time: 00:00:00.051

The bottom toolbar includes icons for file operations, search, and various tools. The status bar shows the connection details: mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db, and the date/time: 11/5/2019 8:30 AM.

File Edit View Help

CONNECTIONS 3\_Problem2\_Group39.sql - disconnected mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db (mart3422) - Azure Data Studio SQLQuery\_1 - mart34...rt3422

**SERVERS**

- > dbo.Account\_Process
- > dbo.Acted
- > dbo.Assembly
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Performer
- > dbo.Process\_Cut
- > dbo.Process\_Fit
- > dbo.Process\_Paint
- > dbo.Processes
- > dbo.Transaction
- > Views
- > Synonyms
- > Programmability
- > External Resources
- > Storage
- > Security

AZURE

```

157     trans_no int FOREIGN KEY REFERENCES [Transaction](trans_no)
158 );
159
160 CREATE TABLE Updates
161 (
162     trans_no int PRIMARY KEY,
163     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
164 );
165
166 CREATE TABLE Has1
167 (
168     assembly_id int PRIMARY KEY,
169     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
170 );
171
172 CREATE TABLE Has2
173 (
174     process_id int PRIMARY KEY,
175     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
176 );
177
178 CREATE TABLE Has3
179 (
180     dept_num int PRIMARY KEY,
181     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
182 );

```

Messages

8:31:15 AM Started executing query at Line 166  
commands completed successfully.  
Total execution time: 00:00:00.047

Ln 182, Col 3 (331 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 0 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 8:31 AM 11/5/2019

Type here to search

File Edit View Help

CONNECTIONS Group39.sql - disconnected mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db (mart3422) - Azure Data Studio SQLQuery\_1 - mart34...rt3422

**SERVERS**

- > dbo.Account
- > dbo.Account\_Assembly
- > dbo.Account\_Department
- > dbo.Account\_Process
- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Performer
- > dbo.Process\_Cut
- > dbo.Process\_Fit
- > dbo.Process\_Paint

AZURE

```

157     trans_no int FOREIGN KEY REFERENCES [Transaction](trans_no)
158 );
159
160 CREATE TABLE Updates
161 (
162     trans_no int PRIMARY KEY,
163     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
164 );
165
166 CREATE TABLE Has1
167 (
168     assembly_id int PRIMARY KEY,
169     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
170 );
171
172 CREATE TABLE Has2
173 (
174     process_id int PRIMARY KEY,
175     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
176 );
177
178 CREATE TABLE Has3
179 (
180     dept_num int PRIMARY KEY,
181     acc_num int FOREIGN KEY REFERENCES Account(acc_num)
182 );

```

Messages

8:31:15 AM Started executing query at Line 166  
Commands completed successfully.  
Total execution time: 00:00:00.047

Ln 175, Col 56 Spaces: 4 UTF-8 CRLF SQL MSSQL 0 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 8:32 AM 11/5/2019

Type here to search

## 5) Java Application

Project - Project\_Shell/src/Program.java - Eclipse IDE

```

2* * This program is a shell for the CS4513 Individual Project
3
4import java.io.BufferedReader;
5import java.io.BufferedWriter;
6import java.io.File;
7import java.io.FileReader;
8import java.io.FileWriter;
9import java.io.PrintWriter;
10import java.sql.*;
11import java.util.*;
12import java.util.Date;
13
14public class Program
15{
16    // Connect to database
17    final static String hostName = "mart3422-sql-server.database.windows.net";
18    final static String dbName = "cs-dsa-4513-sql-db";
19    final static String user = "mart3422";
20    final static String password = "Atticus666418!";
21    final static String url =
22        String.format("jdbc:sqlserver://:%s:1433;database=%s;user=%s;password=%s;encrypt=true",
23                      dbName, user, password);
24
25    public static void main(String[] args)
26    {
27        //Display welcome message and call method that prints out options.
28        System.out.println("Welcome to JS Accounting System.");
29        System.out.println("Below are options 1-18 that you can use.");
30    }
31}

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Project - Project\_Shell/src/Program.java - Eclipse IDE

```

36    System.out.println("Below are options 1-18 that you can use.");
37
38    Scanner scanner = new Scanner(System.in);
39    int option = 0;
40    do
41    {
42        DisplayOptions();
43
44        option = scanner.nextInt();
45        scanner.nextLine();
46
47        switch(option)
48        {
49            case 1:
50                InsertCustomer(0);
51                break;
52            case 2:
53                InsertDepartment(0);
54                break;
55            case 3:
56                InsertAssemblyInfo();
57                break;
58            case 4:
59                InsertProcessInfo();
60                break;
61            case 5:
62                CreateAccount();
63                break;
64            case 6:
65                InsertJobInfo();
66        }
67    }
68
69    while(option != 0);
70
71    System.out.println("Thank you for using the JS Accounting System!");
72}

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Task List    Quick Access

Program.java

```

65         InsertJobInfo();
66         break;
67     case 7:
68         ConfirmJobCompletion();
69         break;
70     case 8:
71         InsertTransactionInfo();
72         break;
73     case 9:
74         RetrieveAssemblyCost();
75         break;
76     case 10:
77         RetrieveDeptLaborTime();
78         break;
79     case 11:
80         RetrieveProcesses();
81         break;
82     case 12:
83         RetrieveJobsCompleted();
84         break;
85     case 13:
86         RetrieveCustomers();
87         break;
88     case 14:
89         DeleteCutJobs();
90         break;
91     case 15:
92         UpdatePaintJob();
93         break;
94     case 16:

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 94 : 25 : 2285

Type here to search

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Task List    Quick Access

Program.java

```

94         case 16:
95             ImportExportCust(0);
96             break;
97         case 17:
98             ImportExportCust(1);
99             break;
100        case 18: //do nothing and break out of loop.
101           break;
102       default:
103           System.out.println("Invalid Option.\nSelect from Option below.");
104       }
105     }
106   }
107   while(option != 18);
108
109   //Ending program.
110   System.out.println("Exiting...\nHave a good day!");
111   System.exit(0);
112 }
//option 16 and 17
public static void ImportExportCust(int flag)
{
    String line = "";
    String splitBy = ","; //for csv
    String userFile = "";
    Scanner sc = new Scanner(System.in);
    try
    {
        Connection connection = DriverManager.getConnection(url); //sql connection.
        Statement statement = connection.createStatement();

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 123 : 25 : 2976

Type here to search

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

123     Statement statement = connection.createStatement();
124
125     if(flag == 0)
126     {
127         System.out.print("Input file name (Include .csv): ");
128         userFile = sc.nextLine();
129         File file = new File(userFile);
130         BufferedReader br = new BufferedReader(new FileReader(file));
131         while((line = br.readLine()) != null)//goes to EOF
132         {
133             String[] customers = line.split(splitBy);
134             //System.out.println("Customer: Name = " + customers[0] + " Address
135             //customers[1] + " Category: " + customers[2]); //debug.
136
137             String sql = "INSERT INTO Customers VALUES ('"
138             + customers[0] + "','" +
139             + customers[1] + "','" +
140             + customers[2] + ")";
141
142             //debug.
143             //System.out.println(sql);
144             statement.execute(sql);
145         }
146         System.out.println("Import Successful");
147         br.close();
148     }
149     else if(flag == 1)
150     {
151         System.out.print("Enter lower category: ");
152         int lowCat = sc.nextInt();
153
154         int highCat = sc.nextInt();
155         sc.nextLine();
156
157         String sql = "SELECT * FROM Customers WHERE"
158             + " category BETWEEN " + lowCat + " AND " + highCat
159             + " ORDER by cust_name";
160
161         ResultSet rs = statement.executeQuery(sql);
162         int randomNum = (int)(Math.random() * 100000 + 1); //generate random number
163         String custfile = "customers" + randomNum + ".csv";
164         File f1 = new File(custfile);
165
166         BufferedWriter writer = new BufferedWriter(new FileWriter(f1, true));
167         writer.write("Name,Address,category\n");
168         while(rs.next())
169         {
170             writer.append(rs.getString("cust_name"));
171             writer.append(",");
172             writer.append(rs.getString("address"));
173             writer.append(",");
174             int category = rs.getInt("category");
175             String ct = "" + category;
176             writer.append(ct);
177             writer.append("\n");
178         }
179         writer.close();
180         System.out.println("File " + custfile + " created.");
181

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search    Writable Smart Insert 152 : 5 : 3890

240 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

152         int lowCat = sc.nextInt();
153         sc.nextLine();
154         System.out.print("Enter high category: ");
155         int highCat = sc.nextInt();
156         sc.nextLine();
157
158         String sql = "SELECT * FROM Customers WHERE"
159             + " category BETWEEN " + lowCat + " AND " + highCat
160             + " ORDER by cust_name";
161
162         ResultSet rs = statement.executeQuery(sql);
163         int randomNum = (int)(Math.random() * 100000 + 1); //generate random number
164         String custfile = "customers" + randomNum + ".csv";
165         File f1 = new File(custfile);
166
167         BufferedWriter writer = new BufferedWriter(new FileWriter(f1, true));
168         writer.write("Name,Address,category\n");
169         while(rs.next())
170         {
171             writer.append(rs.getString("cust_name"));
172             writer.append(",");
173             writer.append(rs.getString("address"));
174             writer.append(",");
175             int category = rs.getInt("category");
176             String ct = "" + category;
177             writer.append(ct);
178             writer.append("\n");
179         }
180         writer.close();
181         System.out.println("File " + custfile + " created.");

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search    Writable Smart Insert 181 : 53 : 4918

240 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Program.java

```

181         System.out.println("File " + custfile + " created.");
182     }
183   } catch(Exception ex)
184   {
185     System.out.println("Error in ImportExportCust. Error: " + ex.toString());
186   }
187 }
188 }
189 //Option 15 Update color on Paint Job.
190 public static void UpdatePaintJob()
191 {
192   try
193   {
194     Connection connection = DriverManager.getConnection(url); //sql connection.
195     Scanner sc = new Scanner(System.in);
196     System.out.print("Enter Job No: ");
197     int jobNo = sc.nextInt();
198     sc.nextLine();
199     System.out.print("Update Color of Job: ");
200     String color = sc.nextLine();
201
202     String sql = "UPDATE Job_Paint SET color = '" + color
203                 + "' WHERE job_no = " + jobNo;
204
205     Statement statement = connection.createStatement();
206     statement.execute(sql);
207     System.out.println("Update Complete..");
208   }
209   catch(Exception ex)
210   {
211     System.out.println("Error in UpdatePaintJob(). Error: " + ex.toString());
212   }
213 }
214
215 //Option 14 Delete Cut jobs based on date range.
216 public static void DeleteCutJobs()
217 {
218   try
219   {
220     Connection connection = DriverManager.getConnection(url); //sql connection.
221     Scanner sc = new Scanner(System.in);
222     System.out.print("Enter low job no: ");
223     int lowJobNo = sc.nextInt();
224     sc.nextLine(); //clear buffer
225
226     System.out.print("Enter high job no: ");
227     int highJobNo = sc.nextInt();
228     sc.nextLine(); //clear buffer.
229
230     String sql = "Delete from Job_Cut where job_no BETWEEN "
231                 + lowJobNo + " AND " + highJobNo;
232
233     //debug.
234     //System.out.println(sql);
235
236     Statement statement = connection.createStatement();
237     statement.execute(sql);
238     System.out.println("Deletions Complete..");
239   }
240   catch(Exception ex)
241   {
242     System.out.println("Error in DeleteCutJobs(). Error: " + ex.toString());
243   }
244 }
245
246 //Option 13 Insert Cut Jobs
247 public static void InsertCutJobs()
248 {
249   try
250   {
251     Connection connection = DriverManager.getConnection(url); //sql connection.
252     Scanner sc = new Scanner(System.in);
253     System.out.print("Enter Job No: ");
254     int jobNo = sc.nextInt();
255     sc.nextLine();
256
257     System.out.print("Enter Cut Type: ");
258     String cutType = sc.nextLine();
259
260     String sql = "Insert into Job_Cut values (" + jobNo + ", "
261                 + cutType + ")";
262
263     Statement statement = connection.createStatement();
264     statement.execute(sql);
265     System.out.println("Insert Complete..");
266   }
267   catch(Exception ex)
268   {
269     System.out.println("Error in InsertCutJobs(). Error: " + ex.toString());
270   }
271 }
272
273 //Option 12 Retrieve Cut Jobs
274 public static void RetrieveCutJobs()
275 {
276   try
277   {
278     Connection connection = DriverManager.getConnection(url); //sql connection.
279     Statement statement = connection.createStatement();
280     ResultSet rs = statement.executeQuery("Select * from Job_Cut");
281
282     while(rs.next())
283     {
284       System.out.println("Job No: " + rs.getInt("job_no") + " Cut Type: " + rs.getString("cut_type"));
285     }
286   }
287   catch(Exception ex)
288   {
289     System.out.println("Error in RetrieveCutJobs(). Error: " + ex.toString());
290   }
291 }
292
293 //Option 11 Retrieve All Cut Jobs
294 public static void RetrieveAllCutJobs()
295 {
296   try
297   {
298     Connection connection = DriverManager.getConnection(url); //sql connection.
299     Statement statement = connection.createStatement();
300     ResultSet rs = statement.executeQuery("Select * from Job_Cut");
301
302     while(rs.next())
303     {
304       System.out.println("Job No: " + rs.getInt("job_no") + " Cut Type: " + rs.getString("cut_type"));
305     }
306   }
307   catch(Exception ex)
308   {
309     System.out.println("Error in RetrieveAllCutJobs(). Error: " + ex.toString());
310   }
311 }
312
313 //Option 10 Confirm Job Completion
314 public static void ConfirmJobCompletion()
315 {
316   try
317   {
318     Connection connection = DriverManager.getConnection(url); //sql connection.
319     Scanner sc = new Scanner(System.in);
320     System.out.print("Enter Job No: ");
321     int jobNo = sc.nextInt();
322     sc.nextLine();
323
324     String sql = "Update Job_Cut Set completed = 1 where job_no = " + jobNo;
325
326     Statement statement = connection.createStatement();
327     statement.execute(sql);
328     System.out.println("Job Completed..");
329   }
330   catch(Exception ex)
331   {
332     System.out.println("Error in ConfirmJobCompletion(). Error: " + ex.toString());
333   }
334 }
335
336 //Option 9 Insert Transaction Info
337 public static void InsertTransactionInfo()
338 {
339   try
340   {
341     Connection connection = DriverManager.getConnection(url); //sql connection.
342     Scanner sc = new Scanner(System.in);
343     System.out.print("Enter Job No: ");
344     int jobNo = sc.nextInt();
345     sc.nextLine();
346
347     System.out.print("Enter Transaction Type: ");
348     String transType = sc.nextLine();
349
350     String sql = "Insert into Transaction_Info values (" + jobNo + ", "
351                 + transType + ")";
352
353     Statement statement = connection.createStatement();
354     statement.execute(sql);
355     System.out.println("Insert Complete..");
356   }
357   catch(Exception ex)
358   {
359     System.out.println("Error in InsertTransactionInfo(). Error: " + ex.toString());
360   }
361 }
362
363 //Option 8 Delete Transaction Info
364 public static void DeleteTransactionInfo()
365 {
366   try
367   {
368     Connection connection = DriverManager.getConnection(url); //sql connection.
369     Scanner sc = new Scanner(System.in);
370     System.out.print("Enter Job No: ");
371     int jobNo = sc.nextInt();
372     sc.nextLine();
373
374     String sql = "Delete from Transaction_Info where job_no = " + jobNo;
375
376     Statement statement = connection.createStatement();
377     statement.execute(sql);
378     System.out.println("Delete Complete..");
379   }
380   catch(Exception ex)
381   {
382     System.out.println("Error in DeleteTransactionInfo(). Error: " + ex.toString());
383   }
384 }
385
386 //Option 7 Retrieve Transaction Info
387 public static void RetrieveTransactionInfo()
388 {
389   try
390   {
391     Connection connection = DriverManager.getConnection(url); //sql connection.
392     Statement statement = connection.createStatement();
393     ResultSet rs = statement.executeQuery("Select * from Transaction_Info");
394
395     while(rs.next())
396     {
397       System.out.println("Job No: " + rs.getInt("job_no") + " Transaction Type: " + rs.getString("trans_type"));
398     }
399   }
400   catch(Exception ex)
401   {
402     System.out.println("Error in RetrieveTransactionInfo(). Error: " + ex.toString());
403   }
404 }
405
406 //Option 6 Insert Process Info
407 public static void InsertProcessInfo()
408 {
409   try
410   {
411     Connection connection = DriverManager.getConnection(url); //sql connection.
412     Scanner sc = new Scanner(System.in);
413     System.out.print("Enter Job No: ");
414     int jobNo = sc.nextInt();
415     sc.nextLine();
416
417     System.out.print("Enter Process Type: ");
418     String procType = sc.nextLine();
419
420     String sql = "Insert into Process_Info values (" + jobNo + ", "
421                 + procType + ")";
422
423     Statement statement = connection.createStatement();
424     statement.execute(sql);
425     System.out.println("Insert Complete..");
426   }
427   catch(Exception ex)
428   {
429     System.out.println("Error in InsertProcessInfo(). Error: " + ex.toString());
430   }
431 }
432
433 //Option 5 Delete Process Info
434 public static void DeleteProcessInfo()
435 {
436   try
437   {
438     Connection connection = DriverManager.getConnection(url); //sql connection.
439     Scanner sc = new Scanner(System.in);
440     System.out.print("Enter Job No: ");
441     int jobNo = sc.nextInt();
442     sc.nextLine();
443
444     String sql = "Delete from Process_Info where job_no = " + jobNo;
445
446     Statement statement = connection.createStatement();
447     statement.execute(sql);
448     System.out.println("Delete Complete..");
449   }
450   catch(Exception ex)
451   {
452     System.out.println("Error in DeleteProcessInfo(). Error: " + ex.toString());
453   }
454 }
455
456 //Option 4 Retrieve Process Info
457 public static void RetrieveProcessInfo()
458 {
459   try
460   {
461     Connection connection = DriverManager.getConnection(url); //sql connection.
462     Statement statement = connection.createStatement();
463     ResultSet rs = statement.executeQuery("Select * from Process_Info");
464
465     while(rs.next())
466     {
467       System.out.println("Job No: " + rs.getInt("job_no") + " Process Type: " + rs.getString("proc_type"));
468     }
469   }
470   catch(Exception ex)
471   {
472     System.out.println("Error in RetrieveProcessInfo(). Error: " + ex.toString());
473   }
474 }
475
476 //Option 3 Insert DeLabor Info
477 public static void InsertDeLaborInfo()
478 {
479   try
480   {
481     Connection connection = DriverManager.getConnection(url); //sql connection.
482     Scanner sc = new Scanner(System.in);
483     System.out.print("Enter Job No: ");
484     int jobNo = sc.nextInt();
485     sc.nextLine();
486
487     System.out.print("Enter DeLabor Type: ");
488     String deLabType = sc.nextLine();
489
490     String sql = "Insert into DeLabor_Info values (" + jobNo + ", "
491                 + deLabType + ")";
492
493     Statement statement = connection.createStatement();
494     statement.execute(sql);
495     System.out.println("Insert Complete..");
496   }
497   catch(Exception ex)
498   {
499     System.out.println("Error in InsertDeLaborInfo(). Error: " + ex.toString());
500   }
501 }
502
503 //Option 2 Delete DeLabor Info
504 public static void DeleteDeLaborInfo()
505 {
506   try
507   {
508     Connection connection = DriverManager.getConnection(url); //sql connection.
509     Scanner sc = new Scanner(System.in);
510     System.out.print("Enter Job No: ");
511     int jobNo = sc.nextInt();
512     sc.nextLine();
513
514     String sql = "Delete from DeLabor_Info where job_no = " + jobNo;
515
516     Statement statement = connection.createStatement();
517     statement.execute(sql);
518     System.out.println("Delete Complete..");
519   }
520   catch(Exception ex)
521   {
522     System.out.println("Error in DeleteDeLaborInfo(). Error: " + ex.toString());
523   }
524 }
525
526 //Option 1 Insert Customer Info
527 public static void InsertCustomerInfo()
528 {
529   try
530   {
531     Connection connection = DriverManager.getConnection(url); //sql connection.
532     Scanner sc = new Scanner(System.in);
533     System.out.print("Enter Cust File: ");
534     String custfile = sc.nextLine();
535
536     String sql = "Insert into Customer_Info values ('" + custfile + "')";
537
538     Statement statement = connection.createStatement();
539     statement.execute(sql);
540     System.out.println("File " + custfile + " created.");
541   }
542   catch(Exception ex)
543   {
544     System.out.println("Error in InsertCustomerInfo(). Error: " + ex.toString());
545   }
546 }
547
548 //Option 0 Main Method
549 public static void main(String[] args)
550 {
551   Program p = new Program();
552   p.main();
553 }
554 
```

Task List

Find All Activate...

Outline

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search

Writable Smart Insert 210 : 29 : 5719

241 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Program.java

```

218         System.out.println("Error in UpdatePaintJob(). Error: " + ex.toString());
219   }
220 }
221
222 //Option 14 Delete Cut jobs based on date range.
223 public static void DeleteCutJobs()
224 {
225   try
226   {
227     Connection connection = DriverManager.getConnection(url); //sql connection.
228     Scanner sc = new Scanner(System.in);
229     System.out.print("Enter low job no: ");
230     int lowJobNo = sc.nextInt();
231     sc.nextLine(); //clear buffer
232
233     System.out.print("Enter high job no: ");
234     int highJobNo = sc.nextInt();
235     sc.nextLine(); //clear buffer.
236
237     String sql = "Delete from Job_Cut where job_no BETWEEN "
238                 + lowJobNo + " AND " + highJobNo;
239
240     //debug.
241     //System.out.println(sql);
242
243     Statement statement = connection.createStatement();
244     statement.execute(sql);
245     System.out.println("Deletions Complete..");
246   }
247   catch(Exception ex)
248   {
249     System.out.println("Error in DeleteCutJobs(). Error: " + ex.toString());
250   }
251 }
252
253 //Option 13 Insert Cut Jobs
254 public static void InsertCutJobs()
255 {
256   try
257   {
258     Connection connection = DriverManager.getConnection(url); //sql connection.
259     Scanner sc = new Scanner(System.in);
260     System.out.print("Enter Job No: ");
261     int jobNo = sc.nextInt();
262     sc.nextLine();
263
264     System.out.print("Enter Cut Type: ");
265     String cutType = sc.nextLine();
266
267     String sql = "Insert into Job_Cut values (" + jobNo + ", "
268                 + cutType + ")";
269
270     Statement statement = connection.createStatement();
271     statement.execute(sql);
272     System.out.println("Insert Complete..");
273   }
274   catch(Exception ex)
275   {
276     System.out.println("Error in InsertCutJobs(). Error: " + ex.toString());
277   }
278 }
279
280 //Option 12 Retrieve Cut Jobs
281 public static void RetrieveCutJobs()
282 {
283   try
284   {
285     Connection connection = DriverManager.getConnection(url); //sql connection.
286     Statement statement = connection.createStatement();
287     ResultSet rs = statement.executeQuery("Select * from Job_Cut");
288
289     while(rs.next())
290     {
291       System.out.println("Job No: " + rs.getInt("job_no") + " Cut Type: " + rs.getString("cut_type"));
292     }
293   }
294   catch(Exception ex)
295   {
296     System.out.println("Error in RetrieveCutJobs(). Error: " + ex.toString());
297   }
298 }
299
300 //Option 11 Retrieve All Cut Jobs
301 public static void RetrieveAllCutJobs()
302 {
303   try
304   {
305     Connection connection = DriverManager.getConnection(url); //sql connection.
306     Statement statement = connection.createStatement();
307     ResultSet rs = statement.executeQuery("Select * from Job_Cut");
308
309     while(rs.next())
310     {
311       System.out.println("Job No: " + rs.getInt("job_no") + " Cut Type: " + rs.getString("cut_type"));
312     }
313   }
314   catch(Exception ex)
315   {
316     System.out.println("Error in RetrieveAllCutJobs(). Error: " + ex.toString());
317   }
318 }
319
320 //Option 10 Confirm Job Completion
321 public static void ConfirmJobCompletion()
322 {
323   try
324   {
325     Connection connection = DriverManager.getConnection(url); //sql connection.
326     Scanner sc = new Scanner(System.in);
327     System.out.print("Enter Job No: ");
328     int jobNo = sc.nextInt();
329     sc.nextLine();
330
331     String sql = "Update Job_Cut Set completed = 1 where job_no = " + jobNo;
332
333     Statement statement = connection.createStatement();
334     statement.execute(sql);
335     System.out.println("Job Completed..");
336   }
337   catch(Exception ex)
338   {
339     System.out.println("Error in ConfirmJobCompletion(). Error: " + ex.toString());
340   }
341 }
342
343 //Option 9 Insert Transaction Info
344 public static void InsertTransactionInfo()
345 {
346   try
347   {
348     Connection connection = DriverManager.getConnection(url); //sql connection.
349     Scanner sc = new Scanner(System.in);
350     System.out.print("Enter Job No: ");
351     int jobNo = sc.nextInt();
352     sc.nextLine();
353
354     System.out.print("Enter Transaction Type: ");
355     String transType = sc.nextLine();
356
357     String sql = "Insert into Transaction_Info values (" + jobNo + ", "
358                 + transType + ")";
359
360     Statement statement = connection.createStatement();
361     statement.execute(sql);
362     System.out.println("Insert Complete..");
363   }
364   catch(Exception ex)
365   {
366     System.out.println("Error in InsertTransactionInfo(). Error: " + ex.toString());
367   }
368 }
369
370 //Option 8 Delete Transaction Info
371 public static void DeleteTransactionInfo()
372 {
373   try
374   {
375     Connection connection = DriverManager.getConnection(url); //sql connection.
376     Scanner sc = new Scanner(System.in);
377     System.out.print("Enter Job No: ");
378     int jobNo = sc.nextInt();
379     sc.nextLine();
380
381     String sql = "Delete from Transaction_Info where job_no = " + jobNo;
382
383     Statement statement = connection.createStatement();
384     statement.execute(sql);
385     System.out.println("Delete Complete..");
386   }
387   catch(Exception ex)
388   {
389     System.out.println("Error in DeleteTransactionInfo(). Error: " + ex.toString());
390   }
391 }
392
393 //Option 7 Retrieve Transaction Info
394 public static void RetrieveTransactionInfo()
395 {
396   try
397   {
398     Connection connection = DriverManager.getConnection(url); //sql connection.
399     Statement statement = connection.createStatement();
400     ResultSet rs = statement.executeQuery("Select * from Transaction_Info");
401
402     while(rs.next())
403     {
404       System.out.println("Job No: " + rs.getInt("job_no") + " Transaction Type: " + rs.getString("trans_type"));
405     }
406   }
407   catch(Exception ex)
408   {
409     System.out.println("Error in RetrieveTransactionInfo(). Error: " + ex.toString());
410   }
411 }
412
413 //Option 6 Insert Process Info
414 public static void InsertProcessInfo()
415 {
416   try
417   {
418     Connection connection = DriverManager.getConnection(url); //sql connection.
419     Scanner sc = new Scanner(System.in);
420     System.out.print("Enter Job No: ");
421     int jobNo = sc.nextInt();
422     sc.nextLine();
423
424     System.out.print("Enter Process Type: ");
425     String procType = sc.nextLine();
426
427     String sql = "Insert into Process_Info values (" + jobNo + ", "
428                 + procType + ")";
429
430     Statement statement = connection.createStatement();
431     statement.execute(sql);
432     System.out.println("Insert Complete..");
433   }
434   catch(Exception ex)
435   {
436     System.out.println("Error in InsertProcessInfo(). Error: " + ex.toString());
437   }
438 }
439
440 //Option 5 Delete Process Info
441 public static void DeleteProcessInfo()
442 {
443   try
444   {
445     Connection connection = DriverManager.getConnection(url); //sql connection.
446     Scanner sc = new Scanner(System.in);
447     System.out.print("Enter Job No: ");
448     int jobNo = sc.nextInt();
449     sc.nextLine();
450
451     String sql = "Delete from Process_Info where job_no = " + jobNo;
452
453     Statement statement = connection.createStatement();
454     statement.execute(sql);
455     System.out.println("Delete Complete..");
456   }
457   catch(Exception ex)
458   {
459     System.out.println("Error in DeleteProcessInfo(). Error: " + ex.toString());
460   }
461 }
462
463 //Option 4 Retrieve Process Info
464 public static void RetrieveProcessInfo()
465 {
466   try
467   {
468     Connection connection = DriverManager.getConnection(url); //sql connection.
469     Statement statement = connection.createStatement();
470     ResultSet rs = statement.executeQuery("Select * from Process_Info");
471
472     while(rs.next())
473     {
474       System.out.println("Job No: " + rs.getInt("job_no") + " Process Type: " + rs.getString("proc_type"));
475     }
476   }
477   catch(Exception ex)
478   {
479     System.out.println("Error in RetrieveProcessInfo(). Error: " + ex.toString());
480   }
481 }
482
483 //Option 3 Insert DeLabor Info
484 public static void InsertDeLaborInfo()
485 {
486   try
487   {
488     Connection connection = DriverManager.getConnection(url); //sql connection.
489     Scanner sc = new Scanner(System.in);
490     System.out.print("Enter Job No: ");
491     int jobNo = sc.nextInt();
492     sc.nextLine();
493
494     System.out.print("Enter DeLabor Type: ");
495     String deLabType = sc.nextLine();
496
497     String sql = "Insert into DeLabor_Info values (" + jobNo + ", "
498                 + deLabType + ")";
499
500     Statement statement = connection.createStatement();
501     statement.execute(sql);
502     System.out.println("Insert Complete..");
503   }
504   catch(Exception ex)
505   {
506     System.out.println("Error in InsertDeLaborInfo(). Error: " + ex.toString());
507   }
508 }
509
510 //Option 2 Delete DeLabor Info
511 public static void DeleteDeLaborInfo()
512 {
513   try
514   {
515     Connection connection = DriverManager.getConnection(url); //sql connection.
516     Scanner sc = new Scanner(System.in);
517     System.out.print("Enter Job No: ");
518     int jobNo = sc.nextInt();
519     sc.nextLine();
520
521     String sql = "Delete from DeLabor_Info where job_no = " + jobNo;
522
523     Statement statement = connection.createStatement();
524     statement.execute(sql);
525     System.out.println("Delete Complete..");
526   }
527   catch(Exception ex)
528   {
529     System.out.println("Error in DeleteDeLaborInfo(). Error: " + ex.toString());
530   }
531 }
532
533 //Option 1 Insert Customer Info
534 public static void InsertCustomerInfo()
535 {
536   try
537   {
538     Connection connection = DriverManager.getConnection(url); //sql connection.
539     Scanner sc = new Scanner(System.in);
540     System.out.print("Enter Cust File: ");
541     String custfile = sc.nextLine();
542
543     String sql = "Insert into Customer_Info values ('" + custfile + "')";
544
545     Statement statement = connection.createStatement();
546     statement.execute(sql);
547     System.out.println("File " + custfile + " created.");
548   }
549   catch(Exception ex)
550   {
551     System.out.println("Error in InsertCustomerInfo(). Error: " + ex.toString());
552   }
553 }
554
555 //Option 0 Main Method
556 public static void main(String[] args)
557 {
558   Program p = new Program();
559   p.main();
560 }
561 
```

Task List

Find All Activate...

Outline

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search

Writable Smart Insert 239 : 55 : 6564

241 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

239     System.out.println("Deletions Complete.");
240 }
241 catch(Exception ex)
242 {
243     System.out.println("Error in DeleteCutJobs(). Error: " + ex.toString());
244 }
245 }
246
//Option 13 Retrieve Customers based on category
247 public static void RetrieveCustomers()
248 {
249     try
250     {
251         Connection connection = DriverManager.getConnection(url); //sql connection.
252         Scanner sc = new Scanner(System.in);
253         System.out.print("Enter lower category: ");
254         int lowCat = sc.nextInt();
255         sc.nextLine();
256         System.out.print("Enter high category: ");
257         int highCat = sc.nextInt();
258         sc.nextLine();
259
260         String sql = "SELECT cust_name FROM Customers WHERE"
261             + " category between " + lowCat + " AND " + highCat
262             + " ORDER by cust_name";
263
264         //debug
265         //System.out.println(sql);
266
267         Statement statement = connection.createStatement();

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search    Writable Smart Insert 268 : 13 : 7324

241 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

268     Statement statement = connection.createStatement();
269     ResultSet resultSet = statement.executeQuery(sql);
270     while(resultSet.next())
271     {
272         System.out.println("Customer: " + resultSet.getString("cust_name"));
273     }
274     catch(Exception ex)
275     {
276         System.out.println("Error in RetrieveCustomers(). Error: " + ex.toString());
277     }
278 }
279
//Option 12 Get jobs based on dept and date completed
280 public static void RetrieveJobsCompleted()
281 {
282     try
283     {
284         Connection connection = DriverManager.getConnection(url); //sql connection.
285         Scanner sc = new Scanner(System.in);
286
287         System.out.print("Enter Date Completed: ");
288         String dateCompleted = sc.nextLine();
289         System.out.print("Enter Department Supervising: ");
290         int deptNum = sc.nextInt();
291         sc.nextLine(); //clear buffer.
292
293         String sql1 = "SELECT job_no FROM Supervises WHERE dept_num = " + deptNum;
294         Statement statement = connection.createStatement();

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search    Writable Smart Insert 297 : 40 : 8247

241 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

297     Statement statement = connection.createStatement();
298     ResultSet rs1 = statement.executeQuery(sql1);
299     List<Integer> jobNos = new ArrayList<Integer>();
300     while(rs1.next())
301     {
302         jobNos.add(rs1.getInt("job_no")); //populate the list.
303     }
304     //Create String from jobNos list. This will be the WHERE clause.
305     String whereClause = "WHERE job_no in (";
306     Iterator<Integer> iterator = jobNos.iterator();
307     while(iterator.hasNext())
308     {
309         Integer i = iterator.next();
310         whereClause += i.toString();
311         whereClause += ", ";
312     }
313     whereClause += " -1) AND date_completed = '" + dateCompleted + "'"; // -1 is
314     String sql2 = "SELECT * FROM Job_Cut " + whereClause;
315     String sql3 = "SELECT * FROM Job_Fit " + whereClause;
316     String sql4 = "SELECT * FROM Job_Paint " + whereClause;
317
318     ResultSet rs2 = statement.executeQuery(sql2);
319     int assId = 0; //for assembly Id;
320     while(rs2.next()) //for Job_Cut
321     {
322         int jobNo = rs2.getInt("job_no");
323         String sql = "SELECT assembly_id FROM Manufactures WHERE job_no = " + jobNo;
324         Statement st2 = connection.createStatement();
325         ResultSet rs = st2.executeQuery(sql);
326         while(rs.next())

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 326 : 33 : 9433

Type here to search

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

326             while(rs.next())
327             {
328                 assId = rs.getInt("assembly_id");
329             }
330             System.out.println("Job No: " + jobNo
331                     + "; Date Started: " + rs2.getString("date_started")
332                     + "; Date Completed: " + rs2.getString("date_completed")
333                     + "; Machine Type: " + rs2.getString("machine_type")
334                     + "; Time Used: " + rs2.getInt("time_used")
335                     + "; Material Used: " + rs2.getString("material_used")
336                     + "; Labor Time: " + rs2.getInt("labor_time"));
337
338
339
340             ResultSet rs3 = statement.executeQuery(sql3);
341             while(rs3.next()) //for Job_Fit
342             {
343                 int jobNo = rs3.getInt("job_no");
344                 String sql = "SELECT assembly_id FROM Manufactures WHERE job_no = " + jobNo;
345                 Statement st2 = connection.createStatement();
346                 ResultSet rs = st2.executeQuery(sql);
347                 while(rs.next())
348                 {
349                     assId = rs.getInt("assembly_id");
350                 }
351                 System.out.println("Job No: " + jobNo
352                     + "; Date Started: " + rs3.getString("date_started")
353                     + "; Date Completed: " + rs3.getString("date_completed")
354                     + "; Labor Time: " + rs3.getInt("labor_time"));
355             }

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 326 : 13 : 9416

Type here to search

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Program.java

```

357     ResultSet rs4 = statement.executeQuery(sql4);
358     while(rs4.next()) //For Job_Paint
359     {
360         int jobNo = rs4.getInt("job_no");
361         String sql = "SELECT assembly_id FROM Manufactures WHERE job_no = " + jobNo;
362         Statement st2 = connection.createStatement();
363         ResultSet rs = st2.executeQuery(sql);
364         while(rs.next())
365         {
366             assId = rs.getInt("assembly_id");
367         }
368         System.out.println("Job No: " + jobNo
369                     + "; Date Started: " + rs4.getString("date_started")
370                     + "; Date Completed: " + rs4.getString("date_completed")
371                     + "; Color: " + rs4.getString("color")
372                     + "; Volume: " + rs4.getInt("volume")
373                     + "; Labor Time: " + rs4.getInt("labor_time"));
374     }
375     catch(Exception ex)
376     {
377         System.out.println("Error in RetrieveJobsCompleted(). Error: " + ex.toString());
378     }
379 }
380 //Option 11 get processes from given assembly id.
381 public static void RetrieveProcesses()
382 {
383     try
384     {

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 386 : 10 : 11467

242 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Program.java

```

387 Connection connection = DriverManager.getConnection(url); //sql connection.
388 Scanner sc = new Scanner(System.in);
389
390 System.out.print("Enter Assembly Id: ");
391 int assId = sc.nextInt();
392 sc.nextLine();
393
394 String sql1 = "SELECT process_id FROM Manufactures WHERE assembly_id = " + assId;
395 Statement statement = connection.createStatement();
396 ResultSet resultSet = statement.executeQuery(sql1); //execute select query.
397 List<Integer> processIds = new ArrayList<Integer>();
398 int laborTime = 0; //initialize variable.
399 while(resultSet.next())
400 {
401     processIds.add(resultSet.getInt("process_id")); //populate the list.
402 }
403 //Create String for jobNos list. This will be the WHERE clause.
404 String whereClause = "WHERE process_id IN (";
405 Iterator<Integer> iterator = processIds.iterator();
406 while(iterator.hasNext())
407 {
408     Integer i = iterator.next();
409     whereClause += i.toString();
410     whereClause += ",";
411 }
412 whereClause += "-1) AND process_data like '%passed%'"; // -1 is breakup of
413
414 String sql2 = "SELECT * FROM Processes " + whereClause;
415
416 //debug.

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 416 : 9 : 12611

242 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

419     ResultSet rs2 = statement.executeQuery(sql2);
420     System.out.println("Processes passed by Assembly: " + assId);
421     while(rs2.next())
422     {
423         System.out.print("Process Id: " + rs2.getInt("process_id")
424                     + " Process Data: " + rs2.getString("process_data"));
425     }
426 }
427 catch(Exception ex)
428 {
429     System.out.println("Error in RetrieveProcesses(). Error: " + ex.toString());
430 }
431
432 //Option 10 Retrieve labor time based on department and it's by jobs completed.
433 public static void RetrieveDepLaborTime()
434 {
435     try
436     {
437         Connection connection = DriverManager.getConnection(url); //sql connection.
438         Scanner sc = new Scanner(System.in);
439
440         System.out.print("Enter Department Number: ");
441         int deptNum = sc.nextInt();
442         sc.nextLine(); //clear buffer.
443
444         System.out.print("Enter Completion Date: ");
445         String dateComplete = sc.nextLine();
446
447         String sql1 = "SELECT job_no FROM Supervises WHERE dept_num = " + deptNum;
448
449         Statement statement = connection.createStatement();
450         ResultSet resultSet = statement.executeQuery(sql1); //execute select query.
451         List<Integer> jobNos = new ArrayList<Integer>();
452         int laborTime = 0; //initialize variable.
453         while(resultSet.next())
454         {
455             jobNos.add(resultSet.getInt("job_no")); //populate the list.
456         }
457         //Create String from jobNos list. This will be the WHERE clause.
458         String whereClause = "WHERE job_no IN (";
459         Iterator<Integer> iterator = jobNos.iterator();
460         while(iterator.hasNext())
461         {
462             Integer i = iterator.next();
463             whereClause += i.toString();
464             whereClause += ", ";
465         }
466         whereClause += "-1 AND date_completed = '" + dateComplete + "'"; // -1 is !
467
468         String sql2 = "SELECT labor_time FROM Job_Cut " + whereClause;
469         String sql3 = "SELECT labor_time FROM Job_Fit " + whereClause;
470         String sql4 = "SELECT labor_time FROM Job_Paint " + whereClause;
471
472         ResultSet rs2 = statement.executeQuery(sql2);
473         while(rs2.next())
474         {
475             laborTime += rs2.getInt("labor_time");
476         }
477
478         ResultSet rs3 = statement.executeQuery(sql3);
    
```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 448 : 14 : 13552

242 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

449         ResultSet rs4 = statement.executeQuery(sql4);
450         while(rs4.next())
451         {
452             laborTime += rs4.getInt("labor_time");
453         }
454
455         System.out.println("Total Labor Time: " + laborTime);
456     }
457 }
458 catch(Exception ex)
459 {
460     System.out.println("Error in RetrieveDepLaborTime(). Error: " + ex.toString());
461 }
462
463 //Option 11 Delete Cut Jobs
464 public static void DeleteCutJobs()
465 {
466     try
467     {
468         Connection connection = DriverManager.getConnection(url); //sql connection.
469         Scanner sc = new Scanner(System.in);
470
471         System.out.print("Enter Department Number: ");
472         int deptNum = sc.nextInt();
473         sc.nextLine(); //clear buffer.
474
475         System.out.print("Enter Completion Date: ");
476         String dateComplete = sc.nextLine();
477
478         String sql1 = "SELECT job_no FROM Supervises WHERE dept_num = " + deptNum;
479
480         Statement statement = connection.createStatement();
481         ResultSet resultSet = statement.executeQuery(sql1); //execute select query.
482         List<Integer> jobNos = new ArrayList<Integer>();
483         int laborTime = 0; //initialize variable.
484         while(resultSet.next())
485         {
486             jobNos.add(resultSet.getInt("job_no")); //populate the list.
487         }
488
489         //Create String from jobNos list. This will be the WHERE clause.
490         String whereClause = "WHERE job_no IN (";
491         Iterator<Integer> iterator = jobNos.iterator();
492         while(iterator.hasNext())
493         {
494             Integer i = iterator.next();
495             whereClause += i.toString();
496             whereClause += ", ";
497         }
498         whereClause += "-1 AND date_completed = '" + dateComplete + "'"; // -1 is !
499
500         String sql2 = "DELETE FROM Job_Cut WHERE job_no IN (" + whereClause + ")";
501         String sql3 = "DELETE FROM Job_Fit WHERE job_no IN (" + whereClause + ")";
502         String sql4 = "DELETE FROM Job_Paint WHERE job_no IN (" + whereClause + ")";
503
504         Statement statement2 = connection.createStatement();
505         statement2.executeUpdate(sql2);
506         statement2.executeUpdate(sql3);
507         statement2.executeUpdate(sql4);
508
509         System.out.println("Jobs Deleted Successfully!");
510     }
511 }
512
513 //Option 12 Import/Export Customer Information
514 public static void ImportExportCust(int choice)
515 {
516     try
517     {
518         Connection connection = DriverManager.getConnection(url); //sql connection.
519         Scanner sc = new Scanner(System.in);
520
521         if(choice == 1)
522         {
523             System.out.print("Enter Customer ID: ");
524             int custId = sc.nextInt();
525             sc.nextLine(); //clear buffer.
526
527             System.out.print("Enter Customer Name: ");
528             String custName = sc.nextLine();
529
530             System.out.print("Enter Customer Address: ");
531             String custAddress = sc.nextLine();
532
533             System.out.print("Enter Customer City: ");
534             String custCity = sc.nextLine();
535
536             System.out.print("Enter Customer State: ");
537             String custState = sc.nextLine();
538
539             System.out.print("Enter Customer Zip: ");
540             String custZip = sc.nextLine();
541
542             String sql1 = "UPDATE Customer SET name = '" + custName +
543                         "', address = '" + custAddress +
544                         "', city = '" + custCity +
545                         "', state = '" + custState +
546                         "', zip = '" + custZip +
547                         "' WHERE id = " + custId;
548
549             Statement statement = connection.createStatement();
550             statement.executeUpdate(sql1);
551
552             System.out.println("Customer Updated Successfully!");
553         }
554         else if(choice == 2)
555         {
556             System.out.print("Enter Customer ID: ");
557             int custId = sc.nextInt();
558             sc.nextLine(); //clear buffer.
559
560             String sql1 = "DELETE FROM Customer WHERE id = " + custId;
561
562             Statement statement = connection.createStatement();
563             statement.executeUpdate(sql1);
564
565             System.out.println("Customer Deleted Successfully!");
566         }
567     }
568 }
569
570 //Option 13 Exit Program
571 public static void main(String[] args)
572 {
573     System.out.println("Welcome to the Manufacturing System!");
574     System.out.println("Please Select an Option:");
575     System.out.println("1. Import/Export Customer Information");
576     System.out.println("2. Update Paint Job");
577     System.out.println("3. Delete Cut Jobs");
578     System.out.println("4. Retrieve Customers");
579     System.out.println("5. Retrieve Processes");
580     System.out.println("6. Retrieve Department Labor Time");
581     System.out.println("7. Retrieve Assembly Cost");
582     System.out.println("8. Insert Transaction Info");
583     System.out.println("9. Confirm Job Completion");
584     System.out.println("10. Insert Job Info");
585
586     int choice = sc.nextInt();
587
588     switch(choice)
589     {
590         case 1:
591             ImportExportCust(1);
592             break;
593         case 2:
594             UpdatePaintJob();
595             break;
596         case 3:
597             DeleteCutJobs();
598             break;
599         case 4:
600             RetrieveCustomers();
601             break;
602         case 5:
603             RetrieveProcesses();
604             break;
605         case 6:
606             RetrieveDepLaborTime();
607             break;
608         case 7:
609             RetrieveAssemblyCost();
610             break;
611         case 8:
612             InsertTransactionInfo();
613             break;
614         case 9:
615             ConfirmJobCompletion();
616             break;
617         case 10:
618             InsertJobInfo();
619             break;
620         default:
621             System.out.println("Invalid Choice!");
622             break;
623     }
624 }
625
626 //Option 14 Exit Program
627 public static void ExitProgram()
628 {
629     System.out.println("Thank you for using the Manufacturing System!");
630     System.out.println("Goodbye!");
631 }
632
633 //Option 15 Exit Program
634 public static void ExitProgram()
635 {
636     System.out.println("Thank you for using the Manufacturing System!");
637     System.out.println("Goodbye!");
638 }
639
640 //Option 16 Exit Program
641 public static void ExitProgram()
642 {
643     System.out.println("Thank you for using the Manufacturing System!");
644     System.out.println("Goodbye!");
645 }
646
647 //Option 17 Exit Program
648 public static void ExitProgram()
649 {
650     System.out.println("Thank you for using the Manufacturing System!");
651     System.out.println("Goodbye!");
652 }
653
654 //Option 18 Exit Program
655 public static void ExitProgram()
656 {
657     System.out.println("Thank you for using the Manufacturing System!");
658     System.out.println("Goodbye!");
659 }
660
661 //Option 19 Exit Program
662 public static void ExitProgram()
663 {
664     System.out.println("Thank you for using the Manufacturing System!");
665     System.out.println("Goodbye!");
666 }
667
668 //Option 20 Exit Program
669 public static void ExitProgram()
670 {
671     System.out.println("Thank you for using the Manufacturing System!");
672     System.out.println("Goodbye!");
673 }
674
675 //Option 21 Exit Program
676 public static void ExitProgram()
677 {
678     System.out.println("Thank you for using the Manufacturing System!");
679     System.out.println("Goodbye!");
680 }
681
682 //Option 22 Exit Program
683 public static void ExitProgram()
684 {
685     System.out.println("Thank you for using the Manufacturing System!");
686     System.out.println("Goodbye!");
687 }
688
689 //Option 23 Exit Program
690 public static void ExitProgram()
691 {
692     System.out.println("Thank you for using the Manufacturing System!");
693     System.out.println("Goodbye!");
694 }
695
696 //Option 24 Exit Program
697 public static void ExitProgram()
698 {
699     System.out.println("Thank you for using the Manufacturing System!");
700     System.out.println("Goodbye!");
701 }
702
703 //Option 25 Exit Program
704 public static void ExitProgram()
705 {
706     System.out.println("Thank you for using the Manufacturing System!");
707     System.out.println("Goodbye!");
708 }
709
710 //Option 26 Exit Program
711 public static void ExitProgram()
712 {
713     System.out.println("Thank you for using the Manufacturing System!");
714     System.out.println("Goodbye!");
715 }
716
717 //Option 27 Exit Program
718 public static void ExitProgram()
719 {
720     System.out.println("Thank you for using the Manufacturing System!");
721     System.out.println("Goodbye!");
722 }
723
724 //Option 28 Exit Program
725 public static void ExitProgram()
726 {
727     System.out.println("Thank you for using the Manufacturing System!");
728     System.out.println("Goodbye!");
729 }
730
731 //Option 29 Exit Program
732 public static void ExitProgram()
733 {
734     System.out.println("Thank you for using the Manufacturing System!");
735     System.out.println("Goodbye!");
736 }
737
738 //Option 30 Exit Program
739 public static void ExitProgram()
740 {
741     System.out.println("Thank you for using the Manufacturing System!");
742     System.out.println("Goodbye!");
743 }
744
745 //Option 31 Exit Program
746 public static void ExitProgram()
747 {
748     System.out.println("Thank you for using the Manufacturing System!");
749     System.out.println("Goodbye!");
750 }
751
752 //Option 32 Exit Program
753 public static void ExitProgram()
754 {
755     System.out.println("Thank you for using the Manufacturing System!");
756     System.out.println("Goodbye!");
757 }
758
759 //Option 33 Exit Program
760 public static void ExitProgram()
761 {
762     System.out.println("Thank you for using the Manufacturing System!");
763     System.out.println("Goodbye!");
764 }
765
766 //Option 34 Exit Program
767 public static void ExitProgram()
768 {
769     System.out.println("Thank you for using the Manufacturing System!");
770     System.out.println("Goodbye!");
771 }
772
773 //Option 35 Exit Program
774 public static void ExitProgram()
775 {
776     System.out.println("Thank you for using the Manufacturing System!");
777     System.out.println("Goodbye!");
778 }
779
780 //Option 36 Exit Program
781 public static void ExitProgram()
782 {
783     System.out.println("Thank you for using the Manufacturing System!");
784     System.out.println("Goodbye!");
785 }
786
787 //Option 37 Exit Program
788 public static void ExitProgram()
789 {
790     System.out.println("Thank you for using the Manufacturing System!");
791     System.out.println("Goodbye!");
792 }
793
794 //Option 38 Exit Program
795 public static void ExitProgram()
796 {
797     System.out.println("Thank you for using the Manufacturing System!");
798     System.out.println("Goodbye!");
799 }
800
801 //Option 39 Exit Program
802 public static void ExitProgram()
803 {
804     System.out.println("Thank you for using the Manufacturing System!");
805     System.out.println("Goodbye!");
806 }
807
808 //Option 40 Exit Program
809 public static void ExitProgram()
810 {
811     System.out.println("Thank you for using the Manufacturing System!");
812     System.out.println("Goodbye!");
813 }
814
815 //Option 41 Exit Program
816 public static void ExitProgram()
817 {
818     System.out.println("Thank you for using the Manufacturing System!");
819     System.out.println("Goodbye!");
820 }
821
822 //Option 42 Exit Program
823 public static void ExitProgram()
824 {
825     System.out.println("Thank you for using the Manufacturing System!");
826     System.out.println("Goodbye!");
827 }
828
829 //Option 43 Exit Program
830 public static void ExitProgram()
831 {
832     System.out.println("Thank you for using the Manufacturing System!");
833     System.out.println("Goodbye!");
834 }
835
836 //Option 44 Exit Program
837 public static void ExitProgram()
838 {
839     System.out.println("Thank you for using the Manufacturing System!");
840     System.out.println("Goodbye!");
841 }
842
843 //Option 45 Exit Program
844 public static void ExitProgram()
845 {
846     System.out.println("Thank you for using the Manufacturing System!");
847     System.out.println("Goodbye!");
848 }
849
850 //Option 46 Exit Program
851 public static void ExitProgram()
852 {
853     System.out.println("Thank you for using the Manufacturing System!");
854     System.out.println("Goodbye!");
855 }
856
857 //Option 47 Exit Program
858 public static void ExitProgram()
859 {
860     System.out.println("Thank you for using the Manufacturing System!");
861     System.out.println("Goodbye!");
862 }
863
864 //Option 48 Exit Program
865 public static void ExitProgram()
866 {
867     System.out.println("Thank you for using the Manufacturing System!");
868     System.out.println("Goodbye!");
869 }
870
871 //Option 49 Exit Program
872 public static void ExitProgram()
873 {
874     System.out.println("Thank you for using the Manufacturing System!");
875     System.out.println("Goodbye!");
876 }
877
878 //Option 50 Exit Program
879 public static void ExitProgram()
880 {
881     System.out.println("Thank you for using the Manufacturing System!");
882     System.out.println("Goodbye!");
883 }
884
885 //Option 51 Exit Program
886 public static void ExitProgram()
887 {
888     System.out.println("Thank you for using the Manufacturing System!");
889     System.out.println("Goodbye!");
890 }
891
892 //Option 52 Exit Program
893 public static void ExitProgram()
894 {
895     System.out.println("Thank you for using the Manufacturing System!");
896     System.out.println("Goodbye!");
897 }
898
899 //Option 53 Exit Program
900 public static void ExitProgram()
901 {
902     System.out.println("Thank you for using the Manufacturing System!");
903     System.out.println("Goodbye!");
904 }
905
906 //Option 54 Exit Program
907 public static void ExitProgram()
908 {
909     System.out.println("Thank you for using the Manufacturing System!");
910     System.out.println("Goodbye!");
911 }
912
913 //Option 55 Exit Program
914 public static void ExitProgram()
915 {
916     System.out.println("Thank you for using the Manufacturing System!");
917     System.out.println("Goodbye!");
918 }
919
920 //Option 56 Exit Program
921 public static void ExitProgram()
922 {
923     System.out.println("Thank you for using the Manufacturing System!");
924     System.out.println("Goodbye!");
925 }
926
927 //Option 57 Exit Program
928 public static void ExitProgram()
929 {
930     System.out.println("Thank you for using the Manufacturing System!");
931     System.out.println("Goodbye!");
932 }
933
934 //Option 58 Exit Program
935 public static void ExitProgram()
936 {
937     System.out.println("Thank you for using the Manufacturing System!");
938     System.out.println("Goodbye!");
939 }
940
941 //Option 59 Exit Program
942 public static void ExitProgram()
943 {
944     System.out.println("Thank you for using the Manufacturing System!");
945     System.out.println("Goodbye!");
946 }
947
948 //Option 60 Exit Program
949 public static void ExitProgram()
950 {
951     System.out.println("Thank you for using the Manufacturing System!");
952     System.out.println("Goodbye!");
953 }
954
955 //Option 61 Exit Program
956 public static void ExitProgram()
957 {
958     System.out.println("Thank you for using the Manufacturing System!");
959     System.out.println("Goodbye!");
960 }
961
962 //Option 62 Exit Program
963 public static void ExitProgram()
964 {
965     System.out.println("Thank you for using the Manufacturing System!");
966     System.out.println("Goodbye!");
967 }
968
969 //Option 63 Exit Program
970 public static void ExitProgram()
971 {
972     System.out.println("Thank you for using the Manufacturing System!");
973     System.out.println("Goodbye!");
974 }
975
976 //Option 64 Exit Program
977 public static void ExitProgram()
978 {
979     System.out.println("Thank you for using the Manufacturing System!");
980     System.out.println("Goodbye!");
981 }
982
983 //Option 65 Exit Program
984 public static void ExitProgram()
985 {
986     System.out.println("Thank you for using the Manufacturing System!");
987     System.out.println("Goodbye!");
988 }
989
990 //Option 66 Exit Program
991 public static void ExitProgram()
992 {
993     System.out.println("Thank you for using the Manufacturing System!");
994     System.out.println("Goodbye!");
995 }
996
997 //Option 67 Exit Program
998 public static void ExitProgram()
999 {
1000    System.out.println("Thank you for using the Manufacturing System!");
1001   System.out.println("Goodbye!");
1002 }
```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 478 : 13 : 14728

243 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

479         while(rs3.next())
480     {
481         laborTime += rs3.getInt("labor_time");
482     }
483
484     ResultSet rs4 = statement.executeQuery(sql4);
485     while(rs4.next())
486     {
487         laborTime += rs4.getInt("labor_time");
488     }
489
490     //debug.
491     //System.out.println(sql2);
492     //System.out.println(sql3);
493     //System.out.println(sql4);
494
495     System.out.println("Department: " + deptNum + "; Total Labor Time: " + laborTime);
496 }
497 catch(Exception ex)
498 {
499     System.out.println("Error in RetrieveDepLaborTime(). Error: " + ex.toString());
500 }
501
502 }
503
504 //Option 9 Retrieve Assembly costs by id.
505 public static void RetrieveAssemblyCost()
506 {
507     try
508     {

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 508 : 10 : 15426

Type here to search 243 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

509 Connection connection = DriverManager.getConnection(url); //sql connection.
510 Scanner sc = new Scanner(System.in);
511
512 System.out.print("Enter Assembly Id: ");
513 int assId = sc.nextInt();
514
515 String sql = "SELECT acc_num FROM Has1 WHERE assembly_id = " + assId;
516 Statement statement = connection.createStatement();
517 ResultSet resultSet = statement.executeQuery(sql); //execute select query.
518 int accNum = 0;
519 while(resultSet.next())
520 {
521     accNum = resultSet.getInt("acc_num");
522 }
523
524 sql = "SELECT cost FROM Account_Assembly WHERE acc_num = " + accNum;
525 resultSet = statement.executeQuery(sql); //execute select query.
526 float cost = 0;
527 while(resultSet.next())
528 {
529     cost = resultSet.getFloat("cost");
530 }
531
532 System.out.println("Assembly Id: " + assId + " ; Costs: " + cost);
533 }
534 catch(Exception ex)
535 {
536     System.out.println("Error in RetrieveAssemblyCost(). Error: " + ex.toString());
537 }
538 }
```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 538 : 6 : 16392

Type here to search 243 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Project\_Shell

Program.java

```

540     //Option 8 Insert Transaction Information
541     public static void InsertTransactionInfo()
542     {
543         try
544         {
545             Connection connection = DriverManager.getConnection(url); //sql connection.
546             Scanner sc = new Scanner(System.in);
547             System.out.print("Enter Transaction No: ");
548             int transNo = sc.nextInt();
549             sc.nextLine(); //clear buffer.
550             System.out.print("Enter sup-cost: ");
551             double cost = sc.nextDouble();
552             sc.nextLine(); //clear buffer.
553
554             System.out.print("Enter job no for transaction: ");
555             int jobno = sc.nextInt();
556             sc.nextLine();
557
558             System.out.print("Enter account id: ");
559             int accId = sc.nextInt();
560             sc.nextLine(); //clear buffer.
561
562             String sql1 = "INSERT INTO [Transaction] VALUES (" + transNo + ", " + cost
563             String sql2 = "INSERT INTO Records VALUES (" + jobno + ", " + transNo + ")"
564             String sql3 = "INSERT INTO Updates VALUES (" + transNo + ", " + accId + ")"
565             //Added all 3 though only one will update since the primary keys are unique
566             String sql4 = "UPDATE Account_Assembly SET cost = " + cost + " WHERE acc_nur
567             String sql5 = "UPDATE Account_Process SET cost = " + cost + " WHERE acc_num
568             String sql6 = "UPDATE Account_Department SET cost = " + cost + " WHERE acc_"
569
570         }
571         Statement statement = connection.createStatement();
572         statement.execute(sql1); //execute insert query.
573         statement.execute(sql2); //execute insert query.
574         statement.execute(sql3); //execute insert query.
575         statement.execute(sql4); //execute insert query.
576         statement.execute(sql5); //execute insert query.
577         statement.execute(sql6); //execute insert query.
578         System.out.println("Insertions/Updates Successful");
579     }
580     catch(Exception ex)
581     {
582         System.out.println("Error in InsertTransactionInfo(). Error: " + ex.toString());
583     }
584
585     //Option 7 Update job with completion date and type info.
586     public static void ConfirmJobCompletion()
587     {
588         try
589         {
590             Connection connection = DriverManager.getConnection(url); //sql connection.
591             String sql1 = "";
592             String sql2 = "";
593             String sql3 = "";
594             Scanner sc = new Scanner(System.in);
595             System.out.print("Enter JobNo: ");
596             int jobNo = sc.nextInt();
597
598             //select statement to see if job exists.
599             sql1 = "select job_no from Job WHERE job_no = " + jobNo;

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Windows Taskbar: Type here to search, Smart Insert, Writable, 569 : 13 : 17698, 243 PM, 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Project\_Shell

Program.java

```

570             Statement statement = connection.createStatement();
571             statement.execute(sql1); //execute insert query.
572             statement.execute(sql2); //execute insert query.
573             statement.execute(sql3); //execute insert query.
574             statement.execute(sql4); //execute insert query.
575             statement.execute(sql5); //execute insert query.
576             statement.execute(sql6); //execute insert query.
577             System.out.println("Insertions/Updates Successful");
578         }
579     }
580     catch(Exception ex)
581     {
582         System.out.println("Error in InsertTransactionInfo(). Error: " + ex.toString());
583     }
584
585     //Option 7 Update job with completion date and type info.
586     public static void ConfirmJobCompletion()
587     {
588         try
589         {
590             Connection connection = DriverManager.getConnection(url); //sql connection.
591             String sql1 = "";
592             String sql2 = "";
593             String sql3 = "";
594             Scanner sc = new Scanner(System.in);
595             System.out.print("Enter JobNo: ");
596             int jobNo = sc.nextInt();
597
598             //select statement to see if job exists.
599             sql1 = "select job_no from Job WHERE job_no = " + jobNo;

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Windows Taskbar: Type here to search, Smart Insert, Writable, 589 : 10 : 18379, 246 PM, 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

601     Statement statement = connection.createStatement();
602     Statement statement2 = connection.createStatement();
603     Statement statement3 = connection.createStatement();
604     ResultSet resultSet = statement.executeQuery(sql1); //execute select query.
605     ResultSet resultSet2 = null;
606     ResultSet resultSet3 = null;
607     //See if in job table if so then loop through other 3 to find job table.
608     int counter = 1; //1 for paint
609     int result = 0;
610     while(resultSet.next())
611     {
612         result = resultSet.getInt("job_no");
613     }
614     if(result == jobNo)
615     {
616         int r2 = 0;
617         System.out.println("Job Found...");
618         sql1 = "SELECT job_no from Job_Paint WHERE job_no = " + jobNo; //new query
619         resultSet2 = statement2.executeQuery(sql1);
620         while(resultSet2.next())
621         {
622             r2 = resultSet2.getInt("job_no");
623         }
624         if(r2 != jobNo)
625         {
626             int r3 = 0;
627             counter++; //2 for fit
628             sql1 = "SELECT job_no from Job_Fit WHERE job_no = " + jobNo;
629             resultSet3 = statement3.executeQuery(sql1);
630             while(resultSet3.next())
631         }
632         r3 = resultSet3.getInt("job_no");
633     }
634     if(r3 != jobNo)
635     {
636         counter++; //3 for cut
637     }
638 }
639 else
640 {
641     System.out.println("Job not found.");
642     return; //exit function.
643 }
644
645 sc.nextLine(); //buffer cleared.
646 System.out.print("Enter date completed: ");
647 String dateComp = sc.nextLine();
648
649 sql2 = "UPDATE Job set date_completed = '" + dateComp + "' where job_no = "
650 if(counter == 1)
651 {
652     System.out.print("Enter Paint Color: ");
653     String color = sc.nextLine();
654     System.out.print("Enter Volume: ");
655     int volume = sc.nextInt();
656     sc.nextLine(); //clear buffer.
657     System.out.print("Enter Labor Time: ");
658     int laborTime = sc.nextInt();
659     sc.nextLine(); //clear buffer.
660 }
```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 630 : 13 : 19780

Windows Taskbar: Type here to search, File Explorer, Task View, Start, Taskbar icons, Date/Time: 246 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

631     r3 = resultSet3.getInt("job_no");
632 }
633 if(r3 != jobNo)
634 {
635     counter++; //3 for cut
636 }
637 }
638 else
639 {
640     System.out.println("Job not found.");
641     return; //exit function.
642 }
643
644 sc.nextLine(); //buffer cleared.
645 System.out.print("Enter date completed: ");
646 String dateComp = sc.nextLine();
647
648 sql2 = "UPDATE Job set date_completed = '" + dateComp + "' where job_no = "
649 if(counter == 1)
650 {
651     System.out.print("Enter Paint Color: ");
652     String color = sc.nextLine();
653     System.out.print("Enter Volume: ");
654     int volume = sc.nextInt();
655     sc.nextLine(); //clear buffer.
656     System.out.print("Enter Labor Time: ");
657     int laborTime = sc.nextInt();
658     sc.nextLine(); //clear buffer.
659 }
```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 660 : 47 : 20593

Windows Taskbar: Type here to search, File Explorer, Task View, Start, Taskbar icons, Date/Time: 246 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

661         sql3 = "UPDATE Job_Paint set date_completed = '" + dateComp
662             + "', color = '" + color
663             + "', volume = " + volume
664             + ", labor_time = " + laborTime + " where job_no = " + jobNo;
665     }
666     else if(counter == 2)
667     {
668         System.out.print("Enter Labor Time: ");
669         int laborTime = sc.nextInt();
670         sc.nextLine(); //clear buffer.
671         sql3 = "UPDATE Job_Fit set date_completed = '" + dateComp
672             + "', labor_time = " + laborTime + " where job_no = " + jobNo;
673     }
674     else if(counter == 3)
675     {
676         System.out.print("Enter Machine Type: ");
677         String machType = sc.nextLine();
678         System.out.print("Enter Time Used: ");
679         int timeUsed = sc.nextInt();
680         sc.nextLine(); //clear buffer.
681         System.out.print("Enter Material Used: ");
682         String material = sc.nextLine();
683         System.out.print("Enter Labor Time: ");
684         int laborTime = sc.nextInt();
685         sc.nextLine(); //clear buffer.
686         sql3 = "UPDATE Job_Cut set date_completed = '" + dateComp
687             + "', machine_type = '" + machType
688             + "', time_used = " + timeUsed
689             + ", material_used = '" + material
690             + ", labor_time = " + laborTime + " where job_no = " + jobNo;
691     }

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search Writable Smart Insert 690 : 57 : 21748

246 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

691     }
692     //Debug.
693     //System.out.println(sql1);
694     //System.out.println(sql2);
695     //System.out.println(sql3);
696
697     statement.execute(sql2); //execute insert query.
698     statement.execute(sql3); //execute insert query.
699     System.out.println("Insertions Successful");
700 }
701 catch (Exception ex)
702 {
703     System.out.println("Error in ConfirmJobCompletion(). Error: " + ex.toString());
704 }
705
706 }
707
708 //Option 6 insert job information
709 public static void InsertJobInfo()
710 {
711     try
712     {
713         Connection connection = DriverManager.getConnection(url); //sql connection.
714         Scanner sc = new Scanner(System.in);
715
716         String sql1 = "";
717         String sql2 = "";
718         String sql3 = "";
719         String sql4 = "";
720         String tableName = "";

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search Writable Smart Insert 720 : 35 : 22517

246 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

721     System.out.print("Enter Job No: ");
722     int jobNo = sc.nextInt();
723     System.out.print("Enter Job Type: 1)Cut 2)Paint 3)Fit Type Number. ");
724     int type = sc.nextInt();
725     sc.nextLine(); //clear buffer.
726     System.out.print("Enter Date for Job Started: ");
727     String dateStarted = sc.nextLine();
728     System.out.print("Enter Process Id for Job: ");
729     int procId = sc.nextInt();
730     sc.nextLine(); //clear buffer.
731     System.out.print("Enter Assembly Id for Job: ");
732     int assId = sc.nextInt();
733     sc.nextLine();
734
735     switch(type)
736     {
737         case 1:
738             tableName = "Job_Cut";
739             break;
740         case 2:
741             tableName = "Job_Paint";
742             break;
743         case 3:
744             tableName = "Job_Fit";
745             break;
746         default:
747             System.out.println("Incorrect Job Type.");
748             return; //break out of function.
749     }
750

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 750 : 13 : 23337

Type here to search 247 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

751     sql1 = "INSERT INTO Job (job_no, date_started) VALUES (" + jobNo + ", " + dateStarted + ")";
752     sql2 = "INSERT INTO " + tableName + " (job_no, date_started) VALUES (" + jobNo + ", " + dateStarted + ")";
753     sql3 = "INSERT INTO Assigned VALUES (" + procId + ", " + jobNo + ")";
754     sql4 = "INSERT INTO Manufactures VALUES (" + assId + ", " + procId + ", " + jobNo + ")";
755
756     String sql5 = "UPDATE Supervises set job_no = " + jobNo + " WHERE process_id";
757     /*
758     //debug.
759     System.out.println(sql1);
760     System.out.println(sql2);
761     System.out.println(sql3);
762     System.out.println(sql4);
763     */
764     Statement statement = connection.createStatement();
765     statement.execute(sql1); //execute insert query.
766     statement.execute(sql2); //execute insert query.
767     statement.execute(sql3); //execute insert query.
768     statement.execute(sql4); //execute insert query.
769     statement.execute(sql5); //execute insert query.
770     System.out.println("Insertions Successful");
771
772 } catch(Exception ex)
773 {
774     System.out.println("Error in InsertJobInfo(). Error: " + ex.toString());
775 }
776
777 //Option 5 Create account and associate it to appropriate thing.
778 public static void CreateAccount()
779

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 780 : 20 : 24546

Type here to search 247 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

782     try
783     {
784         Connection connection = DriverManager.getConnection(url); //sql connection.
785         String sql1 = "";
786         String sql2 = "";
787         String sql3 = "";
788         long millis=System.currentTimeMillis();
789         java.sql.Date date=new java.sql.Date(millis);
790         Scanner sc = new Scanner(System.in);
791         System.out.print("Enter Account Number: ");
792         int accNum = sc.nextInt();
793         sc.nextLine();
794         System.out.println("Enter type of account: 1)Process 2)Assembly 3)Department");
795         int type = sc.nextInt();
796         sc.nextLine();
797         int procId = 0;
798         int assId = 0;
799         int deptNum = 0;
800         switch(type)
801         {
802             //Process Account
803             case 1:
804                 System.out.print("Enter process id: ");
805                 procId = sc.nextInt();
806                 sql2 = "INSERT INTO Account_Process VALUES (" + accNum + ", '" + date + ' '
807                     + sql3 = "INSERT INTO Has1 VALUES (" + procId + ", " + accNum + ")";
808                 break;
809             //Assembly Account
810             case 2:
811                 System.out.print("Enter Assembly id: ");

```

Problems Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search    Writable Smart Insert 811: 57 : 25551 247 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

812         assId = sc.nextInt();
813         sql2 = "INSERT INTO Account_Assembly VALUES (" + accNum + ", '" + date +
814             + sql3 = "INSERT INTO Has1 VALUES (" + assId + ", " + accNum + ")";
815         break;
816     //Department Account
817     case 3:
818         System.out.print("Enter Department number: ");
819         deptNum = sc.nextInt();
820         sql2 = "INSERT INTO Account_Department VALUES (" + accNum + ", '" + date +
821             + sql3 = "INSERT INTO Has3 VALUES (" + deptNum + ", " + accNum + ")";
822         break;
823     default:
824         System.out.println("Incorrect type for process.");
825     }
826
827     sql1 = "INSERT INTO Account VALUES (" + accNum + ", '" + date + ')';
828
829     //System.out.println(sql1); //debug
830     //System.out.println(sql2); //debug
831     //System.out.println(sql3); //debug
832
833     Statement statement = connection.createStatement();
834     statement.execute(sql1); //execute insert query.
835     statement.execute(sql2); //execute insert query.
836     statement.execute(sql3); //execute insert query.
837     System.out.println("Insertions Successful");
838 }
839 catch (Exception ex)
840 {
841     System.out.println("Error in CreateAccount(). Error: " + ex.toString());

```

Problems Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search    Writable Smart Insert 841: 29 : 26643 247 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

842     }
843   }
844
845   //Option 4 enter process information followed by department information for supervisor
846   public static void InsertProcessInfo()
847   {
848     try
849     {
850       Connection connection = DriverManager.getConnection(url); //sql connection.
851       String sql1 = "";
852       String sql2 = "";
853
854       String procType = "";
855       String method = "";
856       String machineType = "";
857       Scanner sc = new Scanner(System.in);
858       System.out.print("Enter Process Id: ");
859       int procId = sc.nextInt();
860       sc.nextLine(); //buffer out
861       System.out.println("Enter the type: 1)Paint 2)Fit 3)Cut. Type number");
862       int type = sc.nextInt();
863       sc.nextLine();
864       switch(type)
865     {
866       //Paint type
867       case 1:
868         System.out.print("Enter paint type: ");
869         procType = sc.nextLine();
870         System.out.print("Enter method: ");
871         method = sc.nextLine();
872
873       sql2 = "INSERT INTO Process_Paint "
874           + "VALUES (" + procId + ", '" + procType + "', '" +
875           + method + ")";
876       break;
877     //Fit type
878     case 2:
879       System.out.print("Enter fit type: ");
880       procType = sc.nextLine();
881       sql2 = "INSERT INTO Process_Fit "
882           + "VALUES (" + procId + ", '" + procType + "')";
883       break;
884     //Cut type
885     case 3:
886       System.out.print("Enter the cut type: ");
887       procType = sc.nextLine();
888       System.out.print("Enter the machine type: ");
889       machineType = sc.nextLine();
890       sql2 = "INSERT INTO Process_Cut "
891           + "VALUES (" + procId + ", '" + procType + "', '" +
892           + machineType + ")";
893       break;
894     default:
895       System.out.println("Incorrect type for process.");
896     }
897
898     //Two sql insertion queries for the main table and tables related.
899
900   }
901 }
```

Problems Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 871 : 40 : 27538

Type here to search 248 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

867     case 1:
868       System.out.print("Enter paint type: ");
869       procType = sc.nextLine();
870       System.out.print("Enter method: ");
871       method = sc.nextLine();
872       sql2 = "INSERT INTO Process_Paint "
873           + "VALUES (" + procId + ", '" + procType + "', '" +
874           + method + ")";
875       break;
876     //Fit type
877     case 2:
878       System.out.print("Enter fit type: ");
879       procType = sc.nextLine();
880       sql2 = "INSERT INTO Process_Fit "
881           + "VALUES (" + procId + ", '" + procType + "')";
882       break;
883     //Cut type
884     case 3:
885       System.out.print("Enter the cut type: ");
886       procType = sc.nextLine();
887       System.out.print("Enter the machine type: ");
888       machineType = sc.nextLine();
889       sql2 = "INSERT INTO Process_Cut "
890           + "VALUES (" + procId + ", '" + procType + "', '" +
891           + machineType + ")";
892       break;
893     default:
894       System.out.println("Incorrect type for process.");
895     }
896
897
898   //Two sql insertion queries for the main table and tables related.
899
900 }
```

Problems Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 896 : 25 : 28314

Type here to search 248 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

897     sql1 = "INSERT INTO Processes VALUES (" + procId + ", '')";
898
899     //Now input department info. Insert into department and supervises table.
900     System.out.print("Enter Department: ");
901     int deptNum = sc.nextInt();
902     sc.nextLine(); //buffer out.
903     System.out.print("Enter Department data: ");
904     String deptData = sc.nextLine();
905
906     String sql3 = "INSERT INTO Department VALUES (" + deptNum + ", " +
907                   + deptData + ")";
908
909     String sql4 = "INSERT INTO Supervises VALUES (" + deptNum + ", " +
910                   + procId + ", '')";
911
912     /*
913      System.out.println(sql1); //debug
914      System.out.println(sql2); //debug
915      System.out.println(sql3); //debug
916      System.out.println(sql4); //debug
917    */
918     Statement statement = connection.createStatement();
919     statement.execute(sql1); //execute insert query.
920     statement.execute(sql2); //execute insert query.
921     statement.execute(sql3); //execute insert query.
922     statement.execute(sql4); //execute insert query.
923     System.out.println("Insertions Successful");
924   }
925
926   catch(Exception ex)
927   {
928     System.out.println("Error in InsertProcessInfo(). Error: " + ex.toString());
929   }
930 }
931
932 //Option 3 Enter Assembly Info.
933 public static void InsertAssemblyInfo()
934 {
935   try
936   {
937     Connection connection = DriverManager.getConnection(url); //sql connection.
938
939     Scanner sc = new Scanner(System.in);
940     System.out.print("Enter Assembly Id: ");
941     int assId = sc.nextInt();
942     sc.nextLine();
943     System.out.print("Enter Date Ordered: ");
944     String dateOrdered = sc.nextLine();
945     System.out.print("Enter Assembly Details: ");
946     String details = sc.nextLine();
947     System.out.print("Enter Customer Name: ");
948     String custName = sc.nextLine();
949
950     /*
951      /debug
952      System.out.println("Inputting following information...");
953      System.out.println("Assembly Id: " + assId + "\nDate Ordered: " + dateOrdered);
954      System.out.println("Assembly Details: " + details + "\nCustomer Name: " + custName);
955    */
956
957     //Two sql insertion queries for the main table and tables related.

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 926 : 29 : 29432

Windows Taskbar: Type here to search, File Explorer, Edge, File Manager, Task View, Settings, Taskbar Icons, 248 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

928     System.out.println("Error in InsertProcessInfo(). Error: " + ex.toString());
929   }
930 }
931
932 //Option 3 Enter Assembly Info.
933 public static void InsertAssemblyInfo()
934 {
935   try
936   {
937     Connection connection = DriverManager.getConnection(url); //sql connection.
938
939     Scanner sc = new Scanner(System.in);
940     System.out.print("Enter Assembly Id: ");
941     int assId = sc.nextInt();
942     sc.nextLine();
943     System.out.print("Enter Date Ordered: ");
944     String dateOrdered = sc.nextLine();
945     System.out.print("Enter Assembly Details: ");
946     String details = sc.nextLine();
947     System.out.print("Enter Customer Name: ");
948     String custName = sc.nextLine();
949
950     /*
951      /debug
952      System.out.println("Inputting following information...");
953      System.out.println("Assembly Id: " + assId + "\nDate Ordered: " + dateOrdered);
954      System.out.println("Assembly Details: " + details + "\nCustomer Name: " + custName);
955    */
956
957     //Two sql insertion queries for the main table and tables related.

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Writable Smart Insert 957 : 64 : 30406

Windows Taskbar: Type here to search, File Explorer, Edge, File Manager, Task View, Settings, Taskbar Icons, 248 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Project\_Shell

Program.java

```

958     String sql1 = "INSERT INTO Assembly VALUES (" + assId + "," +
959             + dateOrdered + "", "" + details + ")";
960
961     String sql2 = "INSERT INTO Orders VALUES (" + assId + ", '" + custName + "'"
962     //System.out.println(sql1); //debug
963     //System.out.println(sql2); //debug
964
965     Statement statement = connection.createStatement();
966     statement.execute(sql1); //execute insert query.
967     statement.execute(sql2); //execute insert query.
968     System.out.println("Insertions Successful");
969 }
970 catch(Exception ex)
971 {
972     System.out.println("Error in the InsertAssemblyInfo. Error: " + ex.toString());
973 }
974 }

//Option 2 Enter Department Information.
public static void InsertDepartment(int flag)
{
    try
    {
        Connection connection = DriverManager.getConnection(url); //sql connection.
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Department: ");
        int deptNum = sc.nextInt();
        sc.nextLine(); //buffer out.
    }
}

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search

Writable Smart Insert 987 : 41 : 31394

248 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Project\_Shell

Program.java

```

988     if(deptNum != -1)
989     {
990         System.out.print("Enter Department data: ");
991         String deptData = sc.nextLine();
992
993         /*
994         //debug
995         System.out.println("Inputting following information...");
996         System.out.println("Department Number: " + deptNum + "\nDepartment Data
997         + deptData);
998         */
999
1000        String sql = "INSERT INTO Department VALUES (" + deptNum + ","
1001            + deptData + ")";
1002        //System.out.println(sql); //debug
1003
1004        Statement statement = connection.createStatement();
1005        statement.execute(sql); //execute insert query.
1006        System.out.println("Insertion Successful");
1007    }
1008 catch(Exception ex)
1009 {
1010     System.out.println("Error in InsertDepartment. Error: " + ex.toString());
1011     InsertDepartment(1);
1012 }
1013 }
1014 }

//Option 1 Enter Customer information
public static void InsertCustomer(int flag)

```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Type here to search

Writable Smart Insert 1017 : 48 : 32257

249 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

```

1019     try
1020     {
1021         Connection connection = DriverManager.getConnection(url); //sql connection.
1022
1023         if(flag > 0)
1024             System.out.println("Type exit to go back to options");
1025
1026         Scanner sc = new Scanner(System.in);
1027         System.out.print("\nPlease input Customer name: ");
1028         String custName = sc.nextLine();
1029
1030         if(!custName.toLowerCase().equals("exit"))
1031         {
1032             System.out.print("\nPlease input Customer address: ");
1033             String address = sc.nextLine();
1034             System.out.print("\nPlease input Customer category (1-10): ");
1035             int category = sc.nextInt();
1036             sc.nextLine(); //buffer out.
1037
1038             /*
1039             //debug
1040             System.out.println("Inputing following information...");
1041             System.out.println("Customer Name: " + custName + "\nAddress: " + address
1042                         + "\nCategory: " + category);
1043             */
1044
1045             String sql = "INSERT INTO Customers VALUES ('" + custName + "','" +
1046                         + address + "','" + category + ")";
1047             //System.out.println(sql); //debug
1048
1049         }
1050
1051         Statement statement = connection.createStatement();
1052         statement.executeUpdate(sql); //execute insert query.
1053         System.out.println("Insertion Successful");
1054     }
1055     catch(Exception ex)
1056     {
1057         System.out.println("Error in InsertCustomer. Error: " + ex.toString());
1058         InsertCustomer(1); //Going back to input.
1059     }
1060 }
1061
1062 //Displays the list of options.
1063 public static void DisplayOptions()
1064 {
1065     System.out.println("\n(1) Enter a new customer");
1066     System.out.println("(2) Enter a new department");
1067     System.out.println("(3) Enter new Assembly; REQUIRED: Customer_Name, "
1068                     + "Assembly_Details, Assembly_Id, Date_Ordered");
1069     System.out.println("(4) Enter new Process_Id and Department Information");
1070     System.out.println("(5) Create new Account and Setup");
1071     System.out.println("(6) Enter new Job; REQUIRED: Job_no, Assembly_id, "
1072                     + "Process_id, Date_of_Job_Start");
1073     System.out.println("(7) Enter Job Completion Date and Type of Job");
1074     System.out.println("(8) Enter Transaction Information");
1075 }
```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

Project - Project\_Shell/src/Program.java - Eclipse IDE

```

1045     String sql = "INSERT INTO Customers VALUES ('" + custName + "','" +
1046                         + address + "','" + category + ")";
1047     //System.out.println(sql); //debug
1048
1049     Statement statement = connection.createStatement();
1050     statement.executeUpdate(sql); //execute insert query.
1051     System.out.println("Insertion Successful");
1052 }
1053 catch(Exception ex)
1054 {
1055     System.out.println("Error in InsertCustomer. Error: " + ex.toString());
1056     InsertCustomer(1); //Going back to input.
1057 }
1058
1059 //Displays the list of options.
1060 public static void DisplayOptions()
1061 {
1062     System.out.println("\n(1) Enter a new customer");
1063     System.out.println("(2) Enter a new department");
1064     System.out.println("(3) Enter new Assembly; REQUIRED: Customer_Name, "
1065                     + "Assembly_Details, Assembly_Id, Date_Ordered");
1066     System.out.println("(4) Enter new Process_Id and Department Information");
1067     System.out.println("(5) Create new Account and Setup");
1068     System.out.println("(6) Enter new Job; REQUIRED: Job_no, Assembly_id, "
1069                     + "Process_id, Date_of_Job_Start");
1070     System.out.println("(7) Enter Job Completion Date and Type of Job");
1071     System.out.println("(8) Enter Transaction Information");
1072 }
```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:38:43 PM)

```

1064     {
1065         System.out.println("\n(1) Enter a new customer");
1066         System.out.println("(2) Enter a new department");
1067         System.out.println("(3) Enter new Assembly; REQUIRED: Customer_Name, "
1068             + "Assembly_Details, Assembly_Id, Date_Ordered");
1069         System.out.println("(4) Enter new Process_Id and Department Information");
1070         System.out.println("(5) Create new Account and Setup");
1071         System.out.println("(6) Enter new Job; REQUIRED: Job_no, Assembly_id, "
1072             + "Process_id, Date_of_Job_Start");
1073         System.out.println("(7) Enter Job Completion Date and Type of Job");
1074         System.out.println("(8) Enter Transaction Information");
1075         System.out.println("(9) Display Assembly Cost; REQUIRED: Assembly_Id");
1076         System.out.println("(10) Display total Labor time of"
1077             + "Job; REQUIRED: Department_Number and Date_Completed");
1078         System.out.println("(11) Display Processes and their Department;"
1079             + "REQUIRED: Process_Id");
1080         System.out.println("(12) Display Completed Jobs; REQUIRED: Date_Completed"
1081             + "and Department_Number");
1082         System.out.println("(13) Display Customers based on category");
1083         System.out.println("(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range");
1084         System.out.println("(15) Change color of a Paint-Job");
1085         System.out.println("(16) Import: Enter new customers from a data file "
1086             + "until the file is empty");
1087         System.out.println("(17) Export: Retrieve customers whose category is in a given"
1088             + "range");
1089         System.out.println("(18) Quit");
1090     }
1091 }
1092 }
1093

```

## 6) Java Execution

Note: Use shows only on insert by the app though all are. Some results have more than 5 or 10 requested. These are there from debugging.

```

2* * This program is a shell for the CS4513 Individual Project
Program Java Application|C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 9, 2019, 10:04:36 AM)
Welcome to JS Accounting System.
Below are options 1-18 that you can use.

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
1
Please input Customer name: Maranda
Please input Customer address: 8273 S. St.
Please input Customer category (1-10): 4

```

**Azure Data Studio**

The screenshot shows the Azure Data Studio interface. On the left is a sidebar with icons for File, Edit, View, Help, CONNECTIONS, SERVERS, and a search bar. The main area has two tabs: SQLQuery\_1 - disconnected and SQLQuery\_2 - mart34...rt3422. The SQLQuery\_2 tab contains the following SQL code:

```

1 Select * from Customers
2
3 select * from Department
4 Select * from Supervises
5 Select * from Job
6 select * from Processes
7 select * from Process_Cut
8 select * from Process_Paint
9 select * from Process_Fit
10 select * from Supervises
11 select * from Job

```

The Results tab displays a table with the following data:

	cust_name	address	category
1	John	101 Beaumont Dr.	1
2	Lisa	8274 S. St.	10
3	Louis	4545 Will Rd.	5
4	Maranda	8273 S. St.	4
5	Sandra	8989 Wind Dr.	3

A message box at the top right says "Restart Azure Data Studio to apply the latest update." with buttons for "Update Now", "Later", and "Release Notes".

**Eclipse IDE**

The screenshot shows the Eclipse IDE interface. The title bar says "Project - Project\_Shell/src/Program.java - Eclipse IDE". The workspace shows a Package Explorer with "Project\_Shell" and a Console tab with the following output:

```

Ln 2, Col 1 (25 selected) Spaces:4 UTF-8 CRLF SQL MSSQL 5 rows 00:00:00 mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db 10:08 AM 11/9/2019
Type here to search
Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 9, 2019, 10:04:36 AM)
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
2
Enter Department: 3
Enter Department data: Owned by Microsoft, cost needs to be under 10000
Insertion Successful

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit

```

The status bar at the bottom shows "10:10 AM 11/9/2019".

**Azure Data Studio**

File Edit View Help

CONNECTIONS

SERVICES

SQLQuery\_1 - disconnected SQLQuery\_2 - mart3422...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```

1 select * from Customers
2
3 select * from Department
4 Select * from Supervises
5 Select * from Job
6 select * from Processes
7 select * from Process_Cut
8 select * from Process_Paint
9 select * from Process_Fit
10 select * from Supervises
11 select * from Job

```

Results Messages

dept_num	dept_data
1	This department is supervised by Thomas in Newkirk
2	This department is a subsidiary of Amazon
3	Owned by Microsoft, cost needs to be under 10000
4	Run by Ryan, location Seattle
5	Run by Jared, stationed in lawton. Cost no higher than 55000
6	Run by Okada, subsidiary in Kyoto, Japan.

Restart Azure Data Studio to apply the latest update. Update Now Later Release Notes

AZURE

Ln 4, Col 1 (26 selected) Spaces:4 UTF-8 CRLF SQL MSSQL 6 rows 00:00:00 mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db 10:14 AM 11/9/2019

Type here to search

**Eclipse IDE**

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Program.java

```

2* * This program is a shell for the CS4513 Individual Project
8
9import java.io.BufferedReader;
18
19
20public class Program
21{
22    // Connect to database
23    final static String hostName = "mart3422-sql-server.database.windows.net";
24    final static String dbName = "cs-dsa-4513-sql-db";
25    final static String user = "user";

```

Task List

Outline

Problems Javadoc Declaration Console

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 12:42:55 PM)

(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Enter Assembly Id: 2  
Enter Date Ordered: 09/09/2019  
Enter Assembly Details: This assembly should have no more than 5 processes.  
Enter Customer Name: Louis

12:45 PM 11/10/2019

File Edit View Help

CONNECTIONS

Servers

```

1 Select * from Customers
2
3 select * from Department
4
5 Select * from Assembly
6 Select * from Orders
7
8 Select * from Supervises
9 Select * from Job
10 select * from Processes
11 select * from Process_Cut

```

SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

assembly_id	date_ordered	details
1	2019-11-01	This assembly needs to be done by 4th quarter.
2	2019-09-09	This assembly should have no more than 5 processes.
3	2019-11-01	Should have 3 processes no more than 10000 cost.
4	2019-10-11	Do assembly 1 before 4.
5	2018-05-23	To be finished by 2020 Q3.
6	2019-04-20	Cost should be under 25000
7	2019-10-11	No more than 2 processes
8	2019-11-09	HIGH COST! Needs to be done by Q2 2022
9	2015-05-18	High Importance! Please make sure processes are done correctly
10	2018-02-15	Needs to be done by Q4 2020.

Results Messages

assembly\_id date\_ordered details

assembly\_id cust\_name

File Edit View Help

CONNECTIONS

Servers

```

1 Select * from Customers
2
3 select * from Department
4
5 Select * from Assembly
6 Select * from Orders
7
8 Select * from Supervises
9 Select * from Job
10 select * from Processes
11 select * from Process_Cut

```

SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

assembly_id	cust_name
1	Sandra
2	Louis
3	John
4	Sandra
5	Maranda
6	Lisa
7	Treyson
8	Gage
9	Tommy
10	Louis

Results Messages

assembly\_id cust\_name

File Edit View Help

CONNECTIONS

Servers

```

1 Select * from Customers
2
3 select * from Department
4
5 Select * from Assembly
6 Select * from Orders
7
8 Select * from Supervises
9 Select * from Job
10 select * from Processes
11 select * from Process_Cut

```

SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

assembly_id	cust_name
1	Sandra
2	Louis
3	John
4	Sandra
5	Maranda
6	Lisa
7	Treyson
8	Gage
9	Tommy
10	Louis

Results Messages

assembly\_id cust\_name

File Edit View Help

CONNECTIONS

Servers

```

1 Select * from Customers
2
3 select * from Department
4
5 Select * from Assembly
6 Select * from Orders
7
8 Select * from Supervises
9 Select * from Job
10 select * from Processes
11 select * from Process_Cut

```

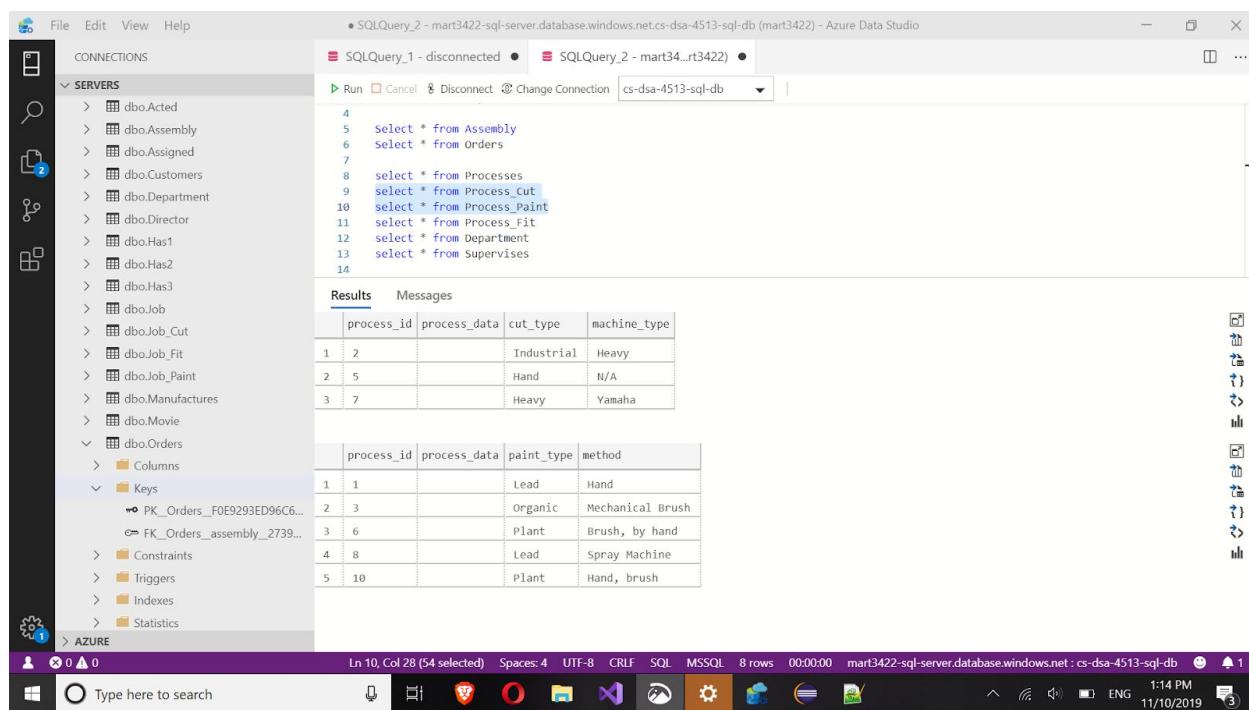
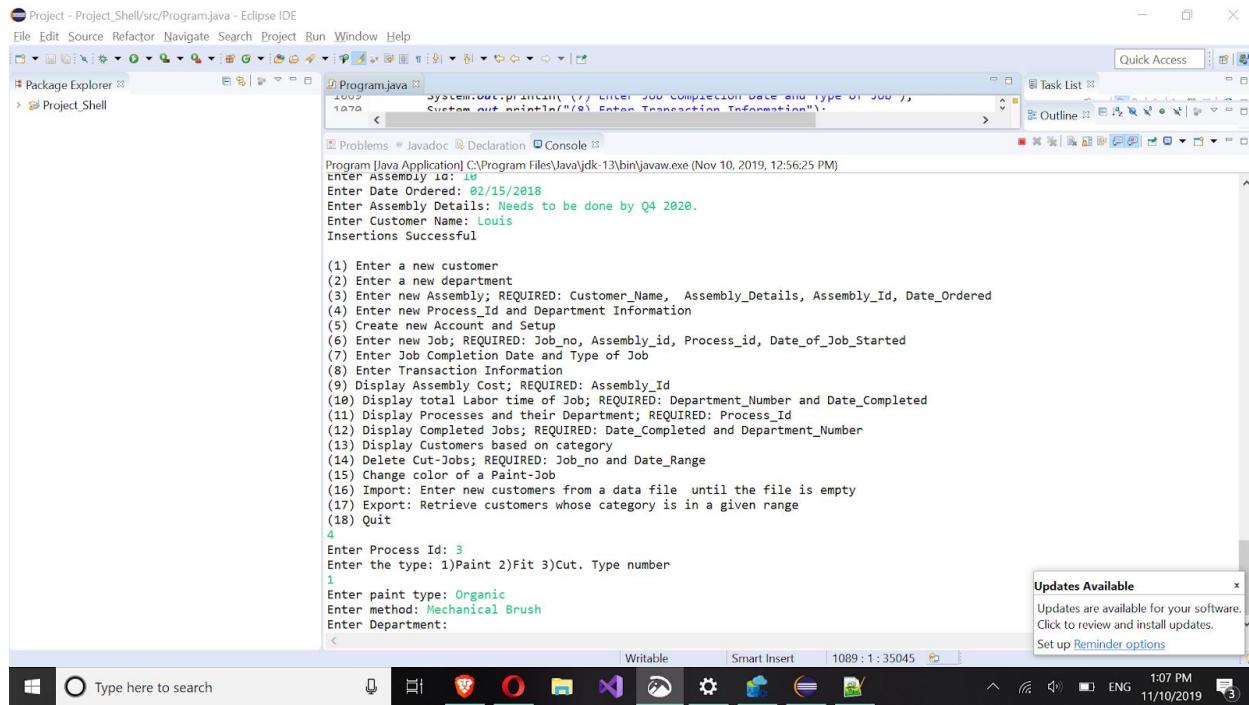
SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

assembly_id	cust_name
1	Sandra
2	Louis
3	John
4	Sandra
5	Maranda
6	Lisa
7	Treyson
8	Gage
9	Tommy
10	Louis

Results Messages

assembly\_id cust\_name



File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - ⇒ FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```
8 select * from Processes
9 select * from Process_Cut
10 select * from Process_Paint
11 select * from Process_fit
12 select * from Supervises
13 select * from Department
14
```

Results Messages

process_id	process_data	fit_type
1	4	light
2	9	Manuel

dept_num	process_id	job_no
1	2	2
2	7	3
3	8	4
4	9	5
5	10	6
6	11	7
7	12	8
8	13	9
9	14	10

Ln 13, Col 1 (53 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 11 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 11:15 PM 11/10/2019

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - ⇒ FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```
10 select * from Process_Paint
11 select * from Process_fit
12 select * from Supervises
13 select * from Department
14
15
```

Results Messages

dept_num	dept_data
1	This department is supervised by Thomas in Newkirk
2	This department is a subsidiary of Amazon
3	Owned by Microsoft, cost needs to be under 10000
4	Run by Ryan, location Seattle
5	Run by Jared, stationed in lawton. Cost no higher than 55000
6	Run by Okada, subsidiary in Kyoto, Japan.
7	Runs most paint processes
8	Stationed in N. Texas
9	N/A
10	Stationed in S. Korea
11	In Oklahoma City, Devon Tower 22nd floor
12	Located in Moore, OK
13	Done in Moore, OK. High cost.
14	Stationed in Yukon. Owned by Terisa.

Ln 14, Col 1 (26 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 14 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 11:15 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

1005 //System.out.println(" / / Enter Your Completion Date and type of Job ");
1070 System.out.println("(8) Enter Transaction Information");
    
```

Problems Javadoc Declaration Console

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 1:11:54 PM)

(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

5  
Enter Account Number: 2  
Enter type of account: 1)Process 2)Assembly 3)Department. Enter a number  
1  
Enter process id: 1  
Insertions Successful

(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Type here to search

File Edit View Help

SQLQuery\_2 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

SQLQuery\_1 - disconnected

SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```

14
15 select * from Account
16 select * from Account_Assembly
17 select * from Account_Department
18 select * from Account_Process
19 select * from Has1
20 select * from Has2
21 select * from Has3
    
```

Results Messages

acc_num	date_created
1	2019-11-05
2	2019-11-10
3	2019-11-10
4	2019-11-10
5	2019-11-10
6	2019-11-10
7	2019-11-10
8	2019-11-10
9	2019-11-10
10	2019-11-10

acc_num	date_created	cost
1	2019-11-05	209.56
2	2019-11-10	0
3	2019-11-10	0

Type here to search

File Edit View Help

11:17 PM 11/10/2019

File Edit View Help

SQLQuery\_1 - disconnected

SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```

14
15 select * from Account
16 select * from Account_Assembly
17 select * from Account_Department
18 select * from Account_Process
19 select * from Has1
20 select * from Has2
21 select * from Has3
    
```

Results Messages

acc_num	date_created
1	2019-11-05
2	2019-11-10
3	2019-11-10
4	2019-11-10
5	2019-11-10
6	2019-11-10
7	2019-11-10
8	2019-11-10
9	2019-11-10
10	2019-11-10

acc_num	date_created	cost
1	2019-11-05	209.56
2	2019-11-10	0
3	2019-11-10	0

Type here to search

File Edit View Help

12:22 PM 11/10/2019

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - ⇒ FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_1 - disconnected ● SQLQuery\_2 - mart34...rt3422) ●

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```

15 select * from Account
16 select * from Account_Assembly
17 select * from Account_Department
18 select * from Account_Process
19 select * from Has1
20 select * from Has2
21 select * from Has3

```

Results Messages

acc_num	date_created	cost	
1	3	2019-11-10	0
2	5	2019-11-10	0
3	7	2019-11-10	0
4	10	2019-11-10	0

acc_num	date_created	cost	
1	2	2019-11-10	0
2	6	2019-11-10	0
3	9	2019-11-10	0

assembly_id	acc_num
1	1
2	3
3	4

Ln 20, Col 1 (85 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 10 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 122 PM 11/10/2019

Type here to search

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - ⇒ FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_1 - disconnected ● SQLQuery\_2 - mart34...rt3422) ●

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```

16 select * from Account_Assembly
17 select * from Account_Department
18 select * from Account_Process
19 select * from Has1
20 select * from Has2
21 select * from Has3
22

```

Results Messages

process_id	acc_num
1	1
2	2
3	4

dept_num	acc_num
1	2
2	4
3	5
4	9

Ln 21, Col 19 (40 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 7 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 123 PM 11/10/2019

Type here to search

**Eclipse IDE - Project - Project\_Shell/src/Program.java**

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer: Project\_Shell

Program.java

```

1005 System.out.println( " /> Enter Job Type Number. 3
1006 System.out.println( " /> Enter Transaction Information");
1007 <
1008
1009 Enter Job No: 3
1010 Enter Job Type: 1)Cut 2)Paint 3)Fit Type Number. 3
1011 Enter Date for Job Started: 10/11/2019
1012 Enter Process Id for Job: 4
1013 Enter Assembly Id for Job: 2
1014 Insertions Successful
1015
1016 (1) Enter a new customer
1017 (2) Enter a new department
1018 (3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
1019 (4) Enter new Process_Id and Department Information
1020 (5) Create new Account and Setup
1021 (6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
1022 (7) Enter Job Completion Date and Type of Job
1023 (8) Enter Transaction Information
1024 (9) Display Assembly Cost; REQUIRED: Assembly_Id
1025 (10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
1026 (11) Display Processes and their Department; REQUIRED: Process_Id
1027 (12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
1028 (13) Display Customers based on category
1029 (14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
1030 (15) Change color of a Paint-Job
1031 (16) Import: Enter new customers from a data file until the file is empty
1032 (17) Export: Retrieve customers whose category is in a given range
1033 (18) Quit

```

Task List: Quick Access

Outline: Problems, Javadoc, Declaration, Console

Windows Taskbar: Type here to search, File, Edit, View, Help, SQLQuery\_2 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio, Run, Cancel, Disconnect, Change Connection, cs-dsa-4513-sql-db, Results, Messages, job\_no, date\_started, date\_completed, job\_no, date\_started, date\_completed, machine\_type, time\_used, material\_used, labor\_time, 13 rows, 00:00:00, mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db, 137 PM, 11/10/2019

Windows Taskbar: Type here to search, File, Edit, View, Help, SQLQuery\_1 - disconnected, SQLQuery\_2 - mart34...rt3422, Run, Cancel, Disconnect, Change Connection, cs-dsa-4513-sql-db, Results, Messages, job\_no, date\_started, date\_completed, job\_no, date\_started, date\_completed, machine\_type, time\_used, material\_used, labor\_time, 13 rows, 00:00:00, mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db, 132 PM, 11/10/2019

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```
24 select * from Job_Cut
25 select * from Job_Fit
26 select * from Job_Paint
27 select * from Assigned
28 select * from Manufactures
29
```

Results Messages

job_no	date_started	date_completed	labor_time
1	3	2019-10-11	NULL
2	7	2019-04-09	NULL
3	9	2019-10-25	NULL

job_no	date_started	date_completed	color	volume	labor_time
1	4	2019-09-11	NULL	NULL	NULL
2	6	2019-01-11	NULL	NULL	NULL
3	8	2019-10-30	NULL	NULL	NULL
4	10	2019-09-25	NULL	NULL	NULL

Ln 27, Col 1 (48 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 7 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 138 PM 11/10/2019

Type here to search

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```
24 select * from Job_Cut
25 select * from Job_Fit
26 select * from Job_Paint
27 select * from Assigned
28 select * from Manufactures
29
```

Results Messages

process_id	job_no
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	10

assembly_id	process_id	job_no
1	1	1
2	2	3
3	2	4
4	3	5
5	3	6

Ln 29, Col 1 (52 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 17 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 138 PM 11/10/2019

Type here to search

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

1005 System.out.println( " / / Enter Job Number and Date of Job ");
1006 System.out.println( " / / Enter Transaction Information " );
1007
1008
1009 Enter JobNo: 2
1010 Job Found...
1011 Enter date completed: 11/09/2019
1012 Enter Machine Type: Heavy
1013 Enter Time Used: 55
1014 Enter Material Used: Aluminium
1015 Enter Labor Time: 90
1016 Insertions Successful
1017
1018 (1) Enter a new customer
1019 (2) Enter a new department
1020 (3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
1021 (4) Enter new Process_Id and Department Information
1022 (5) Create new Account and Setup
1023 (6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
1024 (7) Enter Job Completion Date and Type of Job
1025 (8) Enter Transaction Information
1026 (9) Display Assembly Cost; REQUIRED: Assembly_Id
1027 (10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
1028 (11) Display Processes and their Department; REQUIRED: Process_Id
1029 (12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
1030 (13) Display Customers based on category
1031 (14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
1032 (15) Change color of a Paint-Job
1033 (16) Import: Enter new customers from a data file until the file is empty
1034 (17) Export: Retrieve customers whose category is in a given range
1035 (18) Quit
    
```

Type here to search

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

RESULTS

job_no	date_started	date_completed
1	2019-11-01	2019-11-03
2	2019-10-16	2019-11-09
3	2019-10-11	2019-10-30
4	2019-09-11	2019-09-30
5	2019-01-09	2019-03-11
6	2019-01-11	2019-11-10
7	2019-04-09	2019-04-11
8	2019-10-30	2019-11-05
9	2019-10-25	2019-10-26
10	2019-09-25	NULL

job_no	date_started	date_completed	machine_type	time_used	material_used	labor_time
1	2019-11-01	2019-11-03	Heavy	18	Steel	45
2	2019-10-16	2019-11-09	Heavy	55	Aluminium	90
3	2019-01-09	2019-03-11	Manuel	90	Copper	120

SQLQuery\_2 - mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db (mart3422) - Azure Data Studio

SQLQuery\_1 - disconnected SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```

22
23 select * from Job
24 select * from Job_Cut
25 select * from Job_Fit
26 select * from Job_Paint
27 select * from Assigned
28 select * from Manufactures
    
```

Results Messages

Type here to search

File Edit View Help

LN 25, Col 1 (42 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 13 rows 00:00:00 mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db 1:44 PM 11/10/2019

Azure Data Studio interface showing a query results grid. The results grid displays two tables:

	job_no	date_started	date_completed	labor_time
1	3	2019-10-11	2019-10-30	10
2	7	2019-04-09	2019-04-11	90
3	9	2019-10-25	2019-10-26	100

	job_no	date_started	date_completed	color	volume	labor_time
1	4	2019-09-11	2019-09-30	Blue	11	30
2	6	2019-01-11	2019-11-10	Gray	76	220
3	8	2019-10-30	2019-11-05	Green	20	80
4	10	2019-09-25	NULL	NULL	NULL	NULL

Eclipse IDE interface showing a Java application running. The application is printing transaction information to the console:

```

Program.java:1705 System.out.println("// Enter Job Completion Date and Type of Job");
Program.java:1706 System.out.println("// Enter Transaction Information");
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
8
Enter Transaction No: 2
Enter sup-cost: 255.69
Enter job no for transaction: 2
Enter account id: 3
Insertions/Updates Successful

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit

```

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_1 - disconnected ● SQLQuery\_2 - mart34...rt3422) ●

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```
30 select * from Job
31 select * from Account_Assembly
32 select * from Account_Process
33 select * from Account_Department
34 select * from [Transaction]
35 select * from Records
36 select * from Updates
```

Results Messages

acc_num	date_created	cost
1	2019-11-05	209.56
2	2019-11-10	30
3	2019-11-10	66.66

acc_num	date_created	cost
1	2019-11-10	20.1
2	2019-11-10	120.99
3	2019-11-10	666.66

acc_num	date_created	cost
1	2019-11-10	1200.99
2	2019-11-10	230.01
3	2019-11-10	500.76
4	2019-11-10	990.32

Ln 34, Col 1 (97 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 10 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 156 PM 11/10/2019

Type here to search

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_1 - disconnected ● SQLQuery\_2 - mart34...rt3422) ●

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```
32 select * from Account_Process
33 select * from Account_Department
34 select * from [Transaction]
35 select * from Records
36 select * from Updates
37
```

Results Messages

trans_no	sup_cost
1	209.56
2	255.69
3	666.66
4	74.25
5	122.23
6	30
7	45.45
8	65

job_no	trans_no
1	1
2	2
3	3
4	5
5	6
6	7

Ln 36, Col 1 (52 selected) Spaces: 4 UTF-8 CRLF SQL MSSQL 15 rows 00:00:00 mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db 156 PM 11/10/2019

Type here to search

A screenshot of the Azure Data Studio interface. The left sidebar shows a tree view of database objects under 'SERVERS'. The main area displays a query window with the following code:

```
32 select * from Account_Process  
33 select * from Account_Department  
34 select * from [transaction]  
35 select * from Records  
36 select * from Updates  
37
```

The results tab shows a table with two columns: 'trans\_no' and 'acc\_num'. The data is as follows:

	trans_no	acc_num
1	1	1
2	2	3
3	3	7
4	5	7
5	6	4
6	7	7
7	8	5

The status bar at the bottom shows the following information: Line 37, Col 1 (23 selected), Spaces: 4, UTF-8, CRLF, SQL, MSSQL, 7 rows, 00:00:00, mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db, 157 PM, 11/10/2019.

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Project\_Shell

```

928 //Option 3 Enter Assembly Info.
929 public static void InsertAssemblyInfo()
930 {
931     try
932     {
933         Connection connection = DriverManager.getConnection(url); //sql connection.
934
935         Scanner sc = new Scanner(System.in);
936         System.out.print("Enter Assembly Id: ");
937         int assId = sc.nextInt();
938         sc.nextLine();
939         System.out.print("Enter Date Ordered: ");
940         String dateOrdered = sc.nextLine();
941         System.out.print("Enter Assembly Details: ");
942         String details = sc.nextLine();
943         System.out.print("Enter Customer Name: ");
944         String custName = sc.nextLine();
945
946         /*
947             /datab...

```

Problems Javadoc Declaration Console

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 1:58:55 PM)

(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
9  
Enter Assembly Id: 1  
Assembly Id: 1; Costs: 209.56  
(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup

Type here to search

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

Project\_Shell

```

928 //Option 3 Enter Assembly Info.
929 public static void InsertAssemblyInfo()
930 {
931     try
932     {
933         Connection connection = DriverManager.getConnection(url); //sql connection.
934
935         Scanner sc = new Scanner(System.in);
936         System.out.print("Enter Assembly Id: ");
937         int assId = sc.nextInt();
938         sc.nextLine();
939         System.out.print("Enter Date Ordered: ");
940         String dateOrdered = sc.nextLine();
941         System.out.print("Enter Assembly Details: ");
942         String details = sc.nextLine();
943         System.out.print("Enter Customer Name: ");
944         String custName = sc.nextLine();
945
946         /*
947             /datab...

```

Problems Javadoc Declaration Console

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 1:58:55 PM)

(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
9  
Enter Assembly Id: 3  
Assembly Id: 3; Costs: 30.0  
(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup

Type here to search

The screenshot shows the Eclipse IDE interface with a Java application running in the Console view. The title bar reads "Project - Project\_Shell/src/Program.java - Eclipse IDE". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help. The Package Explorer shows a single project "Project\_Shell". The Console tab is active, displaying the output of the Java application:

```
928 //Option 3 Enter Assembly Info.
929
930 Enter Assembly Id: 4
931 Assembly Id: 4; Costs: 66.66
932
933 (1) Enter a new customer
934 (2) Enter a new department
935 (3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
936 (4) Enter new Process_Id and Department Information
937 (5) Create new Account and Setup
938 (6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Started
939 (7) Enter Job Completion Date and Type of Job
940 (8) Enter Transaction Information
941 (9) Display Assembly Cost; REQUIRED: Assembly_Id
942 (10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
943 (11) Display Processes and their Department; REQUIRED: Process_Id
944 (12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
945 (13) Display Customers based on category
946 (14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
947 (15) Change color of a Paint-Job
948 (16) Import: Enter new customers from a data file until the file is empty
949 (17) Export: Retrieve customers whose category is in a given range
950 (18) Quit
```

```

Project - Project_Shell/src/Program.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Task List Quick Access
Project_Shell
Program.java
928 //Option 3 Enter Assembly Info.

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 1:58:55 PM)
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
18
Enter Department Number: 2
Enter Completion Date: 09/30/2019
Department: 2; Total Labor Time: 30

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit


```

```

Project - Project_Shell/src/Program.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Task List Quick Access
Project_Shell
Program.java
928 //Option 3 Enter Assembly Info.

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 1:58:55 PM)
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
18
Enter Department Number: 11
Enter Completion Date: 11/10/2019
Department: 11; Total Labor Time: 220

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit


```

The screenshot shows the Eclipse IDE interface with a Java application running in the Console view. The application is a menu-based system for managing customer data. The user has selected option 3 to enter assembly information. The application then prompts for a department number (14) and completion date (10/25/2019). It displays the entered data and then lists all available options again.

```
Project - Project_Shell/src/Program.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Task List Quick Access
Project.Shell
Program.java
928 //Option 3 Enter Assembly Info.
...
Problems Javadoc Declaration Console
Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 1:58:55 PM)
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
18
Enter Department Number: 14
Enter Completion Date: 10/25/2019
Department: 14; Total Labor Time: 0

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Started
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
```

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)

```

(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
11
Enter Assembly Id: 3
Processes passed by Assembly: 3
Process Id: 5; Process Data: passed
Process Id: 6; Process Data: passed

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit

```

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)

```

(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
11
Enter Assembly Id: 5
Processes passed by Assembly: 5
Process Id: 10; Process Data: passed

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit

```

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Project\_Shell

Program.java

```
processList.add(resultset.getString("process_id")); //populate the list.
```

Problems Javadoc Declaration Console

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)

(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
11  
Enter Assembly Id: 1  
Processes passed by Assembly: 1  
Process Id: 1; Process Data: passed  
(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Type here to search 226 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```
300 //Create String from jobNos list. This will be the WHERE clause.
```

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)

(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
12  
Enter Date Completed: 11/10/2019  
Enter Department Supervising: 11  
Job No: 6; Date Started: 2019-01-11; Date Completed: 2019-11-10; Color: Gray; Volume: 76; Labor Time: 220  
  
(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```
300 //Create String from jobNos list. This will be the WHERE clause.
```

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)

(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
12  
Enter Date Completed: 03/11/2019  
Enter Department Supervising: 10  
Job No: 5; Date Started: 2019-01-09; Date Completed: 2019-03-11; Machine Type: Manuel; Time Used: 90; Material Used: Copper; Lat  
  
(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

The screenshot shows the Eclipse IDE interface with a Java application running in the terminal. The terminal window displays a menu of 18 options related to job management, followed by user input and system responses.

Program.java //Create String from jobNos list. This will be the WHERE clause.

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)

(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
12  
Enter Date Completed: 10/30/2019  
Enter Department Supervising: 8  
Job No: 3; Date Started: 2019-10-11; Date Completed: 2019-10-30; Labor Time: 10  
(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

```

Project - Project_Shell/src/Program.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Task List Quick Access
Project_Shell

Program.java
processus.add(resultset.getint("process_id")); //populate the list.

Problems Javadoc Declaration Console
Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)
(18) Quit
13
Enter lower category: 1
Enter high category: 5
Customer: John
Customer: Louis
Customer: Maranda
Customer: Sandra
Customer: Tommy
Customer: Treyson

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit

```

```

Project - Project_Shell/src/Program.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Task List Quick Access
Project_Shell

Program.java
processus.add(resultset.getint("process_id")); //populate the list.

Problems Javadoc Declaration Console
Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
13
Enter lower category: 4
Enter high category: 5
Customer: Louis
Customer: Maranda
Customer: Treyson

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit

```

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```
processList.add(resultset.getString("process_id")); //populate the list.
```

Problems Javadoc Declaration Console

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)

(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

13  
Enter lower category: 7  
Enter high category: 10  
Customer: Gage  
Customer: Lisa

(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Type here to search 228 PM 11/10/2019

```

Project - Project_Shell/src/Program.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Task List Quick Access
Project_Shell

Program.java
processus.add(resultset.getint("process_id")); //populate the list.

Problems Javadoc Declaration Console
Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
14
Enter low job no: 4
Enter high job no: 10
Deletions Complete.

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit

```

```

Project - Project_Shell/src/Program.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer Task List Quick Access
Project_Shell

Program.java
processus.add(resultset.getint("process_id")); //populate the list.

Problems Javadoc Declaration Console
Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit
14
Enter low job no: 10
Enter high job no: 11
Deletions Complete.

(1) Enter a new customer
(2) Enter a new department
(3) Enter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
(4) Enter new Process_Id and Department Information
(5) Create new Account and Setup
(6) Enter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
(7) Enter Job Completion Date and Type of Job
(8) Enter Transaction Information
(9) Display Assembly Cost; REQUIRED: Assembly_Id
(10) Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed
(11) Display Processes and their Department; REQUIRED: Process_Id
(12) Display Completed Jobs; REQUIRED: Date_Completed and Department_Number
(13) Display Customers based on category
(14) Delete Cut-Jobs; REQUIRED: Job_no and Date_Range
(15) Change color of a Paint-Job
(16) Import: Enter new customers from a data file until the file is empty
(17) Export: Retrieve customers whose category is in a given range
(18) Quit

```

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```
processus.add(resultset.getint("process_id")); //populate the list.
```

Problems Javadoc Declaration Console

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:18:58 PM)

(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

14

Enter low job no: 2  
Enter high job no: 10  
Deletions Complete.

(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Type here to search

File Edit View Help

CONNECTIONS

SERVICES

SQLQuery\_1 - disconnected SQLQuery\_2 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```
35 select * from Records
36 select * from Updates
37
38 select * from Supervises
39 select * from Job_Cut
40
41 SELECT * from Manufacturers
```

Results Messages

job_no	date_started	date_completed	machine_type	time_used	material_used	labor_time
1	2019-11-01	2019-11-03	Heavy	18	Steel	45

File Edit View Help

IN 40, Col 1 (23 selected) Spaces:4 CRLF SQL MSSQL 1 rows 00:00:00 mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db ENG 230 PM 11/10/2019

A screenshot of the Azure Data Studio interface. The left sidebar shows a tree view of database objects under the 'SERVERS' node, including tables like 'dbo.Acted', 'dbo.Assembly', etc., and system objects like 'Orders', 'Keys', 'Constraints', 'Triggers', 'Indexes', and 'Statistics'. The main area displays a query editor with the following SQL code:

```
35 select * from Records
36 select * from Updates
37
38 select * from Supervises
39 select * from Job_Paint
40
41 SELECT * from Manufacturers
```

The results tab shows a table with the following data:

	job_no	date_started	date_completed	color	volume	labor_time
1	4	2019-09-11	2019-09-30	Blue	11	30
2	6	2019-01-11	2019-11-10	Gray	76	220
3	8	2019-10-30	2019-11-05	Green	20	80
4	10	2019-09-25	NULL	NULL	NULL	NULL

The bottom status bar shows the following information: Line 40, Col 1 (25 selected), Spaces:4, UTF-8, CRLF, SQL, MSSQL, 4 rows, 00:00:00, mart3422-sql-server.database.windows.net : cs-dsa-4513-sql-db, 23:30 PM, 11/10/2019.

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

193     try {
194         Connection connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/test", "root", "root");
195     } catch (SQLException e) {
196         e.printStackTrace();
197     }
198 }
```

Problems Javadoc Declaration Console

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:32:18 PM)

(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
15  
Enter Job No: 4  
Update Color of Job: Red  
Update Complete.

(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processed and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

193     try {
194         Connection connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/test", "root", "root");
195     } catch (SQLException e) {
196         e.printStackTrace();
197     }
198 }
```

Problems Javadoc Declaration Console

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:32:18 PM)

(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
15  
Enter Job No: 6  
Update Color of Job: Purple  
Update Complete.

(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processed and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Problems Javadoc Declaration Console

Program.java

```

try
{
    Connection connection = DriverManager.getConnection("url", "username", "password");
}

```

Task List    Outline    main(String[] args) void

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:32:18 PM)

(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
15  
Enter Job No: 8  
Update Color of Job: silver  
Update Complete.  
  
(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processed and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Type here to search    233 PM 11/10/2019

File Edit View Help

CONNECTIONS

Servers

- > dbo.Acted
- > dbo.Assembly
- > dbo.Assigned
- > dbo.Customers
- > dbo.Department
- > dbo.Director
- > dbo.Has1
- > dbo.Has2
- > dbo.Has3
- > dbo.Job
- > dbo.Job\_Cut
- > dbo.Job\_Fit
- > dbo.Job\_Paint
- > dbo.Manufactures
- > dbo.Movie
- > dbo.Orders
- > Columns
- > Keys
  - PK\_Orders\_F0E9293ED96C6...
  - FK\_Orders\_assembly\_2739...
- > Constraints
- > Triggers
- > Indexes
- > Statistics

AZURE

SQLQuery\_2 - mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db (mart3422) - Azure Data Studio

SQLQuery\_1 - disconnected    SQLQuery\_2 - mart34...rt3422

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```

35 select * from Records
36 select * from Updates
37
38 select * from Supervises
39 select * from Job_Paint
40
41 SELECT * from Manufactures

```

Results    Messages

	job_no	date_started	date_completed	color	volume	labor_time
1	4	2019-09-11	2019-09-30	Red	11	30
2	6	2019-01-11	2019-11-10	Purple	76	220
3	8	2019-10-30	2019-11-05	Silver	20	80
4	10	2019-09-25	NULL	NULL	NULL	NULL

Type here to search    233 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

String details = sc.nextLine();
Customer cust = null;
System.out.println("Enter Customer Name: ");

```

Problems Javadoc Declaration Console

Program Java Application C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 12:56:25 PM)

(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
16  
Input file name (Include .csv): cust.csv  
Import Successful  
(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processes and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Type here to search

File Edit View Help

CONNECTIONS

SERVICES

SQLQuery\_1 - disconnected SQLQuery\_2 - mart3422-sql-server.database.windows.net.cs-dsa-4513-sql-db (mart3422) - Azure Data Studio

Run Cancel Disconnect Change Connection cs-dsa-4513-sql-db

```

1 Select * from Customers
2
3 select * from Department
4
5 Select * from Assembly
6 Select * from Orders
7 Select * from Supervises
8 Select * from Job
9 select * from Processes
10 select * from Process_Cut
11 select * from Process_Paint

```

Results Messages

	cust_name	address	category
1	Gage	6969 Dr.	9
2	John	101 Beaumont Dr.	1
3	Lisa	8274 S. St.	10
4	Louis	4545 Will Rd.	5
5	Maranda	8273 S. St.	4
6	Sandra	8989 Wind Dr.	3
7	Tommy	87 Rd.	2
8	Treyson	1234 St.	5

Type here to search

Ln 2, Col 1 (25 selected) Spaces:4 CRLF SQL MSSQL 8 rows 00:00:00 mart3422-sql-server.database.windows.net:cs-dsa-4513-sql-db 12:59 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```

193     try {
194         Connection connection = DriverManager.getConnection("url", "username", "password");
195     } catch (SQLException e) {
196         e.printStackTrace();
197     }
198 }
```

Problems Javadoc Declaration Console

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:32:18 PM)

(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit  
17  
Enter lower category: 1  
Enter high category: 5  
File customers22176.csv created.

(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processed and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

Type here to search    234 PM 11/10/2019

customers22176.csv - Excel

File Home Insert Draw Page Layout Formulas Data Review View Help Share Comments

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Name	Address	Category														
2	John	101 Beaumont Dr.	1														
3	Louis	4545 Will Rd.	5														
4	Maranda	8273 S. St.	4														
5	Sandra	8989 Wind Dr.	3														
6	Tommy	87 Rd.	2														
7	Treyson	1234 St.	5														
8																	
9																	
10																	
11																	
12																	
13																	
14																	
15																	
16																	
17																	
18																	
19																	
20																	
21																	

customers22176

Type here to search    235 PM 11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer    Quick Access

Program.java

```
try
{
    Connection connection = DriverManager.getConnection("url", "username", "password");
}
catch (SQLException e)
{
    e.printStackTrace();
}
```

Problems Javadoc Declaration Console

<terminated> Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 2:32:18 PM)

(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

17  
Enter lower category: 1  
Enter high category: 5  
File customers22176.csv created.

(1) Enter a new customer  
(2) Enter a new department  
(3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered  
(4) Enter new Process\_Id and Department Information  
(5) Create new Account and Setup  
(6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start  
(7) Enter Job Completion Date and Type of Job  
(8) Enter Transaction Information  
(9) Display Assembly Cost; REQUIRED: Assembly\_Id  
(10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed  
(11) Display Processed and their Department; REQUIRED: Process\_Id  
(12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number  
(13) Display Customers based on category  
(14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range  
(15) Change color of a Paint-Job  
(16) Import: Enter new customers from a data file until the file is empty  
(17) Export: Retrieve customers whose category is in a given range  
(18) Quit

18  
Exiting...  
Have a good day!

Type here to search

234 PM  
11/10/2019

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

259     //Option 3 Enter Assembly Info.
260     public static void InsertAssemblyInfo()
261     {
262         try
263         {
264             ...
265         }
266     }

```

Problems Javadoc Declaration Console

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 5, 2019, 9:30:56 AM)

- (4) Enter new Process\_Id and Department Information
- (5) Create new Account and Setup
- (6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start
- (7) Enter Job Completion Date and Type of Job
- (8) Enter Transaction Information
- (9) Display Assembly Cost; REQUIRED: Assembly\_Id
- (10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed
- (11) Display Processes and their Department; REQUIRED: Process\_Id
- (12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number
- (13) Display Customers based on category
- (14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range
- (15) Change color of a Paint-Job
- (16) Import: Enter new customers from a data file until the file is empty
- (17) Export: Retrieve customers whose category is in a given range
- (18) Quit

Please input Customer name: Jerry

Please input Customer address: 3421 Blink Rd.

Please input Customer category (1-10): 11

Error in InsertCustomer. Error: com.microsoft.sqlserver.jdbc.SQLServerException: The INSERT statement conflicted with the CHECK constraint 'CustomerCategory'. The conflict occurred in database 'CustomerDB', table 'Customer'. The statement has been terminated.

Type exit to go back to options

Please input Customer name:

Project - Project\_Shell/src/Program.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer Task List Quick Access

Program.java

```

1069     System.out.println("// Enter Job Completion Date and Type of Job");
1070     System.out.println("// Enter Transaction Information");

```

Problems Javadoc Declaration Console

Program [Java Application] C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 1:11:54 PM)

- (17) Export: Retrieve customers whose category is in a given range
- (18) Quit

7

Enter JobNo: 4

Job Found...

Enter date completed: 01/01/2001

Enter Paint Color: Red

Enter Volume: 15

Enter Labor Time: 30

Error in ConfirmJobCompletion(). Error: com.microsoft.sqlserver.jdbc.SQLServerException: The UPDATE statement conflicted with the CHECK constraint 'CustomerCategory'. The conflict occurred in database 'CustomerDB', table 'Customer'. The statement has been terminated.

- (1) Enter a new customer
- (2) Enter a new department
- (3) Enter new Assembly; REQUIRED: Customer\_Name, Assembly\_Details, Assembly\_Id, Date\_Ordered
- (4) Enter new Process\_Id and Department Information
- (5) Create new Account and Setup
- (6) Enter new Job; REQUIRED: Job\_no, Assembly\_id, Process\_id, Date\_of\_Job\_Start
- (7) Enter Job Completion Date and Type of Job
- (8) Enter Transaction Information
- (9) Display Assembly Cost; REQUIRED: Assembly\_Id
- (10) Display total Labor time of Job; REQUIRED: Department\_Number and Date\_Completed
- (11) Display Processes and their Department; REQUIRED: Process\_Id
- (12) Display Completed Jobs; REQUIRED: Date\_Completed and Department\_Number
- (13) Display Customers based on category
- (14) Delete Cut-Jobs; REQUIRED: Job\_no and Date\_Range
- (15) Change color of a Paint-Job
- (16) Import: Enter new customers from a data file until the file is empty
- (17) Export: Retrieve customers whose category is in a given range
- (18) Quit

The screenshot shows the Eclipse IDE interface. In the center, the code editor displays `Program.java` with several print statements. The `Console` view at the bottom shows the execution of the program, displaying user input and system output. The output includes:

```

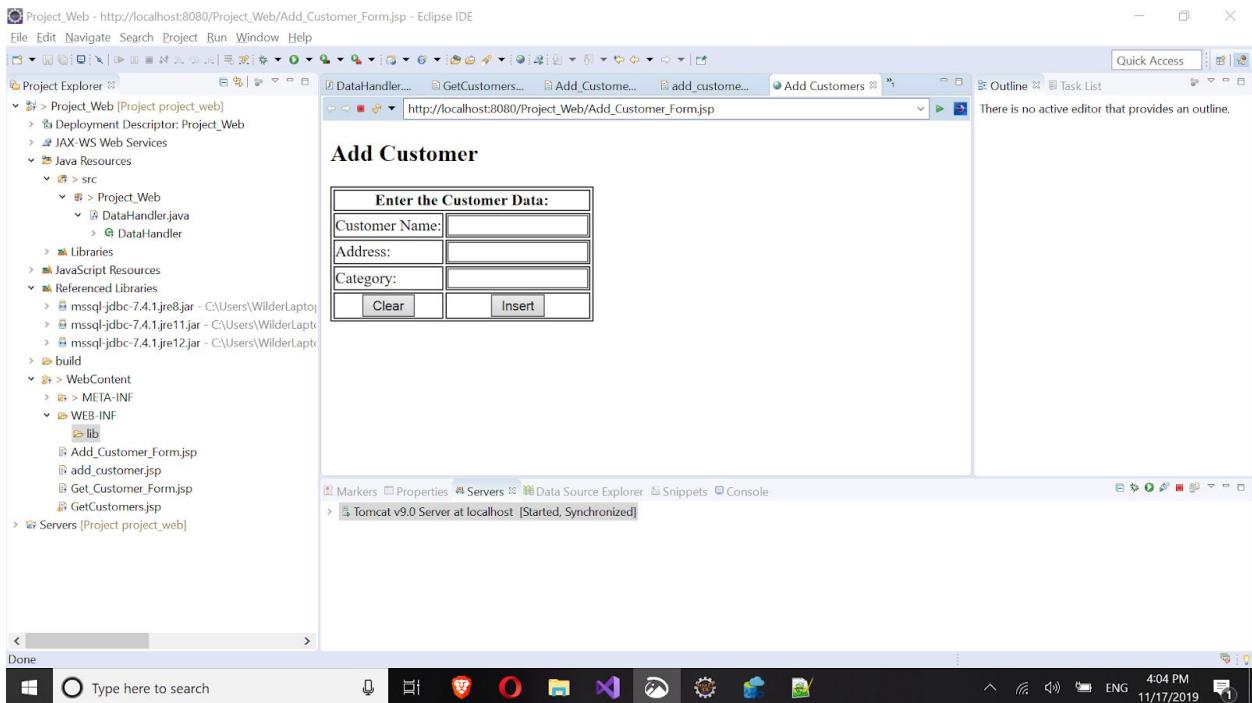
Program Java Application|C:\Program Files\Java\jdk-13\bin\javaw.exe (Nov 10, 2019, 3:13:05 PM)
Export: Retrieve customers whose category is in a given range
Quit

- Transaction No: 13
- sup-cost: -1897.21
- job no for transaction: 3
- account id: 6
- in InsertTransactionInfo(). Error: com.microsoft.sqlserver.jdbc.SQLServerException: The INSERT statement conflicted with the CI
nter a new customer
nter a new department
nter new Assembly; REQUIRED: Customer_Name, Assembly_Details, Assembly_Id, Date_Ordered
nter new Process_Id and Department Information
reate new Account and Setup
nter new Job; REQUIRED: Job_no, Assembly_id, Process_id, Date_of_Job_Start
nter Job Completion Date and Type of Job
nter Transaction Information
isplay Assembly Cost; REQUIRED: Assembly_Id
Display total Labor time of Job; REQUIRED: Department_Number and Date_Completed

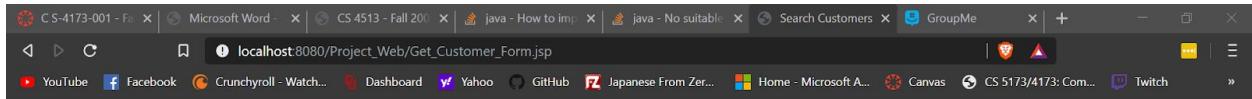
```

## 7) Web Application

Below is the web service running in Eclipse.

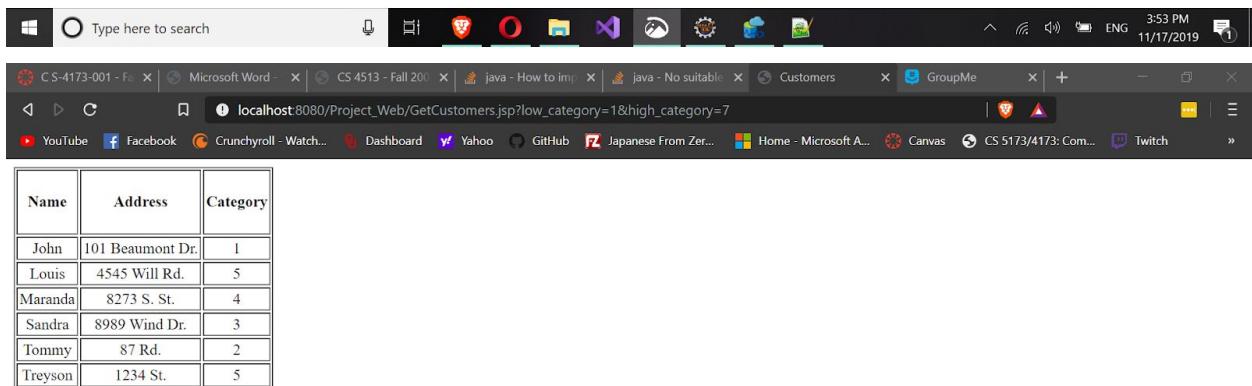


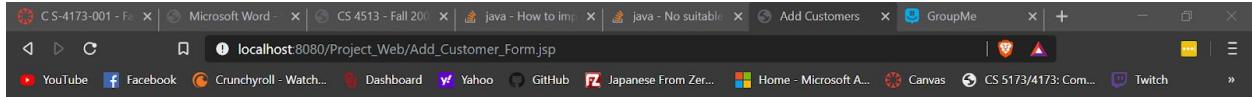
Below are images of Query 1 and 13 being run back to back in a browser.



### Search Customers by Category

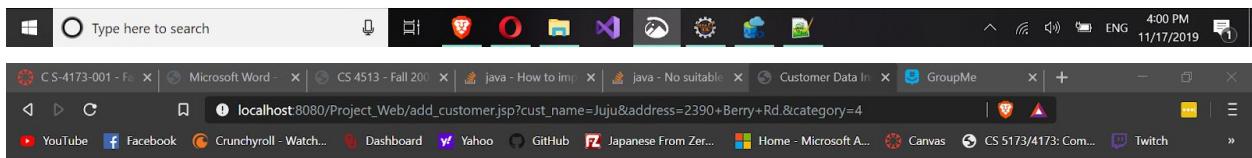
Enter the Customer Categories:	
Low Category:	<input type="text" value="1"/>
High Category:	<input type="text" value="7"/>
<input type="button" value="Clear"/>	<input type="button" value="Search"/>





## Add Customer

Enter the Customer Data:	
Customer Name:	<input type="text" value="Juju"/>
Address:	<input type="text" value="2390 Berry Rd."/>
Category:	<input type="text" value="4"/>
<input type="button" value="Clear"/>	<input type="button" value="Insert"/>



### The Customer Info:

- Customer Name: Juju
- Address: 2390 Berry Rd.
- Category: 4

Was successfully inserted.

[See Customers.](#)



Name	Address	Category
John	101 Beaumont Dr.	1
Juju	2390 Berry Rd.	4
Louis	4545 Will Rd.	5
Maranda	8273 S. St.	4
Sandra	8989 Wind Dr.	3
Tommy	87 Rd.	2
Treysen	1234 St.	5

Below is the code for the DataHandler and 4 jsp forms.

```
package Project_Web;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.DriverManager;
import java.sql.PreparedStatement;

public class DataHandler
{
    private Connection conn;
    // Azure SQL connection credentials
    private String server = "mart3422-sql-server.database.windows.net";
    private String database = "cs-dsa-4513-sql-db";
    private String username = "mart3422";
    private String password = "Atticus666418!";
    // Resulting connection string
    final private String url =
        String.format("jdbc:sqlserver://%s:1433;database=%s;user=%s;" +
                      "password=%s;encrypt=true;trustServerCertificate=false;" +
                      "hostNameInCertificate=*.database.windows.net;"
```

```
        + "loginTimeout=30;"  
        server, database, username, password);  
// Initialize and save the database connection  
private void getDBConnection() throws SQLException  
{  
    if (conn != null)  
    {  
        System.out.println("HERE");  
        return;  
    }  
    System.out.println("HERE2");  
    try  
    {  
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        this.conn = DriverManager.getConnection(url);  
        System.out.println("HERE3");  
    }  
    catch (ClassNotFoundException e)  
    {  
        // TODO Auto-generated catch block  
        e.printStackTrace();  
    }  
}  
  
// Return the result of selecting everything from the movie_night table  
public ResultSet getAllCustomers() throws SQLException  
{  
    getDBConnection();  
    final String sqlQuery = "SELECT * FROM Customers";  
    final PreparedStatement stmt = conn.prepareStatement(sqlQuery);  
    return stmt.executeQuery();  
}  
  
// Return the result of selecting everything from the movie_night table  
public ResultSet GetCustomers(String lowCat, String highCat)  
    throws SQLException  
{  
    getDBConnection();  
    final String sqlQuery = "SELECT * FROM Customers WHERE"  
        + " category between " + lowCat + " AND " + highCat  
        + " ORDER by cust_name";  
  
    System.out.println(sqlQuery);
```

```

        final PreparedStatement stmt = conn.prepareStatement(sqlQuery);
        return stmt.executeQuery();
    }

    // Inserts a record into the movie_night table with the given attribute values
    public boolean addCustomer(String name, String address, String category)
            throws SQLException
    {
        System.out.println("In addCustomer");
        getDBConnection(); // Prepare the database connection
        System.out.println("Made Connection");
        // Prepare the SQL statement
        final String sqlQuery = "INSERT INTO Customers VALUES (" + name + "", ""
                + address + ", " + category + ")";
        final PreparedStatement stmt = conn.prepareStatement(sqlQuery);
        System.out.println("Prepared Query");
        // Replace the '?' in the above statement with the given attribute values
        //stmt.setString(1, name);
        //stmt.setString(2, address);
        //stmt.setString(3, category);
        // Execute the query, if only one record is updated, then we indicate success by
        return stmt.executeUpdate() == 1;
    }
}

```

Web forms:

Add\_Customer\_Form:

```

<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>Add Customers</title>
    </head>
    <body>
        <h2>Add Customer</h2>
        <!--
        Form for collecting user input for the new movie_night record.
        Upon form submission, add_movie.jsp file will be invoked.
        -->
        <form action="add_customer.jsp">
            <!-- The form organized in an HTML table for better clarity. -->
<table border=1>
<tr>

```

```

<th colspan="2">Enter the Customer Data:</th>
</tr>
<tr>
    <td>Customer Name:</td>
    <td><div style="text-align: center;">
        <input type=text name=cust_name>
    </div></td>
</tr>
<tr>
    <td>Address:</td>
    <td><div style="text-align: center;">
        <input type=text name=address>
    </div></td>
</tr>
<tr>
    <td>Category:</td>
    <td><div style="text-align: center;">
        <input type=text name=category>
    </div></td>
</tr>
<tr>
    <td><div style="text-align: center;">
        <input type=reset value=Clear>
    </div></td>
    <td><div style="text-align: center;">
        <input type=submit value=Insert>
    </div></td>
</tr>
</table>
</form>
</body>
</html>

```

Add\_Customer.jsp:

```

<%@ page language="java" contentType="text/html; charset=UTF8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Customer Data Insert</title>
</head>
<body>

```

```

<%@page import="Project_Web.DataHandler"%>
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Array"%>
<%
// The handler is the one in charge of establishing the connection.
DataHandler handler = new DataHandler();
// Get the attribute values passed from the input form.
String name = request.getParameter("cust_name");
String address = request.getParameter("address");
String category = request.getParameter("category");

/*
 * If the user hasn't filled out all the time, movie name and duration. This is very
simple
checking.
*/
if (name.equals("") || address.equals("") || category.equals(""))
{
    response.sendRedirect("Add_Customer_Form.jsp");
}
else
{
    // Now perform the query with the data from the form.
    boolean success = handler.addCustomer(name, address, category);
    if (!success)
    { // Something went wrong
        %>
        <h2>There was a problem inserting the customer</h2>
    <%
    }
    else
    { // Confirm success to the user
        %>
        <h2>The Customer Info:</h2>
        <ul>
        <li>Customer Name: <%=name%></li>
        <li>Address: <%=address%></li>
        <li>Category: <%=category%></li>
        </ul>
        <h2>Was successfully inserted.</h2>
        <a href="Get_Customer_Form.jsp">See Customers.</a>
    <%
    }
}

```

```
    }

    %>
</body>
</html>

Get_Customer_Form:
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Search Customers</title>
</head>
<body>
<h2>Search Customers by Category</h2>
<!--
Form for collecting user input for the new movie_night record.
Upon form submission, add_movie.jsp file will be invoked.
-->
<form action="GetCustomers.jsp">
    <!-- The form organized in an HTML table for better clarity. -->
<table border=1>
<tr>
    <th colspan="2">Enter the Customer Categories:</th>
</tr>
<tr>
    <td>Low Category:</td>
    <td><div style="text-align: center;">
        <input type=text name=low_category>
    </div></td>
</tr>
<tr>
    <td>High Category:</td>
    <td><div style="text-align: center;">
        <input type=text name=high_category>
    </div></td>
</tr>
<tr>
    <td><div style="text-align: center;">
        <input type=reset value=Clear>
    </div></td>
    <td><div style="text-align: center;">
        <input type=submit value=Search>
    </div></td>
</tr>
```

```

        </div></td>
    </tr>
</table>
</form>
</body>
</html>

```

GetCustomers.jsp:

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Customers</title>
</head>
<body>
<%@page import="Project_Web.DataHandler"%>
<%@page import="java.sql.ResultSet"%>
<%
// We instantiate the data handler here, and get all the movies from the database
final DataHandler handler = new DataHandler();
final ResultSet customers = handler.GetCustomers
    (request.getParameter("low_category"),
request.getParameter("high_category"));
%>
<!-- The table for displaying all the movie records -->
<table cellspacing="2" cellpadding="2" border="1">
<tr> <!-- The table headers row -->
<td align="center">
<h4>Name</h4>
</td>
<td align="center">
<h4>Address</h4>
</td>
<td align="center">
<h4>Category</h4>
</td>
</tr>
<%
while(customers.next()) { // For each movie_night record returned...
// Extract the attribute values for every row returned

```

```
final String name = customers.getString("cust_name");
final String address = customers.getString("address");
final String category = customers.getString("category");
System.out.println(name + ", " + address + ", " + category);
out.println("<tr>"); // Start printing out the new table row
out.println( // Print each attribute value
"<td align=\"center\">" + name +
"</td><td align=\"center\"> " + address +
"</td><td align=\"center\"> " + category + "</td>");
out.println("</tr>");
}
%>
</table>
</body>
</html>
```