

## # CLIENT LEVEL USER command script

1. Command 1: Query: list the supplier number and supplier name for those suppliers who ship every green part.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **client-level user**.

Please enter any valid SQL query or update below:

```
select snum, sname
from suppliers
where not exists
(
  select *
  from parts
  where parts.color = "green"
  and not exists
  (
    select *
    from shipments
    where shipments.snum = suppliers.snum
    and shipments.pnum = parts.pnum
  )
)
```

Execute Command    Reset    Clear Results

All execution results will appear below this line:

Database Results:

snum	sname
S6	Nicola Gianniberti

2. # Command 2: This command will not execute for the client-level user.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **client-level user**.

Please enter any valid SQL query or update below:

```
update jobs
set city = "Madrid"
where city = "Paris";
```

Execute Command    Reset    Clear Results

All execution results will appear below this line:

Database Results:

Update access Denied  
for Client- Level User

Command 3: This is a basic select command using an aggregation operator

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **client-level** user.

Please enter any valid SQL query or update below:

```
select count(*) as total_shipments_with_quantity_over_100
from shipments
where quantity > 100;
```

Execute Command   Reset   Clear Results

All execution results will appear below this line:

Database Results:

total_shipments_with_quantity_over_100
22

4. # Command 4: This command will not execute for the client-level user.

# Output should show the message indicating that the command was denied to the client-level user.

replace into suppliers values ("S1", "Michael Schumacher",4,"Frankfurt");

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **client-level** user.

Please enter any valid SQL query or update below:

```
replace into suppliers values ("S1", "Michael Schumacher",4,"Frankfurt");
```

Execute Command   Reset   Clear Results

All execution results will appear below this line:

Database Results:

replace access Denied for Client- Level User
---

## # ROOT LEVEL USER command script

1. # Command 1: Query: list the supplier number and supplier name for those suppliers who ship every part.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

```
select snum, sname
from suppliers
where not exists
(
  select *
  from parts
  where parts.color = "green"
  and not exists
  (
    select *
    from shipments
    where shipments.snum = suppliers.snum
    and shipments.pnum = parts.pnum
  )
);
```

Execute Command Reset Clear Results

All execution results will appear below this line:

Database Results:

snum	sname
S6	Nicola Gianniberti

2. # Command 1:

# Command 2A: Query: list all supplier information.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

```
select *
from suppliers;
```

Execute Command Reset Clear Results

All execution results will appear below this line:

Database Results:

snum	sname	status	city
S1	Michael Schumacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Hirtak	7	Paris
S12	Eddy Merckx	1	Brussels
S13	Caroline Evansgrat	3	Cape Town
S14	Adriana Lima	4	Sao Paulo
S15	Jennifer Lawrence	8	Overseas
S16	Fernando Alonso	4	Madrid
S17	Rubens Barrichello	3	Sao Paulo
S18	Toni Blomqvist	2	Brussels
S19	John McEnroe	1	Enkhai
S2	Juan Pablo Montoya	4	Interlago
S20	Daniel Ricci	2	Milan
S21	Lizette Arnold	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Delirich Thureau	1	Berlin
S13	Berni Schuster	2	Berlin
S33	Ruf Adag	3	Berlin
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	4	London
S58	Margarete Voss	8	Zandvoort
S6	Nicola Gianniberti	2	Milan
S7	Christian Albers	3	Oslo
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Raikkonen	2	Helsinki

## # Command 2B: add a new record to shipments table (S5, P6, J7, 400)

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

```
insert into shipments
values ('S5', 'P6', 'J7', 400);
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

The statement executed successfully. 1 row(s) affected.

Business Logic Detected! - Updating Supplier Status

Business logic updated 1 supplier status mark.

## # Command 2C: list all supplier information.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

```
select *
from suppliers;
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

snam	sname	status	city
S1	Michael Schumacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Eddy Merckx	1	Brussels
S13	Candice Swanepoel	3	Cape Town
S14	Adriana Lima	4	Sao Paulo
S15	Jennifer Lawrence	6	Oswestry
S16	Fernando Alonso	4	Madrid
S17	Rubens Barrichello	3	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Massauer	1	Enkloz
S2	Juan Pablo Montoya	4	Interlago
S20	Dario Rossi	2	Milan
S21	Lizzie Armitstead	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	1	Berlin
S32	Bernd Schneider	2	Berlin
S33	Ralf Ardig	3	Berlin
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	9	London
S56	Matthias Vos	8	Zandvoort
S6	Nicola Girotti	2	Viterbo
S7	Christian Abani	3	Ottawa
S8	Giovanni Piccinelli	3	Milan
S9	Kimi Raikkonen	2	Heinola

# Command 3 is a multi-part that does not cause the business logic to trigger

# Command 3A list all supplier information

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a root-level user.

Please enter any valid SQL query or update below:

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

snum	sname	status	city
S1	Michael Schumacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Edy Merckx	1	Brussels
S13	Candice Swanepoel	3	Cape Town
S14	Adriana Lima	4	Sao Paulo
S15	Jennifer Lawrence	6	Owensboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barrichello	3	Sao Paulo
S18	Toni Bonner	2	Brussels
S19	John Massoner	1	Tokyo
S2	Juan Pablo Montoya	4	Interlago
S20	Danilo Rossi	2	Milan
S21	Lizzie Armstrong	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Detrich Thunau	1	Berlin
S32	Bernd Schneider	2	Berlin
S33	Ralf Ailing	3	Berlin
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	9	London
S56	Martina Vitas	8	Zurichhof
S6	Nicola Giannini	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Paschella	3	Milan
S9	Kimi Raikkonen	2	Helsinki

# Command 3B: add a new record to the supplier table (S39, Candice Swanepoel, 10, Cardiff)

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a root-level user.

Please enter any valid SQL query or update below:

```
insert into suppliers
values ('S39','Candice Swanepoel',10,'Cardiff');
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

The statement executed successfully. 1 row(s) affected.

Business Logic Triggered! No status marks updated though.

# Command 3C: list all supplier information.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

Product ID  
From Suppliers

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

idnum	name	status	city
S1	Michael Schumacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Villeneuve	7	Paris
S12	Eddy Merckx	1	Brussels
S13	Carlos Sainz	3	Cape Town
S14	Kristen Lima	4	Sao Paulo
S15	Jennifer Lawrence	6	Osaka
S16	Fernando Alonso	4	Madrid
S17	Rubens Barrichello	3	Sao Paulo
S18	Tom Brannen	2	Brussels
S19	Johan Heisenberg	1	Bahia
S2	Juan Pablo Montoya	4	Interlaken
S20	Dario Franchitti	2	Milan
S21	Lizzy Armitstead	1	Hempstead
S22	Jan Ulrich	5	Bonn
S3	Dieter Thues	1	Berlin
S32	Berni Schuster	2	Berlin
S33	Ralf Aulig	3	Berlin
S39	Carlos Sainz	10	Cardiff
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	9	London
S58	Matthias Voss	8	Zurich
S6	Nicole Greenhalgh	2	Milan
S7	Christian Klien	3	Osaka
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Raikkonen	2	Helsinki

# Command 3D: add a new record to shipments table (S39, P3, J1, 20), business logic will trigger

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

Insert into shipments  
values ('S39','P3','J1', 20);

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

The statement executed successfully. 1 row(s) affected.  
Business Logic Triggered! No status marks updated though.

## # Command 3E: list all supplier information

```
select *  
from suppliers;
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

snum	sname	status	city
S1	Michael Schumacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Eddy Merckx	1	Brussels
S13	Candice Swanepoel	3	Cape Town
S14	Adriana Lima	4	Sao Paulo
S15	Jennifer Lawrence	6	Oversboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barichello	3	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Messiauw	1	Eekloo
S2	Juan Pablo Montoya	4	Interlagos
S20	Daniilo Rossi	2	Milan
S21	Lizzie Armstrong	1	Hampstead
S22	Jan Ullrich	5	Bonn
S3	Dirk Thirau	1	Berlin
S32	Berni Schneider	2	Berlin
S33	Rolf Mitzig	3	Berlin
S39	Candice Swanepoel	18	Cardiff
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	9	London
S56	Marianne Vos	8	Zandvoort
S6	Nicola Giannini	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Raikkonen	2	Helsinki

## # Command 4: List the snum, sname, and pnum for those suppliers who ship the

# same part to every job. This is a fairly complex SQL nested query.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

```
select distinct suppliers.snum, suppliers.sname, shipments.pnum  
from suppliers natural join shipments  
where shipments.pnum in  
  (select pnum  
   from parts  
   where not exists  
     (select *  
      from jobs  
      where not exists  
        (select *  
         from shipments  
         where shipments.snum = suppliers.snum  
           and shipments.pnum = parts.pnum  
           and shipments.jnum = jobs.jnum  
        )  
     )  
  )
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

snum	sname	pnum
S1	Michael Schumacher	P8

# Command 5 is a multipart transaction that will cause the business logic to trigger

#

# The first part is a query to illustrate all shipment information before the update.

# The second part performs the update and causes the business logic to trigger.

# The third part is a query that illustrates all shipment information after the update/

# Command 5A: List all supplier information

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level user**.

Please enter any valid SQL query or update below:

```
select *
from suppliers;
```

[Execute Command](#) [Reset](#) [Clear Results](#)

All execution results will appear below this line:

Database Results:

enum	ename	status	city
S1	Michael Schumacher	1	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Edoardo Motta	1	Buenos
S13	Candice Swanepoel	3	Cape Town
S14	Adriana Lima	4	Sao Paulo
S15	Jennifer Lawrence	6	Owensboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barrichello	3	Sao Paulo
S18	Tim Bonten	2	Buenos
S19	Johan Massaro	1	Colonia
S2	Juan Pablo Montoya	4	Interlaken
S20	Dario Rosset	2	Milan
S21	Lizzie Armstrong	1	Hempstead
S22	Jan Ulrich	5	Bonn
S3	Dietrich Thurns	1	Berlin
S32	Bernd Schneider	2	Berlin
S33	Ralf Aldag	3	Berlin
S35	Candice Swanepoel	10	Cardiff
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	9	London
S6	Manfred Voss	6	Zandvoort
S8	Nicola Giannini	2	Milan
S7	Christian Albers	3	Ostende
S8	Giuseppe Fisichella	3	Milan
S9	Kimi Raikkonen	2	Helsinki

# Command 5B: List all shipment information



Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a root user.

Please enter any valid SQL query or update below:

Execute CommandResetClear Results

All execution results will appear below this line:

idnum	prnum	prnum	quantity
11	P1	25	100
11	P2	25	100
11	P3	25	100
11	P4	25	100
11	P5	25	100
11	P6	25	100
11	P7	25	100
11	P8	25	100
11	P9	25	100
11	P10	25	100
11	P11	25	100
11	P12	25	100
11	P13	25	100
11	P14	25	100
11	P15	25	100
11	P16	25	100
11	P17	25	100
11	P18	25	100
11	P19	25	100
11	P20	25	100
11	P21	25	100
11	P22	25	100
11	P23	25	100
11	P24	25	100
11	P25	25	100
11	P26	25	100
11	P27	25	100
11	P28	25	100
11	P29	25	100
11	P30	25	100
11	P31	25	100
11	P32	25	100
11	P33	25	100
11	P34	25	100
11	P35	25	100
11	P36	25	100
11	P37	25	100
11	P38	25	100
11	P39	25	100
11	P40	25	100
11	P41	25	100
11	P42	25	100
11	P43	25	100
11	P44	25	100
11	P45	25	100
11	P46	25	100
11	P47	25	100
11	P48	25	100
11	P49	25	100
11	P50	25	100
11	P51	25	100
11	P52	25	100
11	P53	25	100
11	P54	25	100
11	P55	25	100
11	P56	25	100
11	P57	25	100
11	P58	25	100
11	P59	25	100
11	P60	25	100
11	P61	25	100
11	P62	25	100
11	P63	25	100
11	P64	25	100
11	P65	25	100
11	P66	25	100
11	P67	25	100
11	P68	25	100
11	P69	25	100
11	P70	25	100
11	P71	25	100
11	P72	25	100
11	P73	25	100
11	P74	25	100
11	P75	25	100
11	P76	25	100
11	P77	25	100
11	P78	25	100
11	P79	25	100
11	P80	25	100
11	P81	25	100
11	P82	25	100
11	P83	25	100
11	P84	25	100
11	P85	25	100
11	P86	25	100
11	P87	25	100
11	P88	25	100
11	P89	25	100
11	P90	25	100
11	P91	25	100
11	P92	25	100
11	P93	25	100
11	P94	25	100
11	P95	25	100
11	P96	25	100
11	P97	25	100
11	P98	25	100
11	P99	25	100
11	P100	25	100

# Command 5C: Update the shipments table by increasing the quantity by 10 every shipment of part P3.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a root user.

Please enter any valid SQL query or update below:

update shipments  
set quantity = quantity + 10  
where prnum = 'P3';

Execute CommandResetClear Results

All execution results will appear below this line:

Database Results:

The statement executed successfully. 13 row(s) affected.

Business Logic Detected! - Updating Supplier Status

Business Logic update 6 supplier status marks.

# Command 5D: List all shipment information

Database Results:			
snum	pnum	pnum	quantity
S1	P1	.11	40
S1	P1	.14	200
S1	P19	.113	200
S1	P3	.113	100
S1	P33	.35	40
S1	P8	.11	80
S1	P8	.113	200
S1	P8	.25	100
S1	P8	.222	340
S1	P8	.33	200
S1	P8	.14	100
S1	P8	.35	220
S1	P8	.35	230
S12	P3	.222	780
S17	P3	.22	810
S2	P3	.11	80
S2	P3	.25	25
S2	P3	.222	80
S2	P3	.33	30
S2	P3	.14	80
S2	P3	.35	101
S2	P4	.11	50
S21	P2	.14	200
S22	P1	.33	120
S3	P19	.14	200
S3	P24	.25	200
S3	P3	.11	100
S3	P3	.35	85
S3	P4	.35	80
S3	P8	.35	100
S3	P8	.11	110
S39	P3	.11	30
S4	P8	.22	50
S44	P44	.22	100
S44	P43	.35	40
S5	P8	.35	400
S5	P1	.22	40
S6	P10	.33	100
S6	P11	.22	40
S6	P12	.14	75
S6	P13	.35	100
S6	P19	.35	80
S6	P2	.22	80
S6	P24	.25	20
S6	P3	.14	510
S6	P33	.14	40
S6	P4	.22	120
S6	P44	.14	20
S6	P43	.25	100
S6	P8	.222	200
S6	P83	.11	25
S6	P8	.222	120
S6	P7	.11	300
S6	P8	.33	110
S6	P8	.14	180
S7	P1	.22	40
S7	P12	.22	40
S7	P8	.22	40

# Command 5E: List all supplier information

<pre>select * from suppliers;</pre>			
Execute Command    Reset    Clear Results			
All execution results will appear below this line:			
Database Results:			
snum	sname	status	city
S1	Michael Schumacher	6	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Eddy Merckx	6	Brussels
S13	Candice Swanepoel	3	Cape Town
S14	Adriana Lima	4	Sao Paulo
S15	Jennifer Lawrence	6	Owensboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barrichello	8	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Messelew	1	Eckloo
S2	Juan Pablo Montoya	9	Interlagos
S20	Danilo Rossi	2	Milan
S21	Lizzie Armstrong	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	6	Berlin
S32	Bernd Schneider	2	Berlin
S33	Rolf Aldag	3	Berlin
S39	Candice Swanepoel	10	Cardiff
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	9	London
S56	Marianne Vos	8	Zandvoort
S6	Nicola Pizzoni	7	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Raikkonen	2	Helsinki

# Command 6: List the snum, and sname for those suppliers who ship only green parts.

# Output should list only supplier number S44 (Beryl Burton)

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a root-level user.

Please enter any valid SQL query or update below:

```
select sname
  from suppliers
    where snum in (select snum
                  from shipments
                    where pnnum in (select pnnum
                                   from parts
                                    where color = 'green'))
      and snum not in (select snum
                      from shipments
                        where pnnum in (select pnnum
                                       from parts
                                        where color <> 'green'))];
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

sname
Beryl Burton

# Command 7: This will generate an intentional error in command syntax.

# Output should show the error message generated by catching an SQLException

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a root-level user.

Please enter any valid SQL query or update below:

```
select alpha
from jobs; |
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

Database Error: java.sql.SQLException: Unknown column 'alpha' in 'field list'

# Command 8: This will not trigger the business logic.

# Output should show the message indicating that the command was executed but the business logic has not triggered

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

```
update jobs
set city = "Madrid"
where city = "Paris"; |
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

The statement executed successfully. 1 row(s) affected.

Business Logic Not Triggered

# Command 9: Verify that command #8 did in fact update the jobs table

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

```
select *
from jobs|
```

Execute Command

Reset

Clear Results

All execution results will appear below this line:

Database Results:

jnum	jname	numworkers	city
J1	Operation DB	45	Berlin
J13	Night Strike	350	Madrid
J2	Really Big Job	500	Melbourne
J22	Project On-Time	200	London
J3	Small Job	100	Chicago
J4	New Job	50	Berlin
J5	My Job	1	Orlando
J6	A New Job	14	Milan

# Command 10: This is a basic select command using an aggregation operator

# Business logic not triggered and no table is updated.

Welcome to 2023 Project 4 Enterprise Database System

A Servlet/JSP-based Multi-tiered Enterprise Application Using a Tomcat Container

You are connected to the Project 4 Enterprise System as a **root-level** user.

Please enter any valid SQL query or update below:

```
select count(*) as total_shipments
from shipments]
```

Execute Command Reset Clear Results

All execution results will appear below this line:

Database Results:

SQL Command not recognized. Please make sure you use a DDL, DML, or DCL command.

## # Data Entry LEVEL USER command script

1. # Command 1: Insert new shipment record for supplier S5.

# This will work ok. Business logic not triggered

# insert into shipments values ("S39", "P6", "J4", 33);

**Welcome to DataEntryHome**

**Supplier**  

snum sname status city Enter Record Clear

0 rows updated.

**Parts**  

snum sname color weight city Enter Record Clear

0 rows updated.

**Jobs**  

jnum jname numworkers city Enter Record Clear

0 rows updated.

**Shipments**  

S39 P6 J4 33 Enter Record Clear

0 rows updated.

## RESULT

**Shipments**  

snum pnum jnum quantity Enter Record Clear

insert successful  
1 rows updated.

# Command 2: This command will not execute due to a referential integrity violation.

# insert into shipments values ("S39", "P55", "J4", 150);

**Shipments**  

S39 P55 J4 150 Enter Record Clear

0 rows updated.

## Result

**Shipments**  

snum pnum jnum quantity Enter Record Clear

Error executing the SQL statement:  
Cannot add or update a child row: a foreign key constraint fails (`project4`.`shipments`, CONSTRAINT `shipments\_ibfk\_2` FOREIGN KEY (`pnum`) REFERENCES `parts` (`pnum`))  
0 rows updated.

# Command 3: This command will work.

```
# insert into suppliers values("S90","Anna-Frieda",34,"Stockholm");
```

---

#### Supplier

S90	Anna-Frieda	34	Stockholm	Enter Record	Clear
-----	-------------	----	-----------	--------------	-------

0 rows updated.

---

result

---

#### Supplier

snum	sname	status	city	Enter Record	Clear
------	-------	--------	------	--------------	-------

1 rows updated.

---

# Command 4: This command will work.

```
# insert into jobs values ("J90","Top Secret Job",3,"Stockholm");
```

---

#### Jobs

J90	Top Secret Job	3	Stockholm	Enter Record	Clear
-----	----------------	---	-----------	--------------	-------

0 rows updated.

---

result

---

#### Jobs

jnum	jname	numworkers	city	Enter Record	Clear
------	-------	------------	------	--------------	-------

1 rows updated.

---

# Command 5: This command will work and trigger the business logic.

```
# insert into shipments values("S39","P6","J6",500);
```

---

#### Shipments

S39	P6	J6	500	Enter Record	Clear
-----	----	----	-----	--------------	-------

0 rows updated.

---

result

---

#### Shipments

snum	pnum	jnum	quantity	Enter Record	Clear
------	------	------	----------	--------------	-------

updated 1 supplier(s)

1 rows updated.

---

# Command 6: This command will work and not trigger the business logic.

```
# insert into parts values("P90","U-plate","black",50,"Cardiff");
```

#### Parts

P90	U-plate	black	50	Cardiff	Enter Record	Clear
-----	---------	-------	----	---------	--------------	-------

0 rows updated.

result

#### Parts

snum	sname	color	weight	city	Enter Record	Clear
------	-------	-------	--------	------	--------------	-------

1 rows updated.

# Command 7: This command will not work.

```
# insert into shipments values("S39","P6","J6",35);
```

#### Shipments

S39	P6	J6	35	Enter Record	Clear
-----	----	----	----	--------------	-------

0 rows updated.

result

#### Shipments

snum	pnum	jnum	quantity	Enter Record	Clear
------	------	------	----------	--------------	-------

Error executing the SQL statement:  
Duplicate entry 'S39-P6-J6' for key 'shipments.PRIMARY'

0 rows updated.

# Command 8: This command will work and trigger the business logic.

```
# insert into shipments values("S90","P90","J90",500);
```

#### Shipments

S90	P90	J90	500	Enter Record	Clear
-----	-----	-----	-----	--------------	-------

0 rows updated.

Result:

#### Shipments

snum	pnum	jnum	quantity	Enter Record	Clear
------	------	------	----------	--------------	-------

updated 1 supplier(s)

1 rows updated.



## # Account LEVEL USER command script

### Command 1

**Accountant User**

- ☒ Get The Maximum Status Value Of All Supplier
- ☐ Get The Total Weight Of All Parts
- ☐ Get The Total Number Of Shipments
- ☐ Get The Name And Number Of Workers Of The Job With The Most Workers
- ☐ List The Name And Status Of Every Supplier

**Execute command** **Clear result**

**Execution Result**

MAX(status)  
39

### Command 2

**Accountant User**

- ☐ Get The Maximum Status Value Of All Supplier
- ☒ Get The Total Weight Of All Parts
- ☐ Get The Total Number Of Shipments
- ☐ Get The Name And Number Of Workers Of The Job With The Most Workers
- ☐ List The Name And Status Of Every Supplier

**Execute command** **Clear result**

**Execution Result**

SUM(weight)  
143

### Command 3

**Accountant User**

- ☐ Get The Maximum Status Value Of All Supplier
- ☐ Get The Total Weight Of All Parts
- ☒ Get The Total Number Of Shipments
- ☐ Get The Name And Number Of Workers Of The Job With The Most Workers
- ☐ List The Name And Status Of Every Supplier

Execute commandClear result

**Execution Result**

```
COUNT(*)
61
```

### Command 4

**Accountant User**

- ☐ Get The Maximum Status Value Of All Supplier
- ☐ Get The Total Weight Of All Parts
- ☐ Get The Total Number Of Shipments
- ☒ Get The Name And Number Of Workers Of The Job With The Most Workers
- ☐ List The Name And Status Of Every Supplier

Execute commandClear result

**Execution Result**

jname	numworkers
Really Big Job	500

## Command 5

### Accountant User

- ☐ Get The Maximum Status Value Of All Supplier
- ☐ Get The Total Weight Of All Parts
- ☐ Get The Total Number Of Shipments
- ☐ Get The Name And Number Of Workers Of The Job With The Most Workers
- ☒ List The Name And Status Of Every Supplier

Execute command

Clear result

### Execution Result

sname	status
Michael Schumacher	6
David Coulthard	2
Bernard Hinault	7
Eddy Merckx	6
Candice Swanepoel	3
Adriana Lima	4
Jennifer Lawrence	6
Fernando Alonso	4
Rubens Barichello	8
Tom Boonen	2
Johan Messeuw	1
Juan Pablo Montoya	9
Danilo Rossi	2
Lizzie Armistead	1
Jan Ullrich	5
Dietrich Thurau	6
Bernd Schnieder	2
Rolf Aldag	3
Candice Swanepoel	15
Mark Webber	5
Beryl Burton	4
Jenson Button	9
Marianne Vos	8
Nicola Gianniberti	7
Christian Albers	3
Giancarlo Fisichella	3
Kimi Rikonnen	2
Anna-Frieda	39



**# DATA ENTRY LEVEL USER** command script