CLIENT LEVEL USER command script
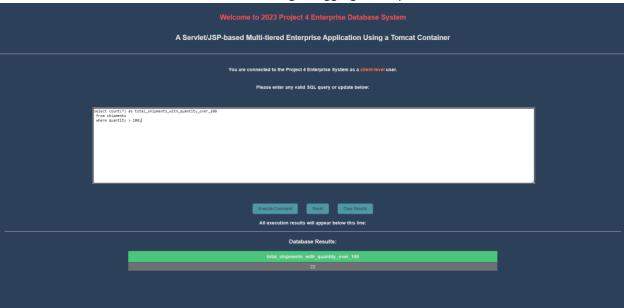
1. Command 1: Query: list the supplier number and supplier name for those suppliers who ship every green part.

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.

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.

2. # Command 1:
Command 2A: Query: list all supplier information.

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', 'J6', 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:

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	Owensboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barichello	3	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Messeuw	1	Eekloo
S2	Juan Pablo Montoya	4	Interlagos
S20	Danilo Rossi	2	Milan
S21	Lizzie Armistead	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	1	Berlin
S32	Bernd Schneider	2	Berlin
S33	Rolf Aldag	3	Berlin
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	9	London
S56	Marianne Vos	8	Zandvoort
S6	Nicola Gianniberti	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	2	Helsinki

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

Command 3A list all supplier information

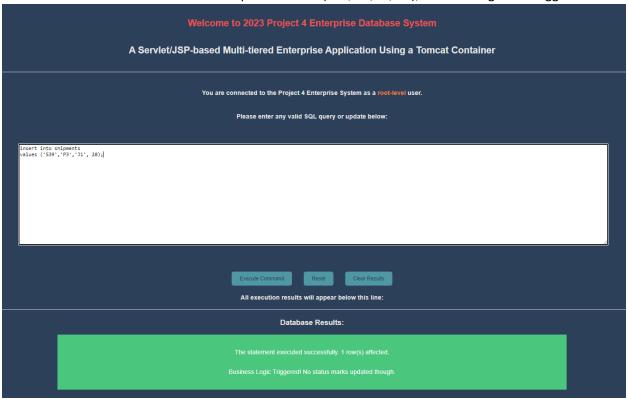**Welcome to 2023 Project 4 Enterprise Database System**

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:

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	Owensboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barichello	3	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Messeuw	1	Eekloo
S2	Juan Pablo Montoya	4	Interlagos
S20	Danilo Rossi	2	Milan
S21	Lizzie Armistead	1	Hempstead
S22	Jan Ullrich	6	Bonn
S3	Dietrich Thurau	1	Berlin
S32	Bernd Schneider	2	Berlin
S33	Rolf Aldag	3	Berlin
S4	Mark Webber	5	Melbourne
S44	Beryl Burton	4	London
S5	Jenson Button	9	London
S56	Marianne Vos	8	Zandvoort
S6	Nicola Gianniberti	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	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');
```

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.

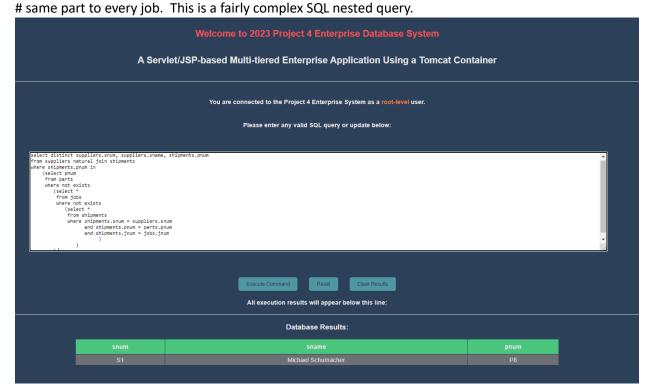# Command 3C: list all supplier information.

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:

```
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	Owensboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barichello	3	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Messeuw	1	Eekloo
S2	Juan Pablo Montoya	4	Interlagos
S20	Danilo Rossi	2	Milan
S21	Lizzie Armistead	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	1	Berlin
S32	Bernd Schnieder	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 Gianniberti	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	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	Owensboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barichello	3	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Messeuw	1	Eekloo
S2	Juan Pablo Montoya	4	Interlagos
S20	Danilo Rossi	2	Milan
S21	Lizzie Armistead	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	1	Berlin
S32	Bernd Schnieder	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 Gianniberti	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	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/
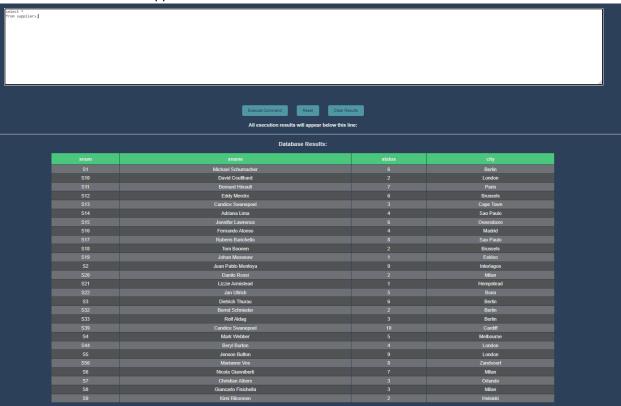
Command 5A: List all supplier information

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 Hinault	7	Paris
S12	Eddy 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 Barichello	3	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Messeuse	1	Eekloo
S2	Juan Pablo Montoya	4	Interlagos
S20	Danilo Rossi	2	Milan
S21	Lizzie Armistead	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thursu	1	Berlin
S32	Bernd Schnieder	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 Gianniberti	2	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	2	Helsinki

Command 5B: List all shipment information

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-level user.

Please enter any valid SQL query or update below:

```
update shipments
set quantity = quantity + 10
where pnum = 'P3';
```

Execute Command   Reset   Clear 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	jnum	quantity
S1	P1	J1	40
S1	P1	J4	200
S1	P19	J13	200
S1	P3	J13	105
S1	P33	J5	40
S1	P6	J1	80
S1	P6	J13	200
S1	P6	J2	100
S1	P6	J22	340
S1	P6	J3	200
S1	P6	J4	100
S1	P6	J5	225
S1	P6	J6	230
S12	P3	J22	760
S17	P3	J2	810
S2	P3	J1	50
S2	P3	J2	25
S2	P3	J22	60
S2	P3	J3	30
S2	P3	J4	60
S2	P3	J5	101
S2	P4	J1	50
S21	P2	J4	200
S22	P1	J3	120
S3	P10	J4	200
S3	P24	J2	200
S3	P3	J1	109
S3	P3	J6	85
S3	P4	J6	80
S3	P6	J6	120
S3	P9	J1	110
S39	P3	J1	30
S4	P6	J2	50
S44	P44	J2	100
S44	P48	J5	40
S5	P6	J6	400
S8	P1	J2	40
S8	P10	J3	100
S8	P11	J2	40
S8	P12	J4	75
S8	P13	J5	100
S8	P19	J6	50
S8	P2	J2	60
S8	P24	J2	20
S8	P3	J4	510
S8	P33	J4	40
S8	P4	J2	120
S8	P44	J4	20
S8	P48	J2	100
S8	P5	J22	200
S8	P63	J1	25
S8	P6	J22	120
S8	P7	J1	300
S8	P8	J3	110
S8	P9	J4	180
S7	P1	J2	40
S7	P12	J2	40
S7	P6	J2	40

Command 5E: List all supplier information

```
select *
from suppliers;
```

Execute Command   Reset   Clear Results

All execution results will appear below this line:

Database Results:

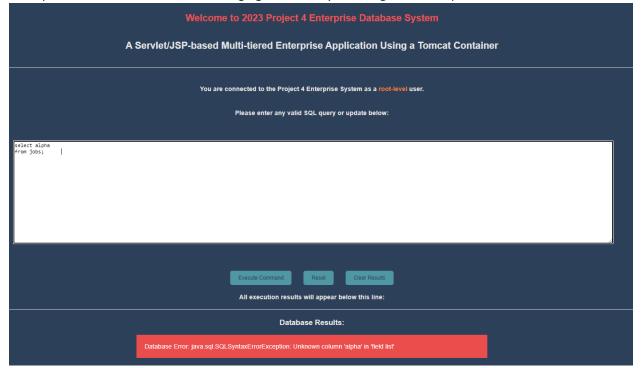| snum	sname	status	city
S1	Michael Schumacher	6	Berlin
S10	David Coulthard	2	London
S11	Bernard Hinault	7	Paris
S12	Eddy Merckx	6	Brussels
S13	Candice Swanepoel	3	Cape Town
S14	Adriana Lima	4	Sao Paulo
S15	Jennifer Lawrence	6	Owensboro
S16	Fernando Alonso	4	Madrid
S17	Rubens Barichello	8	Sao Paulo
S18	Tom Boonen	2	Brussels
S19	Johan Messeuw	1	Eekloo
S2	Juan Pablo Montoya	9	Interlagos
S20	Danilo Rossi	2	Milan
S21	Lizzie Armistead	1	Hempstead
S22	Jan Ullrich	5	Bonn
S3	Dietrich Thurau	6	Berlin
S32	Bernd Schnieder	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 Gianniberti	7	Milan
S7	Christian Albers	3	Orlando
S8	Giancarlo Fisichella	3	Milan
S9	Kimi Rikonnen	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)

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 pnum in (select pnum
                            from parts
                                where color = 'green'))
                        and
                    snum not in (select snum
                            from shipments
                                where pnum in (select pnum
                                        from parts
                                            where color <> 'green') );
```

Execute Command   Reset   Clear Results

All execution results will appear below this line:

Database Results:

sname
Beryl Burton

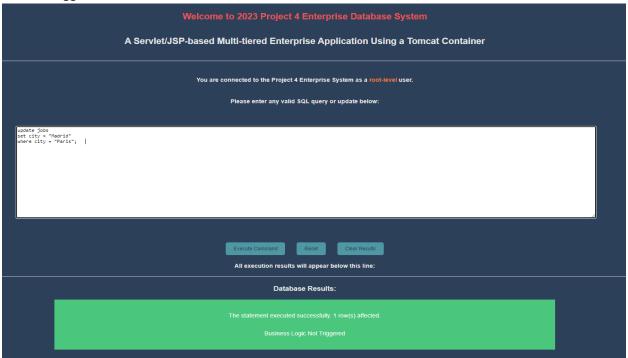# Command 7: This will generate an intentional error in command syntax.

Output should show the error message generated by catching an SQLEception

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.SQLSyntaxErrorException: Unknown column 'alpha' in 'field list'
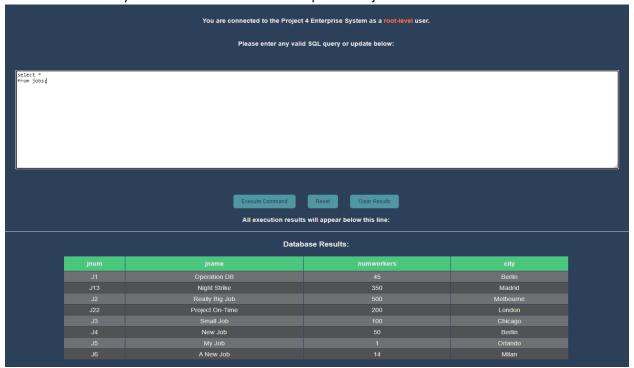
Command 8: This will not trigger the business logic.

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:

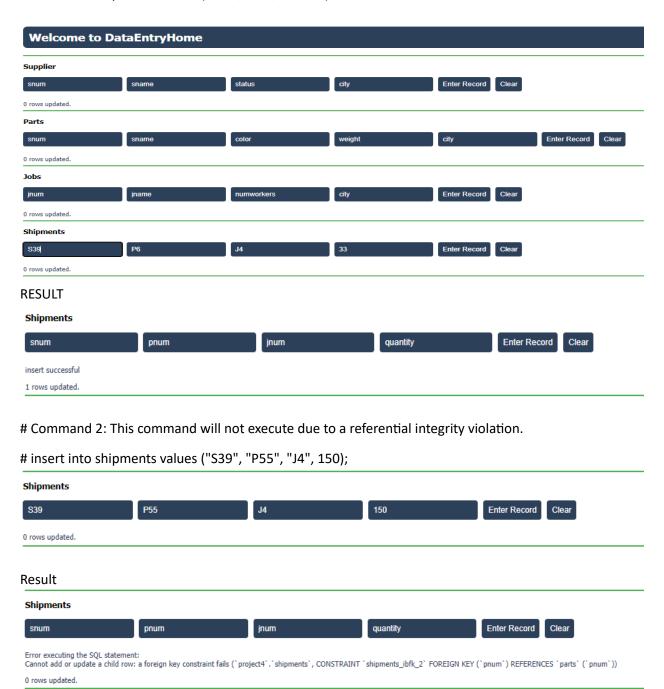| jnum	jname	numworkers	city
J1	Operation DB	45	Berlin
J13	Night Strike	350	Madrid
J2	Really Big Job	500	Melbourne
J22	Project On-Time	200	London
J3	Small Job	100	Chicago
J4	New Job	50	Berlin
J5	My Job	1	Orlando
J6	A New Job	14	Milan

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:

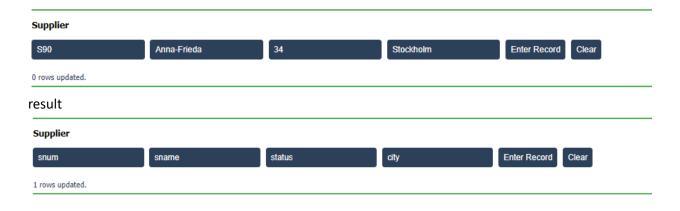SQL Command not recognized. Please make sure you use a DDL, DML, or DCL command.

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

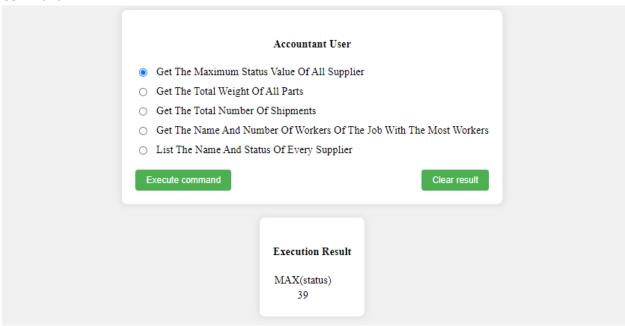| S39 | P6 | J6 | 500 | Enter Record | Clear |

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

Command 4

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