

DBMS PROJECT REPORT

ON

Hostel Food Wastage Management System



THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

by

Sehaj Verma (101703497)

Sehaj Pal Singh (101703498)

Saloni Malhotra (101703473)

Submitted To

Dr. Manisha Kaushal

Acknowledgements

We would like to express our deepest appreciation to Dr. Manisha Kaushal without whom every DBMS class would not be something to look forward to. She has always been there as a mentor and pillar of support throughout the making of the project without her guidance and persistent help, this dissertation would not have been possible. We are highly indebted for her guidance and constant supervision as well as for providing necessary information regarding the report.

In view of that, we worked as a team; each playing instrumental role with utmost enthusiasm and applied our knowledge and understanding of the different factors and elements related to schooling and education.

We also wish to express our gratitude to the officials and other staff members of Thapar Institute of Engineering and Technology who extended their help during the period of our project.

INDEX

Serial Number	Topic	Page Number
1	Acknowledgement	2
2	Requirement Analysis	4
3	ER Diagram and Tables	5
4	Table Names and Attributes	6
5	PL/SQL Codes	8
6	Snapshots	13
7	References	21

REQUIREMENT ANALYSIS

Background of the area

Hostel Messes in various institutions serve food to the students. Though the food is made according to an estimation of the previous day. Most of the time food is not finished and the leftover food gets wasted. This food wastage cannot be used for the next day as the processed food starts to perish. Food wastage is massive as every institution have an average of 5 hostels.

Need of the Project

The Hostel Wastage Management project is required to minimize the food wastage and make everybody realise the wastage of food they do every day.

Objectives of the project

This project aims to minimize the food wastage by displaying the leftover food at the online portal for the needy to see it and collect the food within a certain time period.

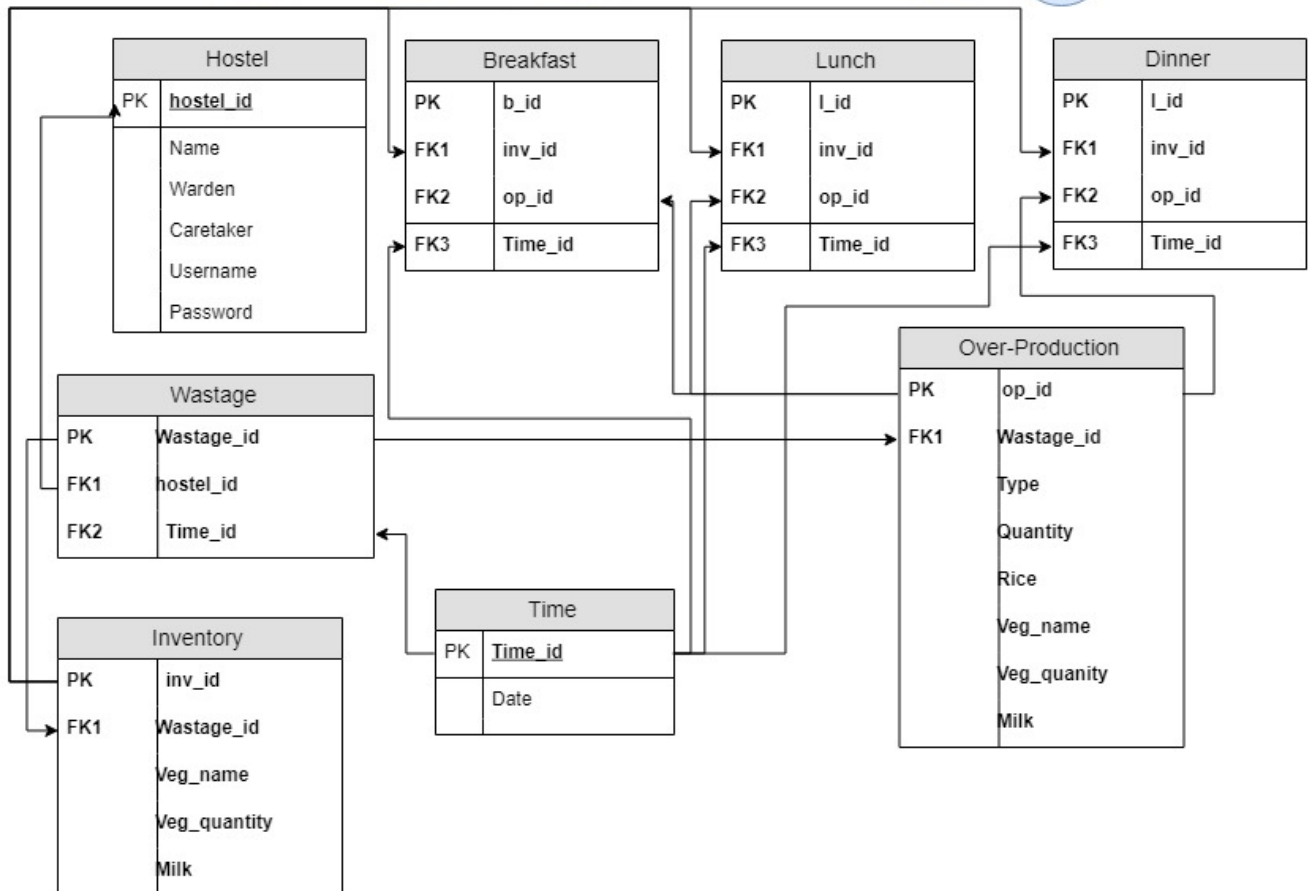
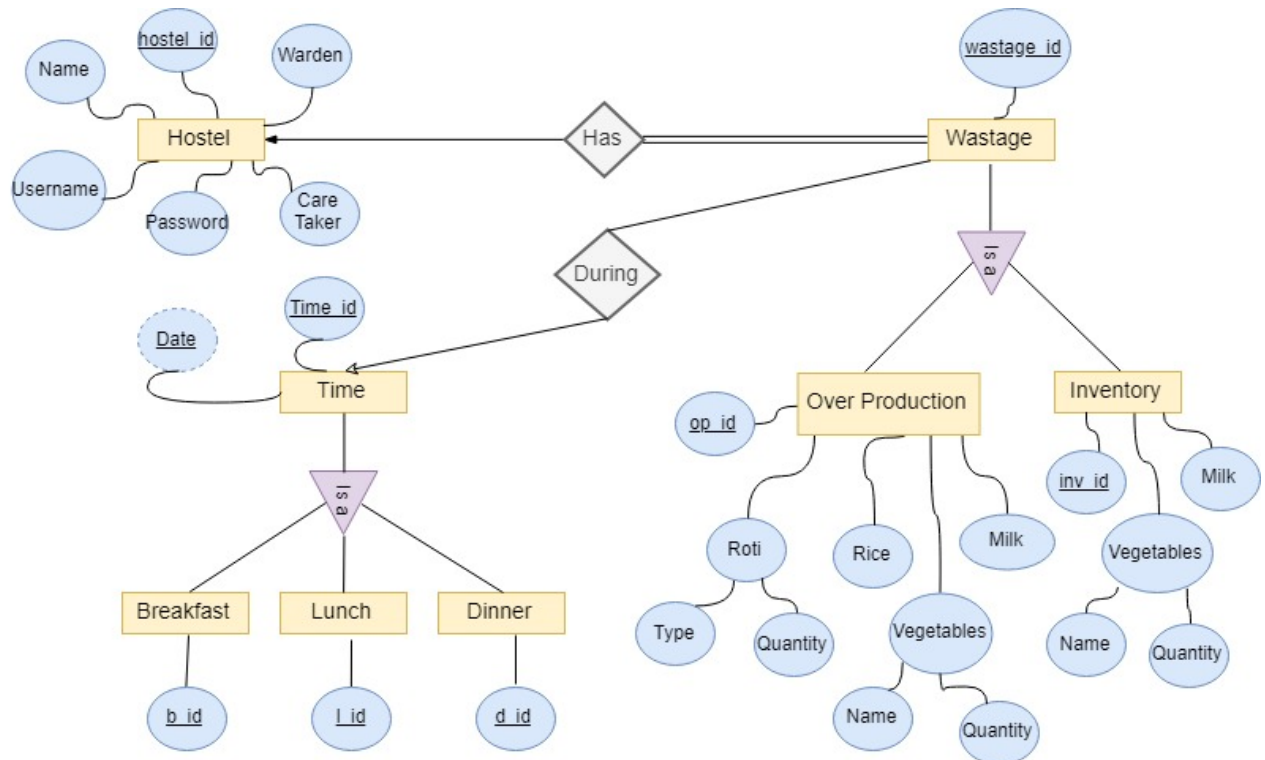
Project Outcomes

This project will minimize the food wastage to some extent and give an idea about the scarcity of food.

About the features/functionalities

In order to create this particular database management system, the software used was sqlplus which offers great inbuilt features such as cursors, functions , procedures, indexes ,sequences and other various inbuilt features which are used to insert, update, delete and view data which helped us to the create the database management system in a very efficient manner.

ER DIAGRAM AND TABLES



TABLES AND ATTRIBUTES

TABLE NAME	ATTRIBUTES
HOSTEL	Hostel_id(Primary key), Name, Warden, Caretaker, Username, Password
BREAKFAST	B_id(Primary Key), inv_id (FK1), op_id(FK2), Time_id(FK3)
LUNCH	B_id(Primary Key), inv_id (FK1), op_id(FK2), Time_id(FK3)
DINNER	B_id(Primary Key), inv_id (FK1), op_id(FK2), Time_id(FK3)
WASTAGE	Wastage_id (Primary Key) Hostel_id(FK) Time_id(FK2)
INVNTORY	Inv_id, wastage_id, ,Veg_name, ,Veg_quantity, Milk
TIME	Time_id(Primary Key), Date
OVER-PRODUCTION	Op_id (Primary Key), Wastge_id(FK1), Type, Quantity, Rice, Veg_name, Veg_quantity, Milk

NORMALIZATION

The tables created are already in the normalized form.

PL/SQL CODE

TABLES

```
create table Hostel(hostel_id number(10),Name varchar(20),Warden varchar(20),Caretaker  
varchar(20),primary key(hostel_id));
```

```
create table Breakfast(b_id number(10), inv_id number(10), op_id number(10), Time_id  
number(10),primary key(b_id), foreign key(inv_id) references Inventory, foreign key(op_id)  
references OverProduction, foreign key(Time_id) references Time);
```

```
create table Lunch(l_id number(10), inv_id number(10), op_id number(10), Time_id  
number(10),primary key(l_id), foreign key(inv_id) references Inventory, foreign key(op_id)  
references OverProduction, foreign key(Time_id) references Time);
```

```
create table Dinner(d_id number(10), inv_id number(10), op_id number(10), Time_id  
number(10),primary key(d_id), foreign key(inv_id) references Inventory, foreign key(op_id)  
references OverProduction, foreign key(Time_id) references Time);
```

```
create table Wastage(Wastage_id number(10), hostel_id number(10), Time_id  
number(10),primary key(Wastage_id), foreign key(hostel_id) references Hostel, foreign  
key(Time_id) references Time);
```

```
create table Time(Time_id number(10), date Date, primary key(Time_id));
```

```
create table Inventory(inv_id number(10), Wastage_id number(10), Veg_name  
varchar(10),Veg_quantity number(10), milk number(10), primary key(inv_id), foreign  
key(Wastage_id) references Wastage);
```

```
create table OverProduction(op_id number(10), Wastage_id number(10),Type varchar(10),  
Quantity number(10), Rice number(10), Veg_name varchar(10),Veg_quantity number(10),  
milk number(10), primary key(op_id), foreign key(Wastage_id) references Wastage);
```

PL/SQL CODES

INSERT

Procedures and Functions

create or replace procedure deleteHostel()

```
begin

    dbms_output.put_line(" Enter the Hostel ID to be deleted");

    p1:=&p1;

    delete from Hostel where hostel_id=p1;

end;
```

create or replace procedure deleteInv()

```
begin

    dbms_output.put_line(' Enter the Inventory ID to be deleted');

    p2:=&p2;

    delete from Inventory where inv_id=p2;

end;
```

create or replace procedure deleteOp()

```
begin

    dbms_output.put_line(" Enter the OverProduction ID to be deleted");

    p3:=&p3;

    delete from Inventory where op_id=p3;

end;
```


create or replace procedure displayOp()

declare

opid OverProduction.op_id%type;

wastageid OverProduction.Wastage_id%type;

typeo OverProduction.Type%type;

Quantity OverProduction.Quantity%type;

Rice OverProduction.Rice%type;

Veg_name2 Inventory.Veg_name%type;

Veg_quantity2 Inventory.Veg_quantity%type;

Milk2 Inventory.Milk%type;

CURSOR c_op is

SELECT op_id, Wastage_id,Type, Quantity,Rice,Veg_name,Veg_quantity,Milk FROM
OverProduction;

begin

OPEN c_op;

LOOP

FETCH c_op into opid, wastageid, typeo,Quantity;

EXIT WHEN c_op%notfound;

dbms_output.put_line(opid || ' ' || wastageid || ' ' || typeo || ' ' ||
Quantity|| ' ' || Rice|| ' ' || Veg_name2|| ' ' || Veg_quantity2|| ' ' || Milk2);

END LOOP;

CLOSE c_op;

end;

create or replace function displayInv()

begin

OPEN c_inv;

LOOP

FETCH c_inv into invid, wastageid, Veg_name,Veg_quantity,Milk;

EXIT WHEN c_inv%notfound;

dbms_output.put_line(wastageid || ' ' || wastageid || ' ' || Veg_name || ' ' ||
Veg_quantity|| ' ' || Milk);

END LOOP;

CLOSE c_inv;

end;

create function displayHostel()

DECLARE

begin

OPEN h_hostel;

LOOP

FETCH h_hostel into hostelid, name, warden,caretaker;

EXIT WHEN h_hostel%notfound;

dbms_output.put_line(hostelid || ' ' || name || ' ' || warden || ' ' || caretaker);

END LOOP;

CLOSE h_hostel;

end;

SHOW (Cursor)

CURSOR h_hostel is

SELECT hostel_id, name, warden, caretaker FROM hostel;

CURSOR c_inv is

SELECT inv_id, Wastage_id, Veg_name, Veg_quantity, Milk FROM Inventory;

CURSOR c_op is

SELECT op_id, Wastage_id, Type, Quantity, Rice, Veg_name, Veg_quantity, Milk FROM OverProduction;

TRIGGER

create or replace trigger caretaker_warden

before insert or update of Warden, Caretaker on Hostel

for each row

begin

if :NEW.Warden=:NEW.Caretaker then

RAISE_APPLICATION_ERROR(-20004, 'A person cannot be caretaker and warden at same time');

end if; end;

/

create or replace trigger timetrigger

before insert on Wastage

begin

insert into Time

(Time_id, cdate) values(time_seq.nextval, sysdate);

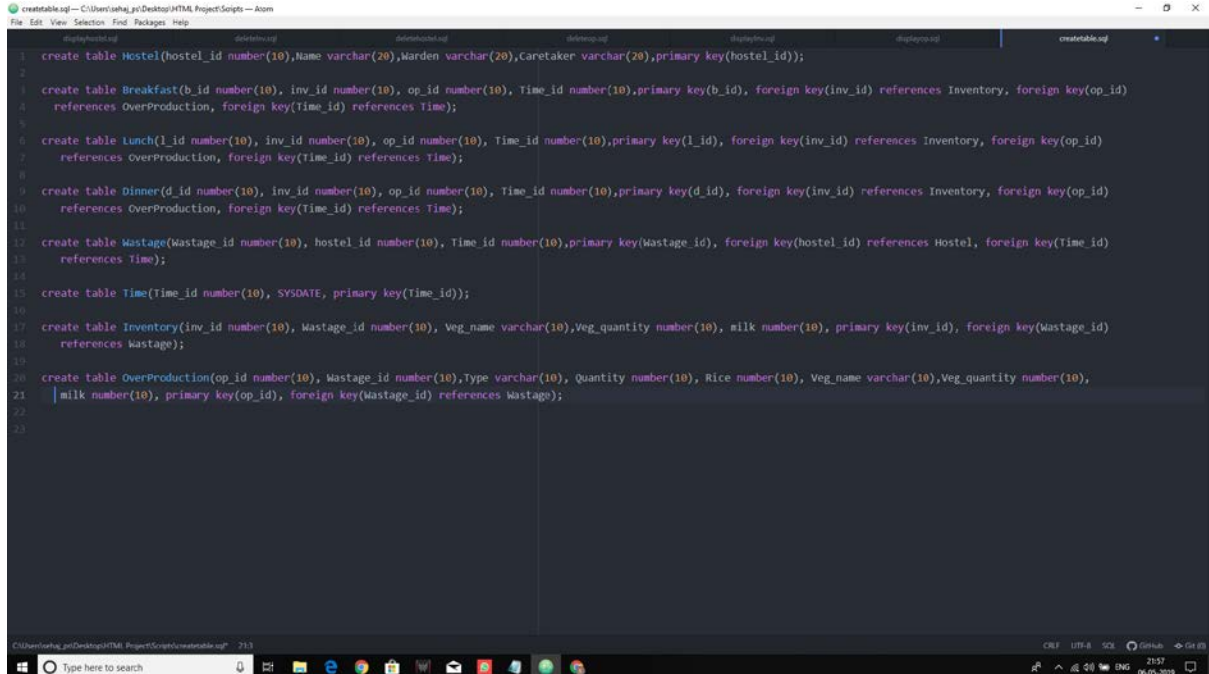
```
end;  
  
/  
  
CREATE OR REPLACE TRIGGER FOREIGN_KEY  
BEFORE INSERT OR UPDATE OF hostel_id ON WASTAGE  
FOR EACH ROW  
  
DECLARE DNO WASTAGE.hostel_id%TYPE;  
  
BEGIN  
  
SELECT hostel_id INTO DNO FROM Hostel WHERE hostel_id=:NEW.hostel_id;  
  
NULL;  
  
EXCEPTION  
  
WHEN NO_DATA_FOUND  
  
THEN RAISE_APPLICATION_ERROR(-20004, 'FOREIGN KEY VIOLATED BECAUSE  
VALUE IS NOT FOUND IN THE PARENT TABLE');  
  
END;  
  
/
```

SEQUENCE

```
create sequence time_seq  
  
start with 1  
  
maxvalue 100  
  
minvalue 1  
  
cycle  
  
cache 20  
  
/
```

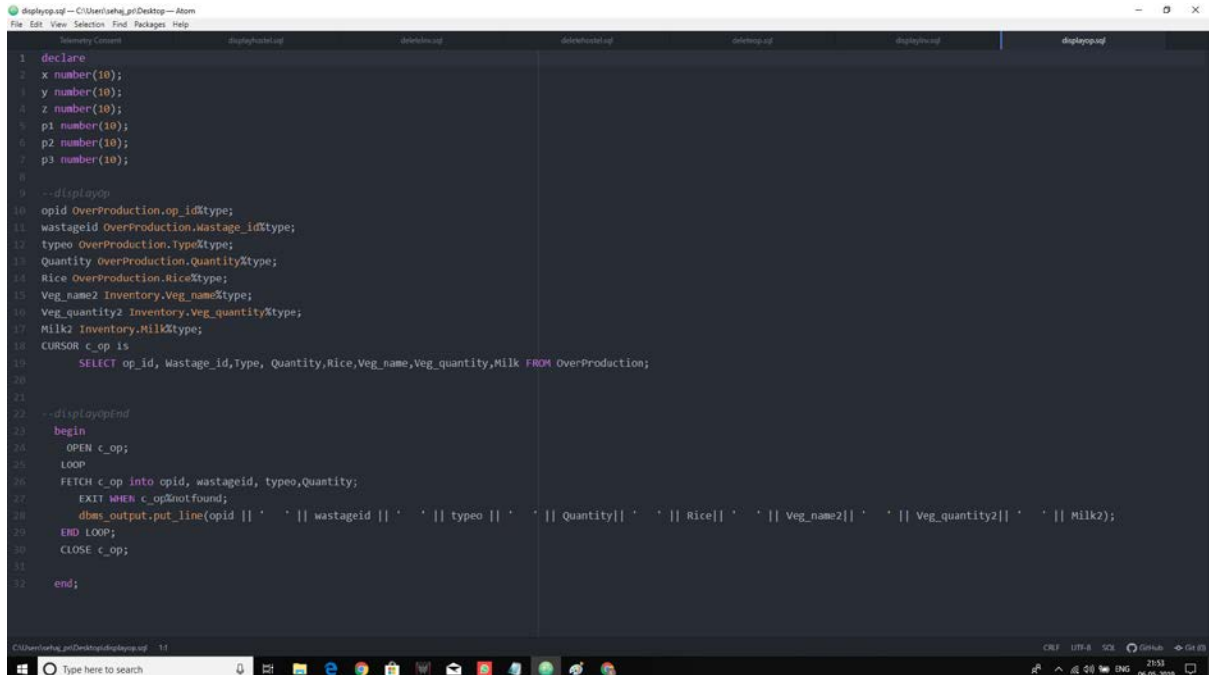
Snapshots

Create Table



```
1 create table Hostel(hostel_id number(10),Name varchar(20),Warden varchar(20),primary key(hostel_id));
2
3 create table Breakfast(b_id number(10), inv_id number(10), op_id number(10), Time_id number(10),primary key(b_id), foreign key(inv_id) references Inventory, foreign key(op_id)
4 references OverProduction, foreign key(Time_id) references Time);
5
6 create table Lunch(l_id number(10), inv_id number(10), op_id number(10), Time_id number(10),primary key(l_id), foreign key(inv_id) references Inventory, foreign key(op_id)
7 references OverProduction, foreign key(Time_id) references Time);
8
9 create table Dinner(d_id number(10), inv_id number(10), op_id number(10), Time_id number(10),primary key(d_id), foreign key(inv_id) references Inventory, foreign key(op_id)
10 references OverProduction, foreign key(Time_id) references Time);
11
12 create table Wastage(Wastage_id number(10), hostel_id number(10), Time_id number(10),primary key(Wastage_id), foreign key(hostel_id) references Hostel, foreign key(Time_id)
13 references Time);
14
15 create table Time(Time_id number(10), SYSDATE, primary key(Time_id));
16
17 create table Inventory(inv_id number(10), Wastage_id number(10), Veg_name varchar(10),Veg_quantity number(10), milk number(10), primary key(inv_id), foreign key(Wastage_id)
18 references Wastage);
19
20 create table OverProduction(op_id number(10), Wastage_id number(10),Type varchar(10), Quantity number(10), Rice number(10), Veg_name varchar(10),Veg_quantity number(10),
21 |milk number(10), primary key(op_id), foreign key(Wastage_id) references Wastage);
22
23
```

Display



```
1 declare
2 x number(10);
3 y number(10);
4 z number(10);
5 p1 number(10);
6 p2 number(10);
7 p3 number(10);
8
9 --display
10 opid OverProduction.op_id%type;
11 wastageid OverProduction.Wastage_id%type;
12 typeeo OverProduction.Type%type;
13 Quantity OverProduction.Quantity%type;
14 Rice OverProduction.Rice%type;
15 Veg_name2 Inventory.Veg_name%type;
16 Veg_quantity2 Inventory.Veg_quantity%type;
17 Milk2 Inventory.Milk%type;
18 CURSOR c_op is
19     SELECT op_id, Wastage_id,Type, Quantity,Rice,Veg_name,Veg_quantity,Milk FROM OverProduction;
20
21 --displayend
22 begin
23     OPEN c_op;
24     LOOP
25         FETCH c_op into opid, wastageid, typeeo,Quantity;
26         EXIT WHEN c_op%notfound;
27         dbms_output.put_line(opid || ' ' || wastageid || ' ' || typeeo || ' ' || Quantity || ' ' || Rice || ' ' || Veg_name2 || ' ' || Veg_quantity2 || ' ' || Milk2);
28     END LOOP;
29     CLOSE c_op;
30
31 end;
```

```
displayinv.sql -- C:\Users\seha\Desktop -- Atom
File Edit View Selection Find Packages Help
displayinv.sql displayhostel.sql displayhostel.sql displayhostel.sql displayhostel.sql displayhostel.sql

1 declare
2 x number(10);
3 y number(10);
4 z number(10);
5 p1 number(10);
6 p2 number(10);
7 p3 number(10);
8
9 --displayinv
10 invid Inventory.inv_id%type;
11 wastageid Inventory.wastage_id%type;
12 Veg_name Inventory.Veg_name%type;
13 Veg_quantity Inventory.Veg_quantity%type;
14 Milk Inventory.Milk%type;
15 CURSOR c_inv is
16     SELECT inv_id, Wastage_id, Veg_name,Veg_quantity,Milk FROM Inventory;
17 --displayinvend
18
19 begin
20     OPEN c_inv;
21     LOOP
22         FETCH c_inv into invid, wastageid, Veg_name,Veg_quantity,Milk;
23         EXIT WHEN c_inv%notfound;
24         dbms_output.put_line(wastageid || ' ' || wastageid || ' ' || Veg_name || ' ' || Veg_quantity || ' ' || Milk);
25     END LOOP;
26     CLOSE c_inv;
27 end;
28

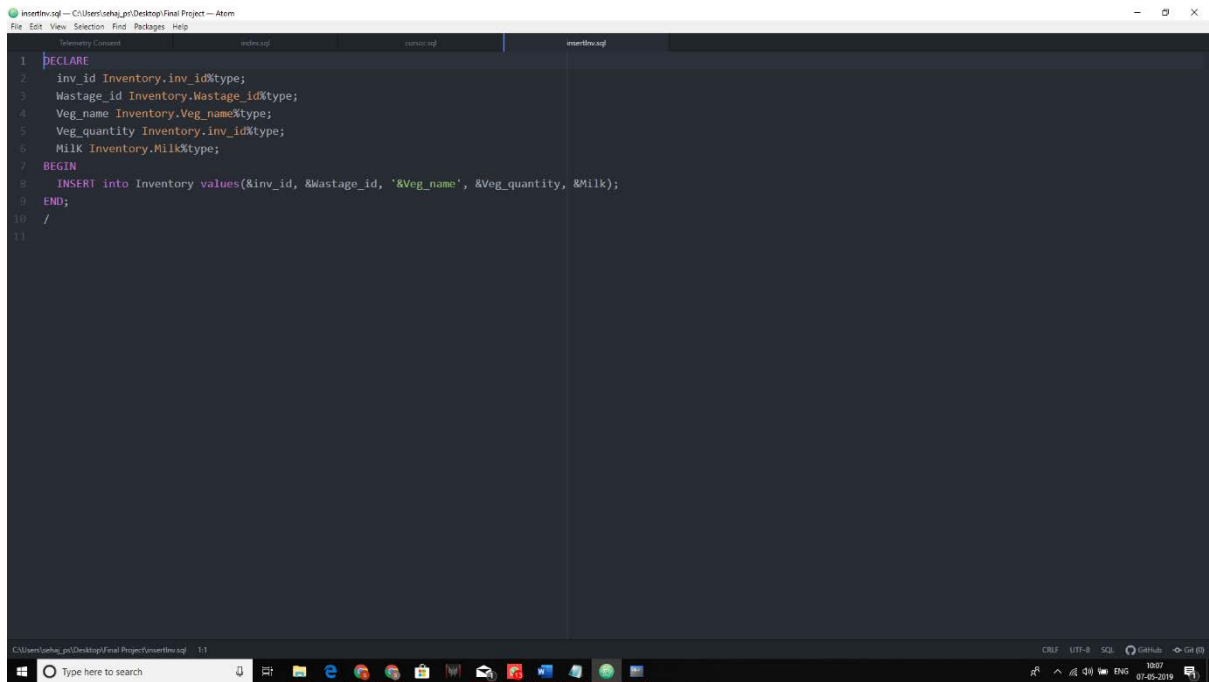
C:\Users\seha\Desktop\displayinv.sql 11
CTRL LTRF SQL GitHub GDB
Type here to search 7:51 06-05-2019
```

```
displayhostel.sql -- C:\Users\seha\Desktop -- Atom
File Edit View Selection Find Packages Help
displayhostel.sql

1 declare
2 x number(10);
3 y number(10);
4 z number(10);
5 p1 number(10);
6 p2 number(10);
7 p3 number(10);
8
9 --displayhostel
10 hostelid hostel.hostel_id%type;
11 name hostel.name%type;
12 warden hostel.warden%type;
13 caretaker hostel.caretaker%type;
14 CURSOR h_hostel is
15     SELECT hostel_id, name, warden,caretaker FROM hostel;
16 --displayhostelend
17
18 begin
19     OPEN h_hostel;
20     LOOP
21         FETCH h_hostel into hostelid, name, warden,caretaker;
22         EXIT WHEN h_hostel%notfound;
23         dbms_output.put_line(hostelid || ' ' || name || ' ' || warden || ' ' || caretaker);
24     END LOOP;
25     CLOSE h_hostel;
26
27 end;
28
29 /

C:\Users\seha\Desktop\displayhostel.sql 10:32
CTRL LTRF SQL GitHub GDB
Type here to search 7:49 06-05-2019
```

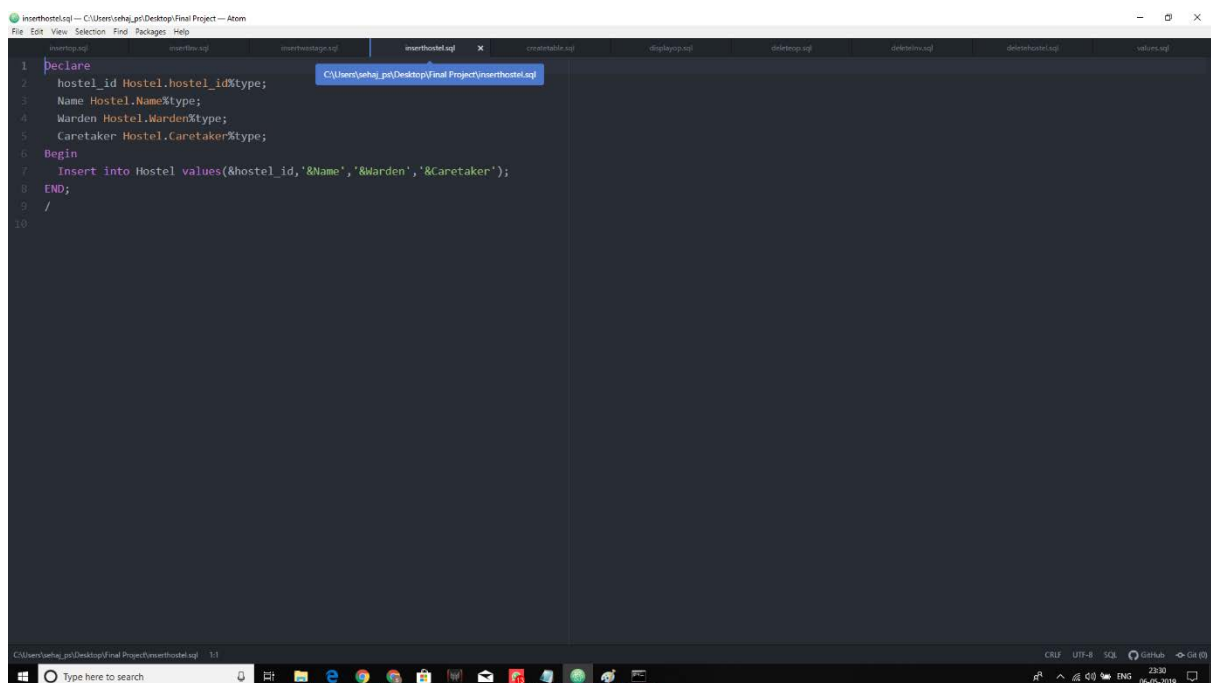
Insertion



The screenshot shows an Atom editor window titled "insertinv.sql - C:\Users\sehaj_pn\Desktop\Final Project - Atom". The editor contains a SQL script with the following code:

```
1 DECLARE
2   inv_id Inventory.inv_id%type;
3   Mastage_id Inventory.Mastage_id%type;
4   Veg_name Inventory.Veg_name%type;
5   Veg_quantity Inventory.inv_id%type;
6   Milk Inventory.Milk%type;
7 BEGIN
8   INSERT into Inventory values(&inv_id, &Mastage_id, '&Veg_name', &Veg_quantity, &Milk);
9 END;
10 /
11
```

The status bar at the bottom indicates the file path "C:\Users\sehaj_pn\Desktop\Final Project\insertinv.sql" and the line number "1:1". The Windows taskbar is visible at the bottom with the search bar and various application icons.



The screenshot shows an Atom editor window titled "inserthostel.sql - C:\Users\sehaj_pn\Desktop\Final Project - Atom". The editor contains a SQL script with the following code:

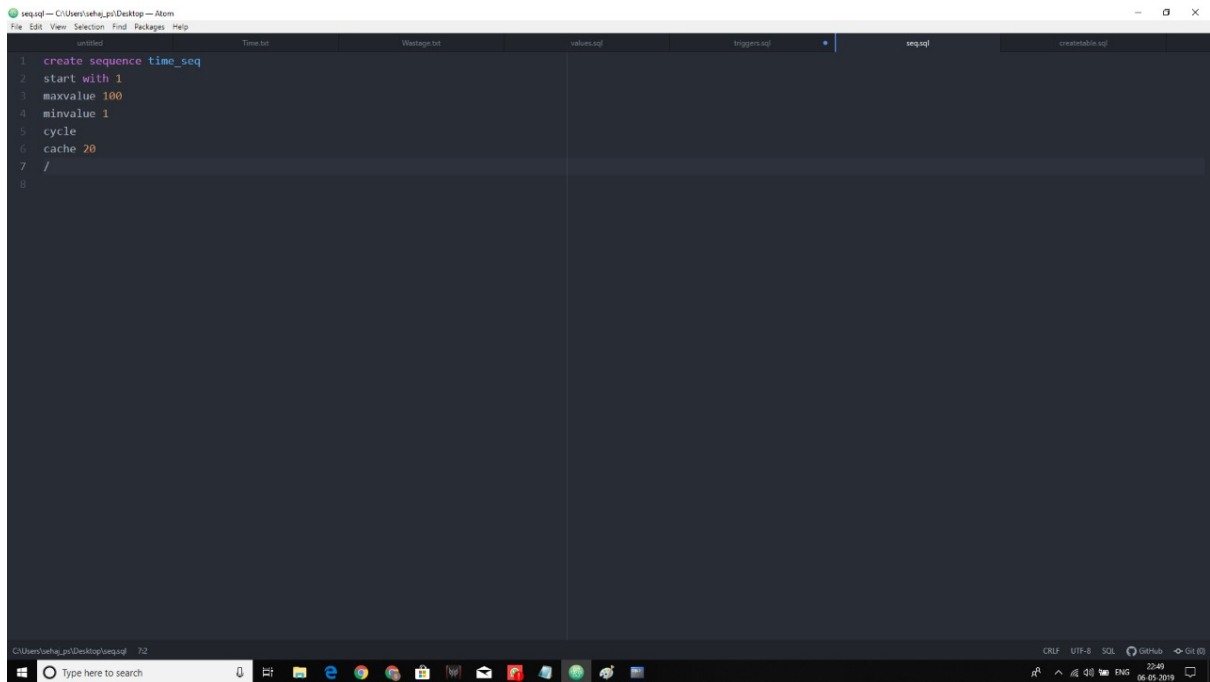
```
1 Declare
2   hostel_id Hostel.hostel_id%type;
3   Name Hostel.Name%type;
4   Warden Hostel.Warden%type;
5   Caretaker Hostel.Caretaker%type;
6 Begin
7   Insert into Hostel values(&hostel_id, '&Name', '&Warden', '&Caretaker');
8 END;
9 /
10
```

The status bar at the bottom indicates the file path "C:\Users\sehaj_pn\Desktop\Final Project\inserthostel.sql" and the line number "1:1". The Windows taskbar is visible at the bottom with the search bar and various application icons.

```
insertop.sql — C:\Users\sehg\Desktop\Final Project — Atom
File Edit View Selection Find Packages Help
insertop.sql insertop.sql insertop.sql insertop.sql
1 DECLARE
2   op_id OverProduction.op_id%type;
3   wastage_id OverProduction.wastage_id%type;
4   rtype OverProduction.rtype%type;
5   quantity OverProduction.quantity%type;
6   rice OverProduction.rice%type;
7   veg_name OverProduction.veg_name%type;
8   veg_quantity OverProduction.veg_quantity%type;
9   milk OverProduction.milk%type;
10 BEGIN
11   Insert into OverProduction values(&op_id, &wastage_id, '&rtype', &quantity, &rice, '&veg_name', &veg_quantity, &milk);
12 END;
13 /
14
```

```
insertwastage.sql — C:\Users\sehg\Desktop\Final Project — Atom
File Edit View Selection Find Packages Help
insertwastage.sql insertwastage.sql insertwastage.sql insertwastage.sql insertwastage.sql insertwastage.sql insertwastage.sql insertwastage.sql insertwastage.sql insertwastage.sql
1 DECLARE
2   wastage_id Wastage.wastage_id%type;
3   hostel_id Wastage.hostel_id%type;
4   time_id Wastage.time_id%type;
5 BEGIN
6   INSERT into Wastage values(&wastage_id,&hostel_id,&time_id);
7 END;
8 /
9
```


Sequences

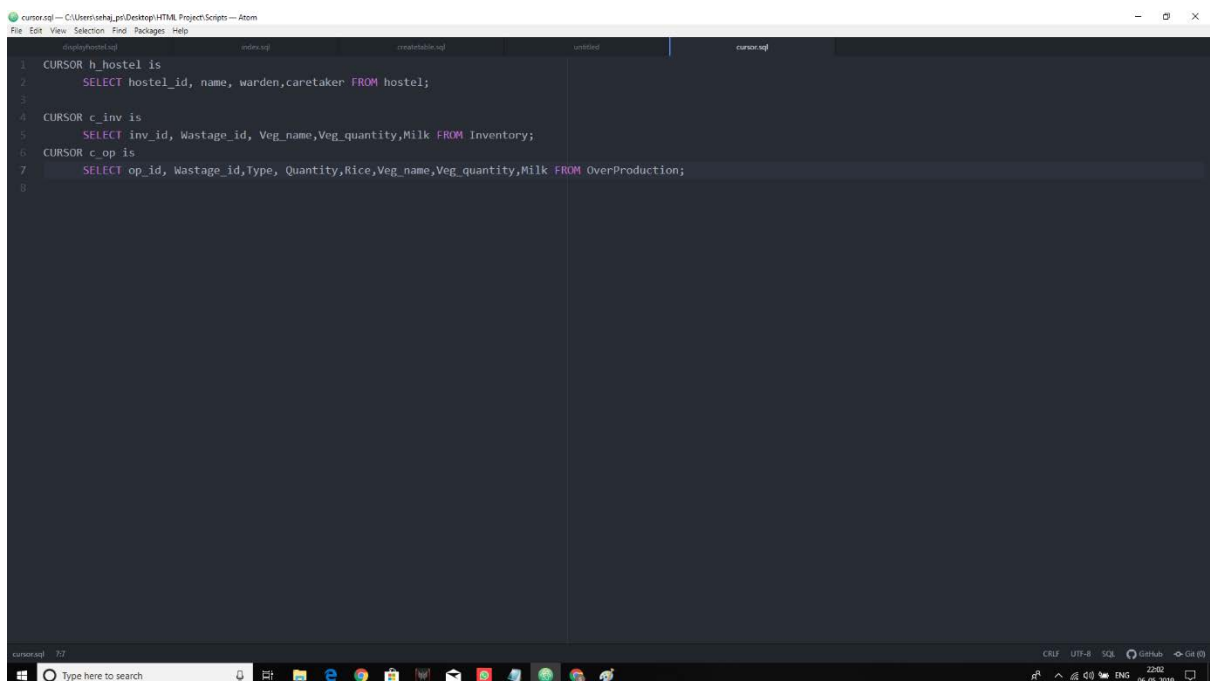


The screenshot shows an Atom editor window with the title bar "seq.sql - C:\Users\sehg_pj\Desktop - Atom". The menu bar includes File, Edit, View, Selection, Find, Packages, and Help. The file explorer on the left shows a project structure with files: untitled, Time.txt, Wastage.txt, values.sql, trigger.sql, seq.sql (selected), and createtable.sql. The main editor area contains the following SQL code:

```
1 create sequence time_seq
2 start with 1
3 maxvalue 100
4 minvalue 1
5 cycle
6 cache 20
7 /
8
```

The status bar at the bottom indicates the file path "C:\Users\sehg_pj\Desktop\seq.sql", the encoding "UTF-8", and the language "SQL". The system tray shows the date and time as "22:49 06-05-2019".

Cursors

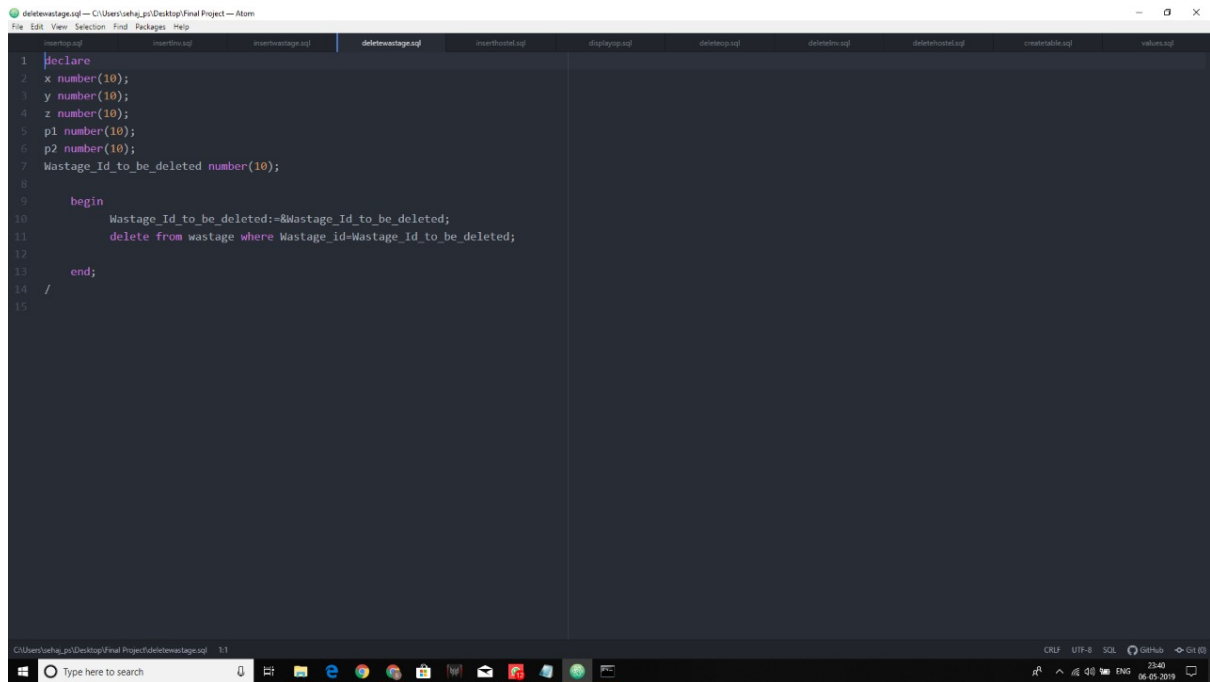


The screenshot shows an Atom editor window with the title bar "cursor.sql - C:\Users\sehg_pj\Desktop\HTML Project Scripts - Atom". The menu bar includes File, Edit, View, Selection, Find, Packages, and Help. The file explorer on the left shows a project structure with files: trigger.sql, createtable.sql, untitled, cursor.sql (selected), and trigger.sql. The main editor area contains the following SQL code:

```
1 CURSOR h_hostel is
2     SELECT hostel_id, name, warden, caretaker FROM hostel;
3
4 CURSOR c_inv is
5     SELECT inv_id, Wastage_id, Veg_name, Veg_quantity, Milk FROM Inventory;
6 CURSOR c_op is
7     SELECT op_id, Wastage_id, Type, Quantity, Rice, Veg_name, Veg_quantity, Milk FROM OverProduction;
8
```

The status bar at the bottom indicates the file path "cursor.sql", the encoding "UTF-8", and the language "SQL". The system tray shows the date and time as "22:52 06-05-2019".

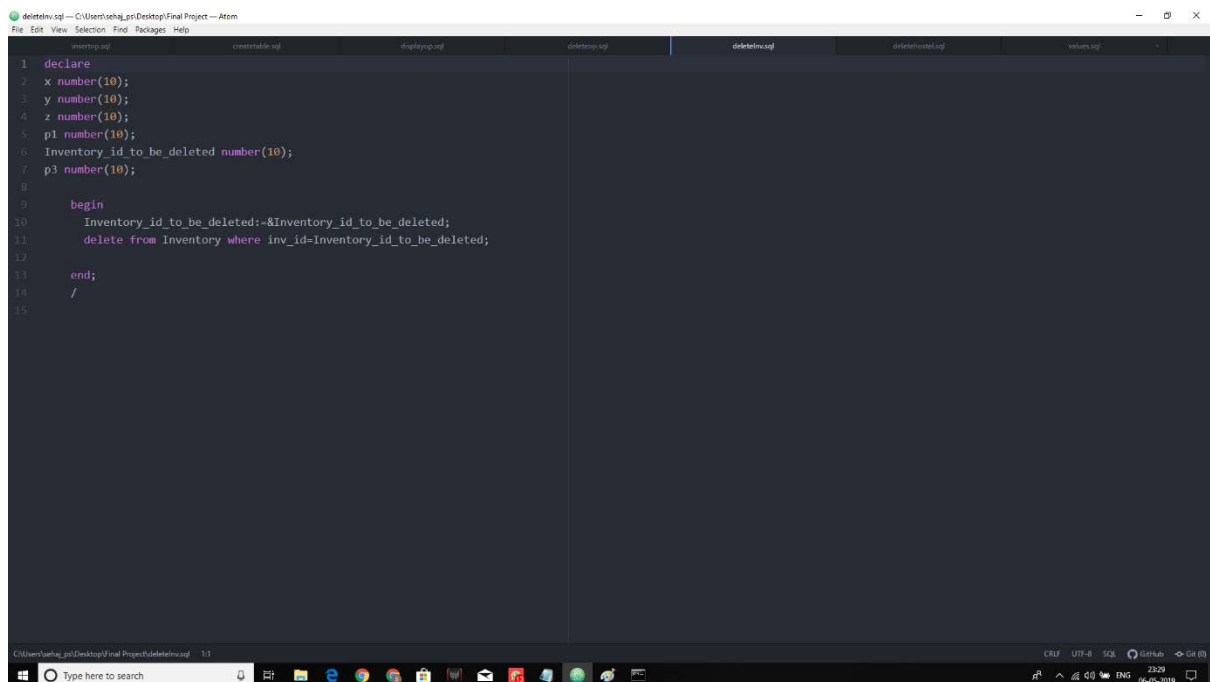
Deletion



The screenshot shows the Atom editor interface with a file named 'deletewastage.sql' open. The script contains the following PL/SQL code:

```
1 declare
2   x number(10);
3   y number(10);
4   z number(10);
5   p1 number(10);
6   p2 number(10);
7   Wastage_Id_to_be_deleted number(10);
8
9   begin
10    Wastage_Id_to_be_deleted:=&Wastage_Id_to_be_deleted;
11    delete from wastage where Wastage_id=Wastage_Id_to_be_Deleted;
12
13   end;
14 /
15
```

The status bar at the bottom indicates the file path 'C:\Users\sahe_pn\Desktop\Final Project\deletewastage.sql' and the time '23:40 06-05-2019'.



The screenshot shows the Atom editor interface with a file named 'deleteinv.sql' open. The script contains the following PL/SQL code:

```
1 declare
2   x number(10);
3   y number(10);
4   z number(10);
5   p1 number(10);
6   Inventory_id_to_be_deleted number(10);
7   p3 number(10);
8
9   begin
10    Inventory_id_to_be_deleted:=&Inventory_id_to_be_deleted;
11    delete from Inventory where inv_id=Inventory_id_to_be_deleted;
12
13   end;
14 /
15
```

The status bar at the bottom indicates the file path 'C:\Users\sahe_pn\Desktop\Final Project\deleteinv.sql' and the time '23:29 06-05-2019'.

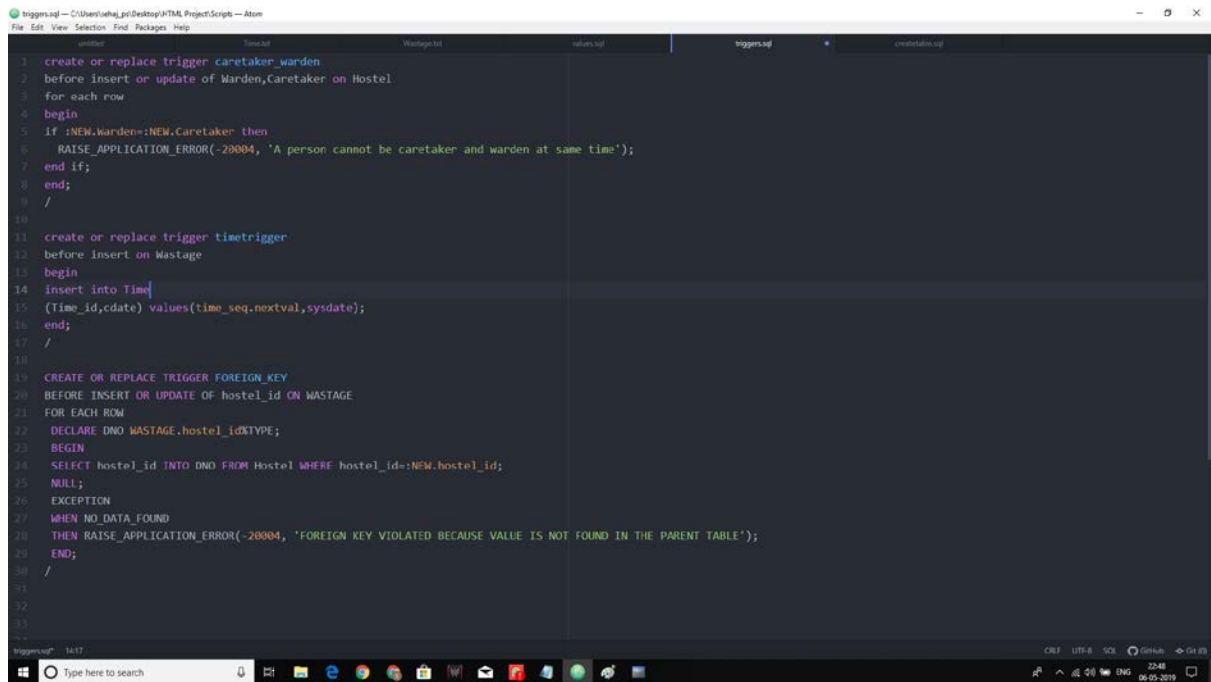
```
deleteop.sql — C:\Users\usba_pr\Desktop\Final Project — Atom
File Edit View Selection Find Packages Help
deleteop.sql
1 declare
2 x number(10);
3 y number(10);
4 z number(10);
5 p1 number(10);
6 p2 number(10);
7 Over_Prod_Id_to_be_deleted number(10);
8
9 begin
10 Over_Prod_Id_to_be_deleted:=&Over_Prod_Id_to_be_deleted;
11 delete from OverProduction where op_id=Over_Prod_Id_to_be_deleted;
12
13 end;
14 /
15

C:\Users\usba_pr\Desktop\Final Project\deleteop.sql 13.7
CRUF UTF-8 SQL GitHub Git
Type here to search 23:28 06-05-2019
```

```
deletehostel.sql — C:\Users\usba_pr\Desktop\Final Project — Atom
File Edit View Selection Find Packages Help
deleteop.sql deletehostel.sql
1 declare
2 x number(10);
3 y number(10);
4 z number(10);
5 hostel_id_to_be_deleted number(10);
6 p2 number(10);
7 p3 number(10);
8
9
10 begin
11 hostel_id_to_be_deleted:=&hostel_id_to_be_deleted;
12 delete from Hostel where hostel_id=hostel_id_to_be_deleted;
13
14 end;
15 /
16

C:\Users\usba_pr\Desktop\Final Project\deletehostel.sql 1.1
CRUF UTF-8 SQL GitHub Git
Type here to search 23:29 06-05-2019
```

Trigger



The screenshot shows a code editor window titled "triggers.sql" with a menu bar (File, Edit, View, Selection, Find, Packages, Help) and a toolbar. The editor contains three tabs: "values.sql", "triggers.sql", and "createTables.sql". The "triggers.sql" tab is active and displays the following SQL code:

```
1 create or replace trigger caretaker_warden
2 before insert or update of Warden,Caretaker on Hostel
3 for each row
4 begin
5 if :NEW.Warden=:NEW.Caretaker then
6   RAISE_APPLICATION_ERROR(-20004, 'A person cannot be caretaker and warden at same time');
7 end if;
8 end;
9 /
10
11 create or replace trigger timetrigger
12 before insert on Wastage
13 begin
14 insert into Time
15 (Time_id,cdate) values(time_seq.nextval,sysdate);
16 end;
17 /
18
19 CREATE OR REPLACE TRIGGER FOREIGN_KEY
20 BEFORE INSERT OR UPDATE OF hostel_id ON WASTAGE
21 FOR EACH ROW
22 DECLARE DNO WASTAGE.hostel_id%TYPE;
23 BEGIN
24 SELECT hostel_id INTO DNO FROM Hostel WHERE hostel_id=:NEW.hostel_id;
25 NULL;
26 EXCEPTION
27 WHEN NO_DATA_FOUND
28 THEN RAISE_APPLICATION_ERROR(-20004, 'FOREIGN KEY VIOLATED BECAUSE VALUE IS NOT FOUND IN THE PARENT TABLE');
29 END;
30 /
31
32
33
```

The Windows taskbar is visible at the bottom, showing the search bar, task view button, and several application icons. The system tray on the right shows the date and time as 22:48 on 06-05-2019.

REFERENCES

- <https://sites.google.com/site/dbms310cse/lecture-notes>
- www.stackoverflow.com
- www.oracle.com
- Oracle 9i SQL.pdf