SCHOOL MANAGEMENT SYSTEM

WORK BY:

1.LANKA JASWANTH (19BIT0061)
2.YESWIN CHOWDARY (19BIT0134)
3.B K ANIRUDH (19BIT0348)

GUIDED BY:

PROF.BIMAL KUMAR RAY

INTRODUCTION

The education system forms the backbone of every nation. Hence it is important and necessary to provide a strong educational foundation to the young generation to ensure the development of open-minded citizens securing the future for everyone. Advanced technology today can play a crucial role in streamlining the education-related processes to promote solidarity among students, teachers, parents, and the school staff.

Existing System

In the current system, we need to keep the number of records related to students and want to enter the marks and records of the students manually. In this system teacher or school, authority views the marks and records of the student and keeps track of the student manually. School authority needs to keep track of all the members of the school including students. This is very time consuming and has a lot of paperwork.

Need for the System

In our proposed system, We have the provision for adding the details pf the student. The overhead of the school authorities and teachers becomes less. Another advantage of the system is that it is very easy to edit the details of the student and delete the records if necessary.

Advantages

- Fast access to the database
- Fewer errors
- More storage capacity
- Student and faculty friendly
- Cost convenient
- Less manpower

The system will be used for conducting day to day teaching activities such as assigning homework to conduct exams along with management tasks such as checking the fees status of the student, keeping a record of the number of students enrolled in a class and subject. It is not easy to do this process manually because it would become very hectic. Hence it is recommended to automate the process by developing the relevant software as the world is moving from manual working to information and technology era where computerization becomes important in all parts of life.

Data Requirements

The school management system has a set of data requirements. The following are the entity types that will be used in our school management system.

1.ADMIN_DPT:

This entity consists of A_ID which is unique to each administrator be it a principal or supervisor. The other attributes are Name, Salary, Designation, Qualification, Email, Phone_no.

2.CLASS:

CLASS as an entity consists of attributes such as Std_Sec which is unique to each class, Total number of students.

3.TEACHER:

TEACHER as an entity consists of attributes T_ID which is unique to each teacher, Qualification, Salary, Name, Phone_number, Class_taken.

4.EXAM:

EXAM as an entity consists of attributes Exam_code which is unique to every exam conducted, Exam_sub, Exam_date, Exam_qpaper.

5.HOMEWORK:

HOMEWORK as an entity consists of HW_ID which is used to uniquely identify homework given by faculty, Sub_assign, Due_date, Numer_of_students.

6.MARKS:

MARKS as an entity type consists of Paper_no as identifying key, Marks_obt, Marks_tot, Subject.

7.SUBJECT:

SUBJECT entity type has attributes Sub_code as identifying key, Name, ,Total_stu_enrolled as other attributes.

8.STUDENT:

STUDENT as an entity type consists of attributes such as Roll_no unique to every student, Section, Name, Class, Fees.

9.PARENT:

PARENT is a **Weak entity of Student** consisting of attributes such as Roll_no which is unique to every parent, name, Phone_num.

Relationships

- Parent GUARDS Student (1- 1)
  Each parent guards only one student and Parent exist only if he has a student in the school. Hence participation of Parent in GUARDS relation is **Total Participation.(Total participation of Parent and Partial participation of student in this relationship)**
- Teahcer GIVES Marks(1-N)
  Each Teacher gives many marks to many students in the miniworld.Hence the relationships becomes (1-N) relation ship.**(Partial Participation of Teacher and Marks in Moniters relationship)**
- Teacher KEEPS Exam (1-1)
  Each Teacher has to post only 1 test regarding his subject. So, the relationship is a 1-1 relationship. **(Partial Participation of both Teacher and Exam)**
- Teacher GIVES Homework(1-N)
  A Teacher may give may Homeworks regarding his subject. So, the relationship becomes a 1-N relationship. **(Partial Participation of both Teacher and Homework)**
- Teacher SPECIALIZE Subject(1-1)
  A Teacher may teach only 1 subject. Hence, the relationship becomes a 1-1 relationship. **(Partial Participation of both Teacher and Subject)**
- Teacher TEACHES Class(1-N)
  A Teacher may have many classes in a day. Hence, the relationship is a 1-N relationship. **(Partial Participation of both Teacher and Class)**
- Admin_dpt MANAGES Class(1-N)
  An administrator(Principal or Supervisor) have to manage all the class. Hence, the relationship is a 1-N relationship. **(Partial Participation of both Admin_dpt and Class)**
- Admin_dpt SUPERVISES Teacher(1-N)
  An administrator(Principal or Supervisor) has to manage all the teachers of the school. Hence, the relationship is a 1-N relationship. **(Partial Participation of both Admin_dpt and Teacher)**

Functional Requirements

Removal of Data

There are certain scenarios when the data stored with the School Management System has to be deleted. This could arise from a situation such as a student, faculty, admin leaving his/her job. In such cases, the data stored about the individual must be deleted to prevent unnecessary storage of data and to ensure the security of our data. Scenarios demanding deletion/removal of data are discussed below:

Case 1:
A situation may arise where the number of students enrolled in a given class is less than the expected number, say 10. In such a case, the class of that particular subject is deleted.

Case 2:
If in case a teacher wants to leave school.in such cases, we have to remove his/her account as well as all the data associated with it so that he/she could not further alter the marks or the homework of students.

Case 3:
There are certain situations such as when the students want to leave school. In such cases, we have to remove his/her account as well as all the data associated with it.

Case 4:
If in case a course does not exist anymore i.e. the school no longer offers a particular course then we have to delete all information regarding that subject.

Modification of Data

School management system a highly efficient system. Therefore, it has to be updated regularly to maintain the correct set of data. And there are situations where the details of an individual changes

Case 1:
Change of Contact Number In case, a student's parent has a change of contact number which has been reported. Then, the admin, on request from the user, changes the student's parent's number
.
Case 2:
In case the marks entered by a teacher has to be changed then only the particular subject teacher is allowed to change the marks

Case 3:
There could be a case where the question paper is to be changed may be due to a moderation

Case 4:
In case a teacher wishes to change the deadline for homework. This function can only be performed by the subject teacher.

Selection of Data

Data retrieval is an operation performed by the system on request by the user or the admin to generate the output which is asked by the user. This could include arithmetic operations such as average or could be simply filtering something out of the database

Case 1:
print out all the students who have secured first division in a particular subject. (first division is marked greater than 95%)

Case 2:
Similarly, we can search for the admins who own an engineering degree.

Case 3:
our System can print the fee status of a student which could be paid. This would be highly helpful in knowing the students whose fees are still pending.

Case 4:
Similarly, we can also find students who have paid the fees.

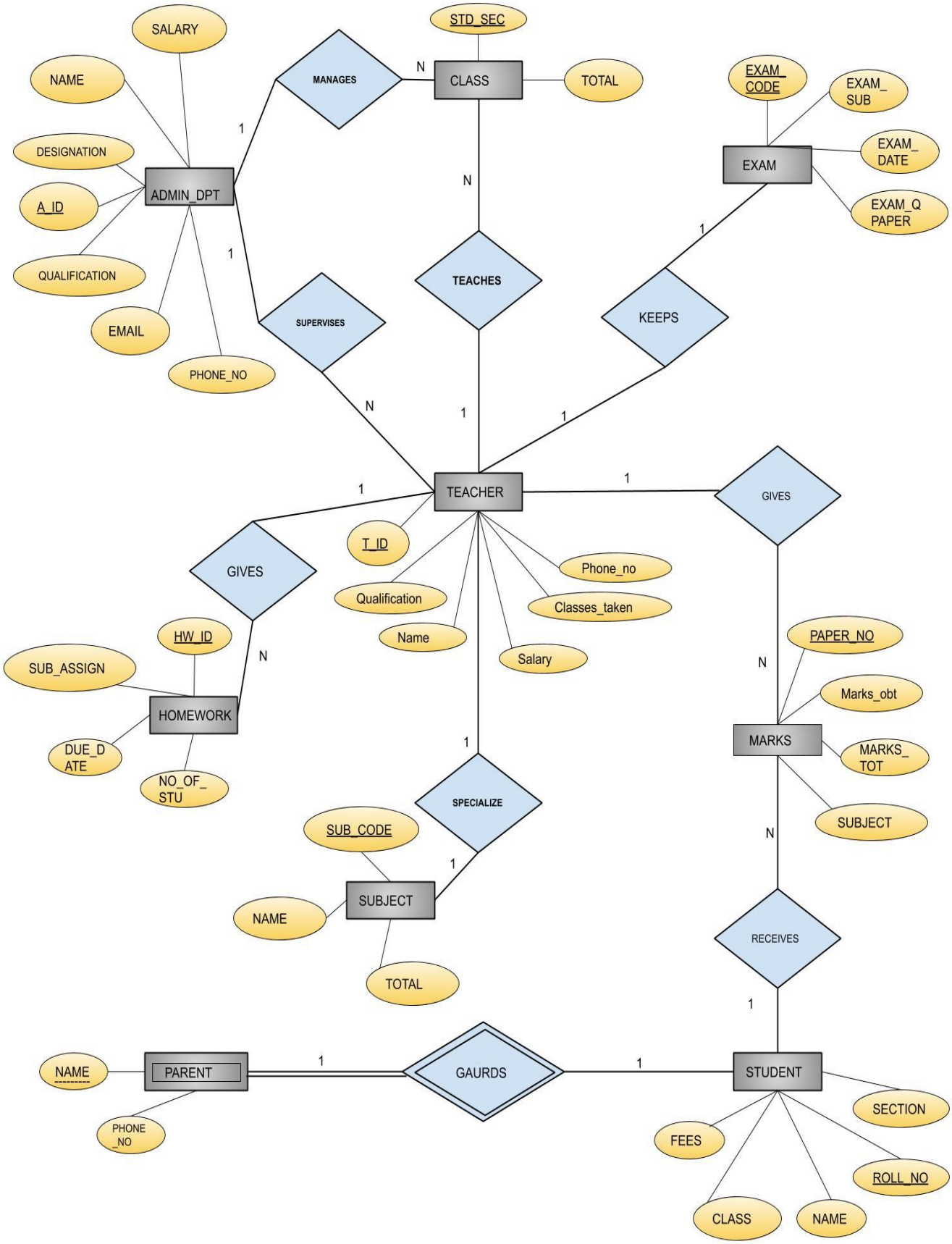
Case 5:
Print students who have received more than 40 in one or more subjects.

Case 6:
using teacher id we could see permanent teachers in the school

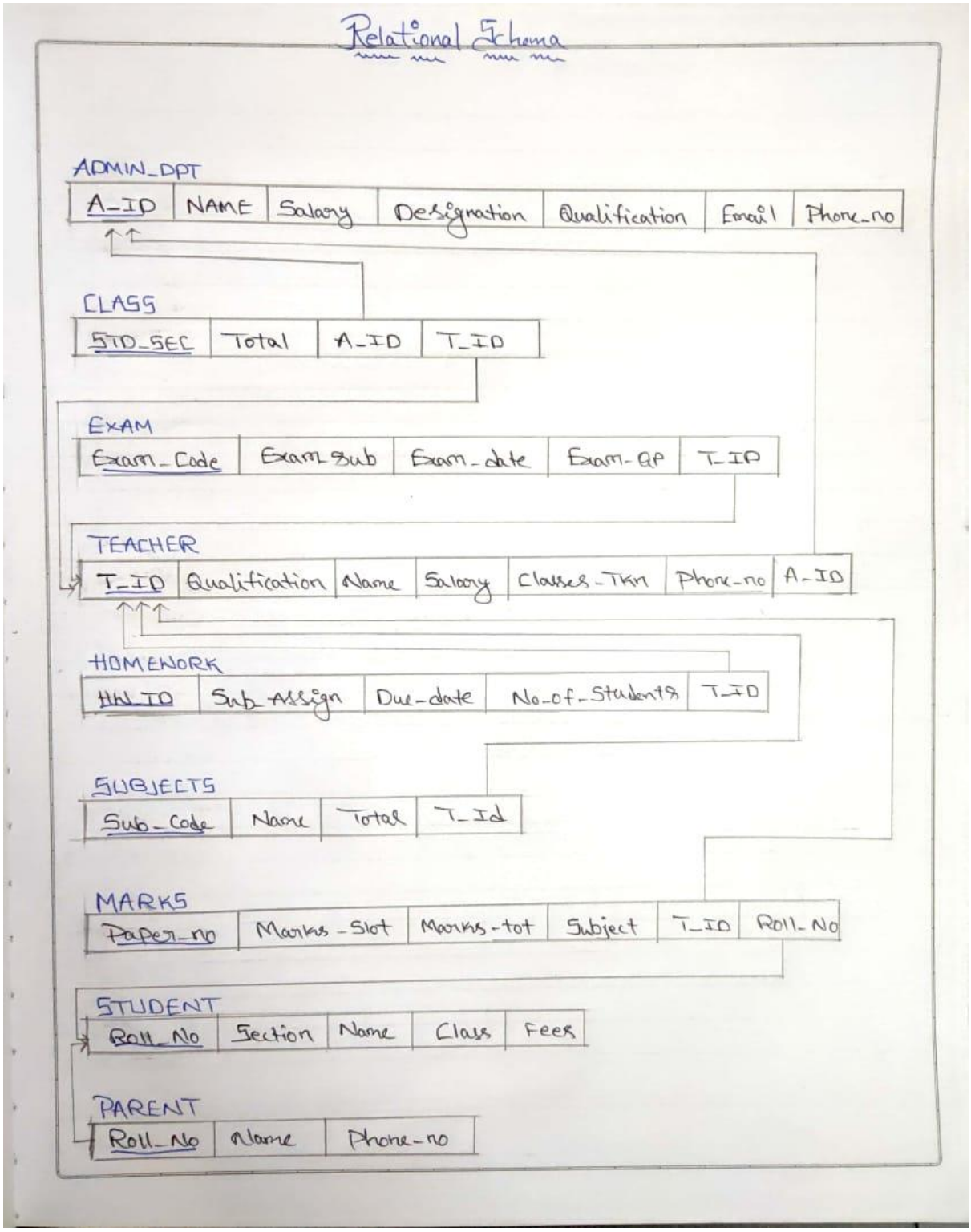
Case 7:
Teachers supervised by a specific coordinator

Case 8:
We can find/filter the teachers who own a Doctorate.

ER DIAGRAM

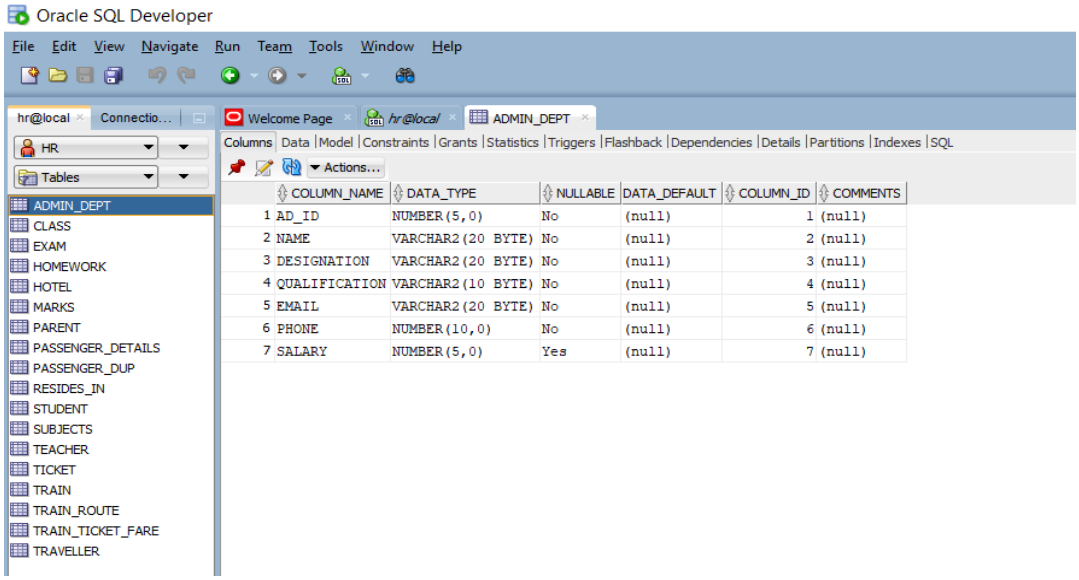

SALARY
STD_SEC
NAME
MANAGES
N
CLASS
TOTAL
EXAM_ CODE
EXAM_ SUB
1
DESIGNATION
EXAM
EXAM_ DATE
ADMIN_DPT
N
A_ID
EXAM_Q PAPER
1
QUALIFICATION
TEACHES
SUPERVISES
KEEPS
EMAIL
PHONE_NO
N
1
1
1
1
TEACHER
GIVES
GIVES
T_ID
Phone_no
Qualification
Classes_taken
Name
Salary
HW_ID
PAPER_NO
N
N
SUB_ASSIGN
Marks_obt
HOMEWORK
MARKS
DUE_D ATE
MARKS_ TOT
1
SPECIALIZE
NO_OF_ STU
SUB_CODE
SUBJECT
N
1
NAME
SUBJECT
RECEIVES
TOTAL
1
NAME
PARENT
1
GAURDS
1
STUDENT
SECTION
PHONE _NO
FEES
ROLL_NO
CLASS
NAME

RELATIONAL SCHEMA



CREATION OF TABLES

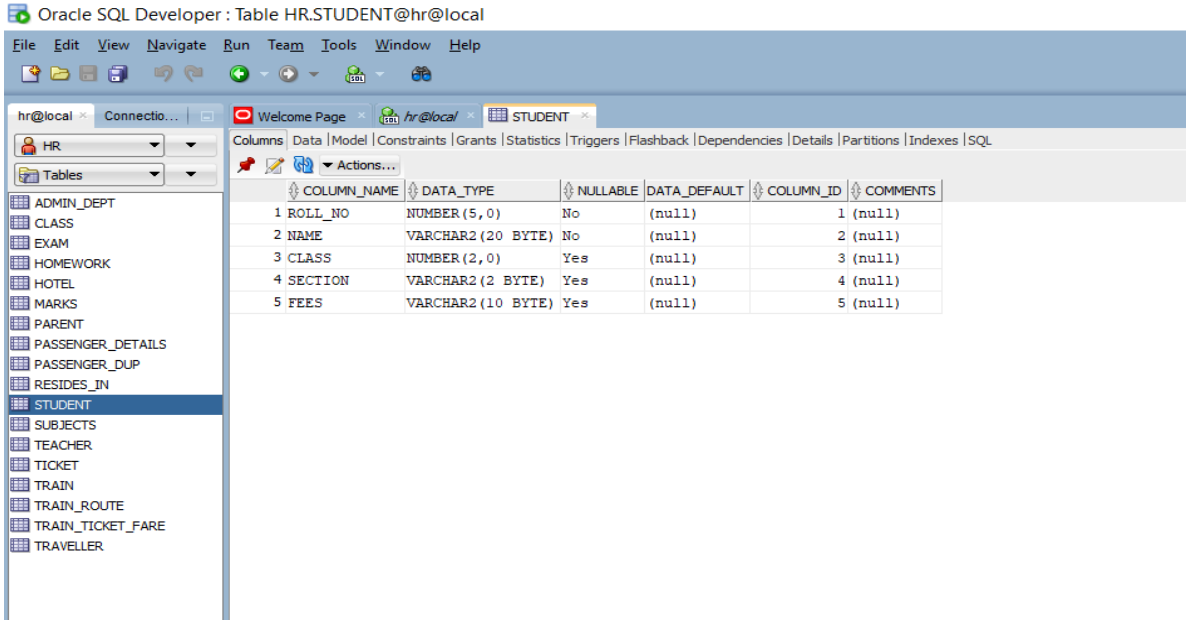
'Admin_dpt':



CODE:

```
create table Admin_Dept(
AD_ID number(5) constraint admin_pk primary key,
NAME varchar(20) not null,
Designation varchar(20) not null,
qualification varchar(10) not null,
email varchar(20) not null,
phone number(10) not null,
salary number(5));
```

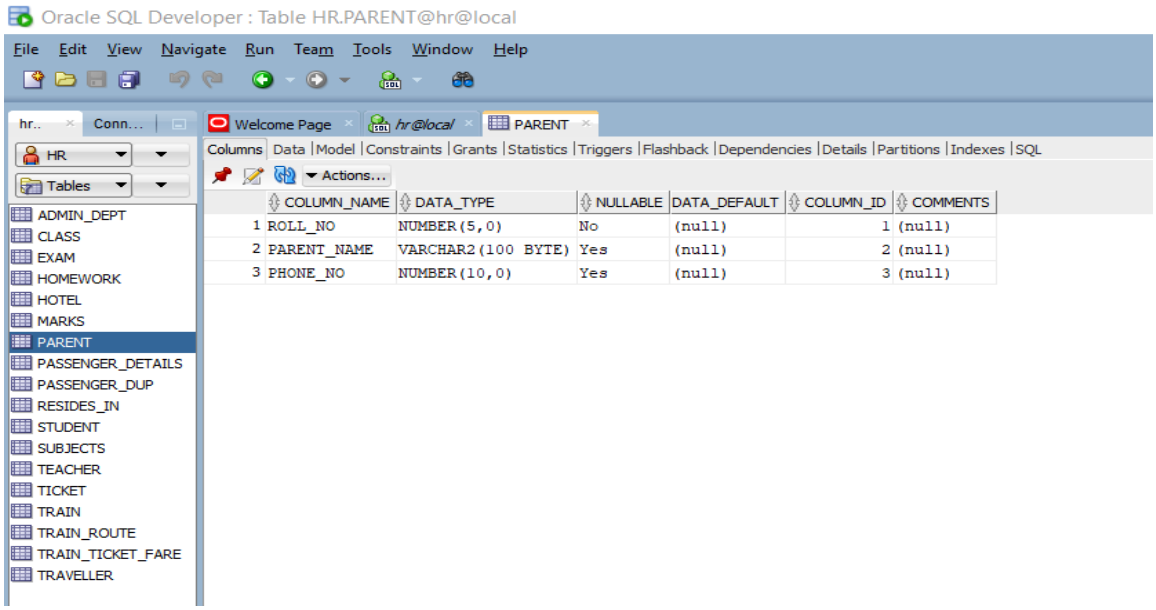
‘Student’:



CODE:

```
create table student(
roll_no number(5) constraint roll_pk primary key,
name varchar(20) constraint n_null not null,
class number(2),
section varchar(2), fees varchar(10));
```

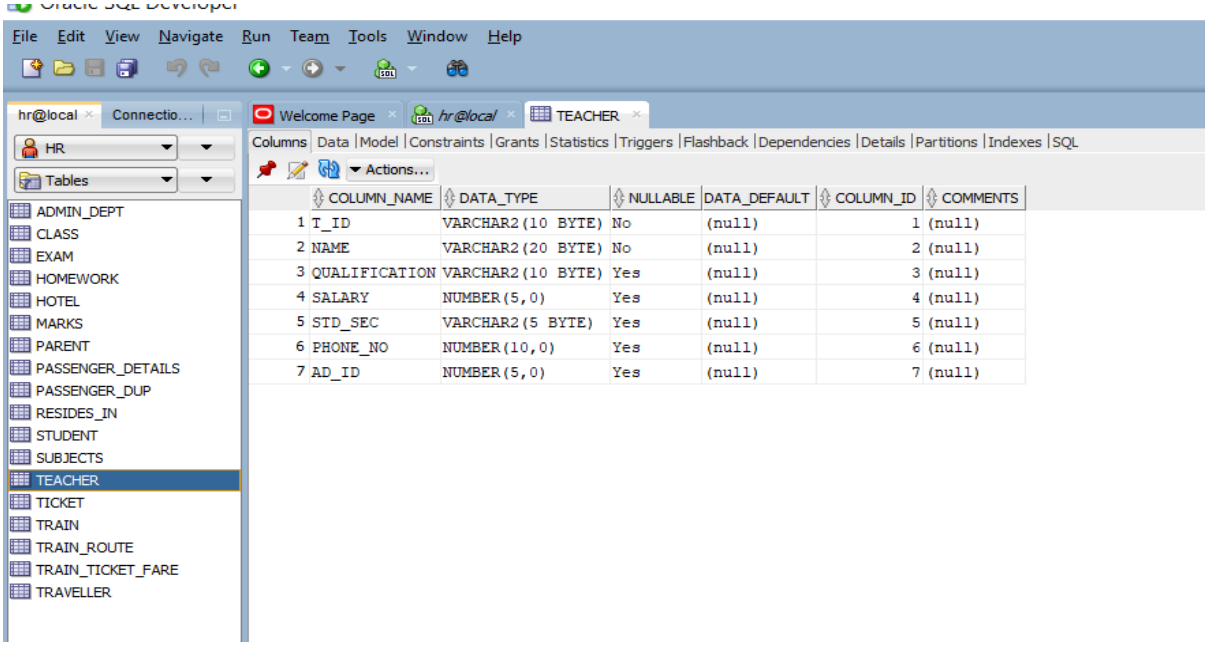
'Parent':



CODE:

```
Create table parent(roll_no number(5) references student,
Parent_name varchar(100),
Phone_No number(10),
primary key(Parent_name,roll_no));
```

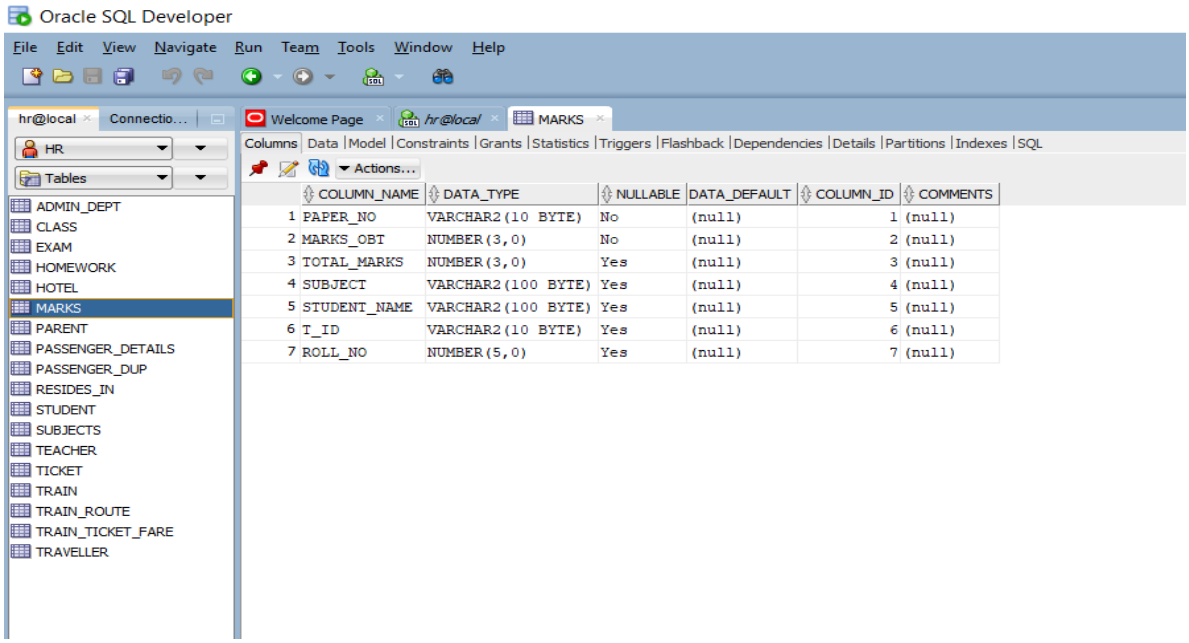
'Teacher':



CODE:

```
create table Teacher(
t_id varchar(10) constraint t_id_pk primary key,
name varchar(20) constraint name_nnull not null,
Qualification varchar(10),
salary number(5),
std_sec varchar(5),
phone_no number(10),
Ad_id number(5),
constraint teacher_adid_fk foreign key(AD_ID) references Admin_Dept(AD_ID));
```

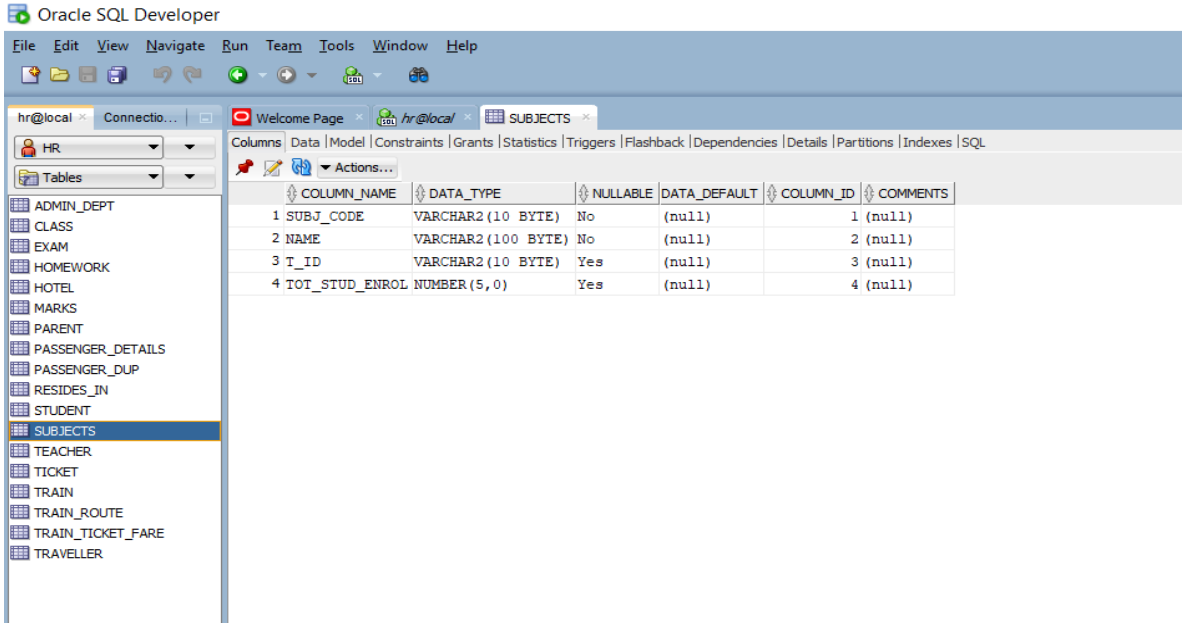
'Marks':



CODE:

```
Create table marks(
Paper_No  varchar(10)  constraint mark_pk primary key,
Marks_obt  number(3)  constraint mk_obt_nnull not null,
Total_marks number(3),
Subject varchar(100),
Student_name varchar(100),
T_ID varchar(10),
Roll_No number(5),
Constraint mk_fk1 foreign key(T_ID) references teacher(T_ID),
Constraint mk_fk2 foreign key(Roll_No) references student(Roll_No));
```

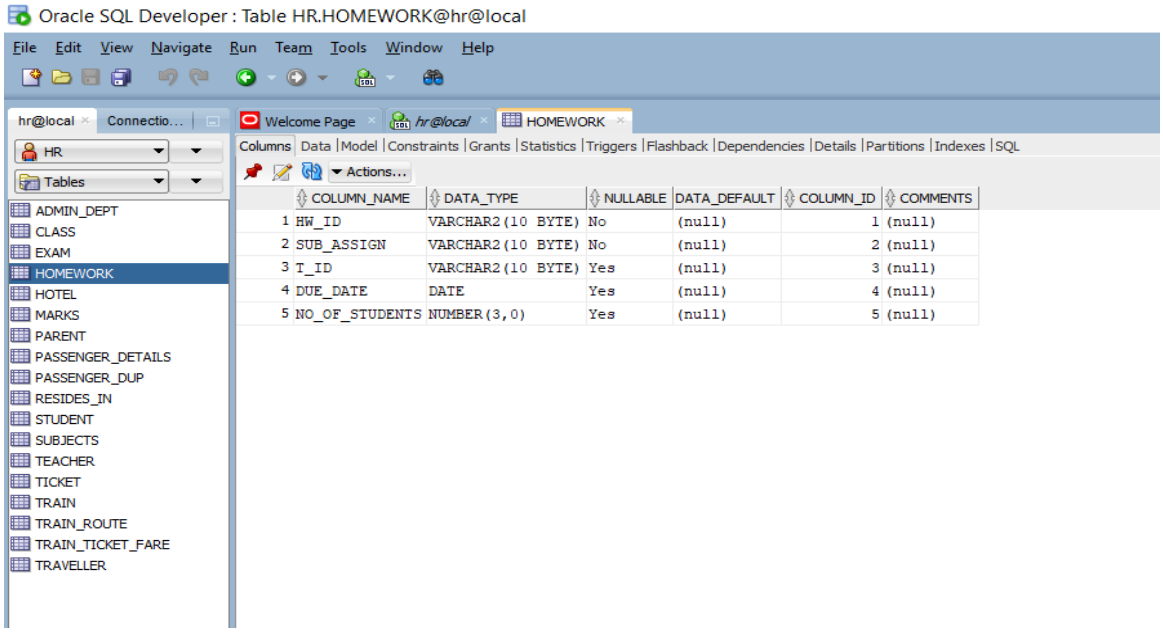
'Subjects':



CODE:

```
Create table subjects(
Subj_code varchar(10) constraint sub_code_pk primary key,
Name varchar(100) constraint subn_nnull not null,
T_ID varchar(10),
Tot_stud_enrol number(5),
Constraint subt_fk foreign key(T_ID) references teacher(T_ID));
```

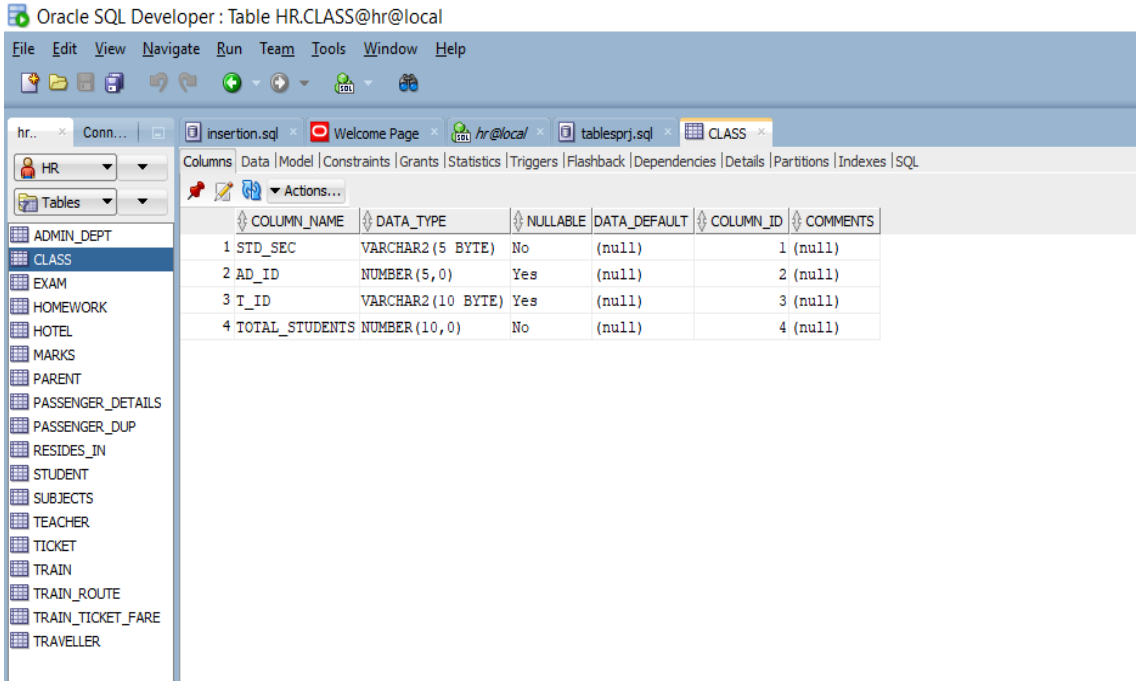
'Homework':



CODE:

```
Create table Homework(
Hw_id varchar(10) constraint hw_id_pk primary key,
Sub_assign varchar(10) constraint sub_nnull not null,
T_ID varchar(10),
Due_date date,
No_of_students number(3),
Constraint hw_fk foreign key(T_ID) references teacher(T_ID));
```

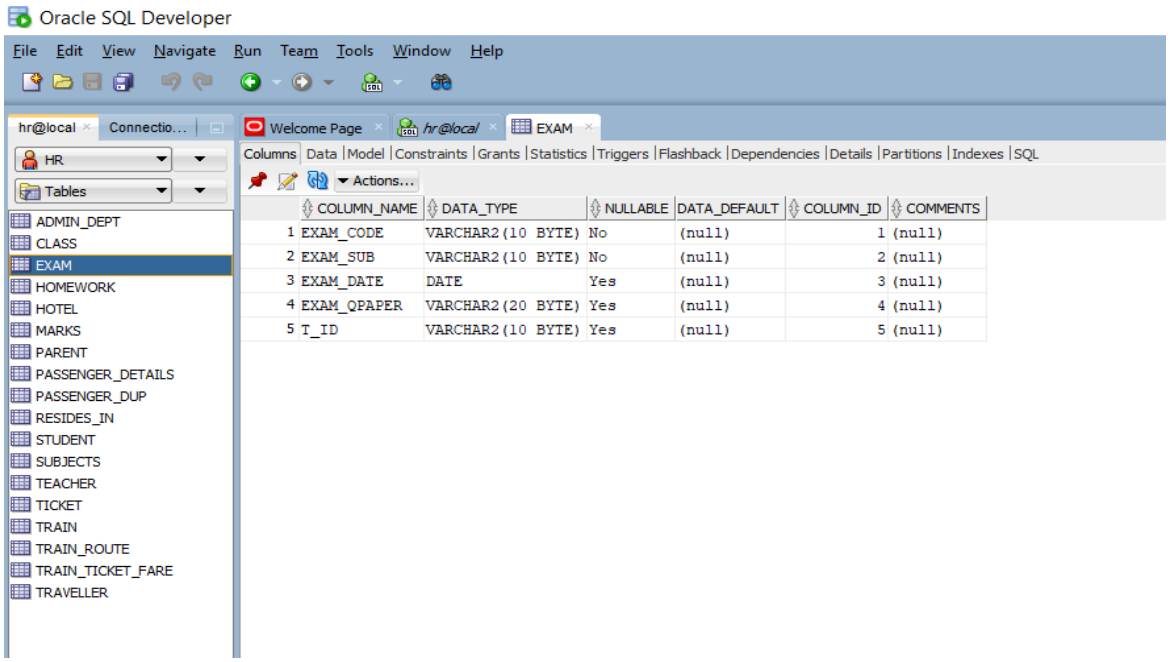
'Class':



CODE:

```
Create table Class(
Std_sec varchar(5) constraint std_sc_pk primary key,
AD_ID number(5),
t_id varchar(10),
Total_students number(10) constraint ts_nnull not null,
Constraint cl_ad_fk foreign key(AD_ID) references Admin_Dept(AD_ID),
CONSTRAINT C_T_FK foreign key(t_id) references Teacher(t_id));
```

'Exam':



CODE:

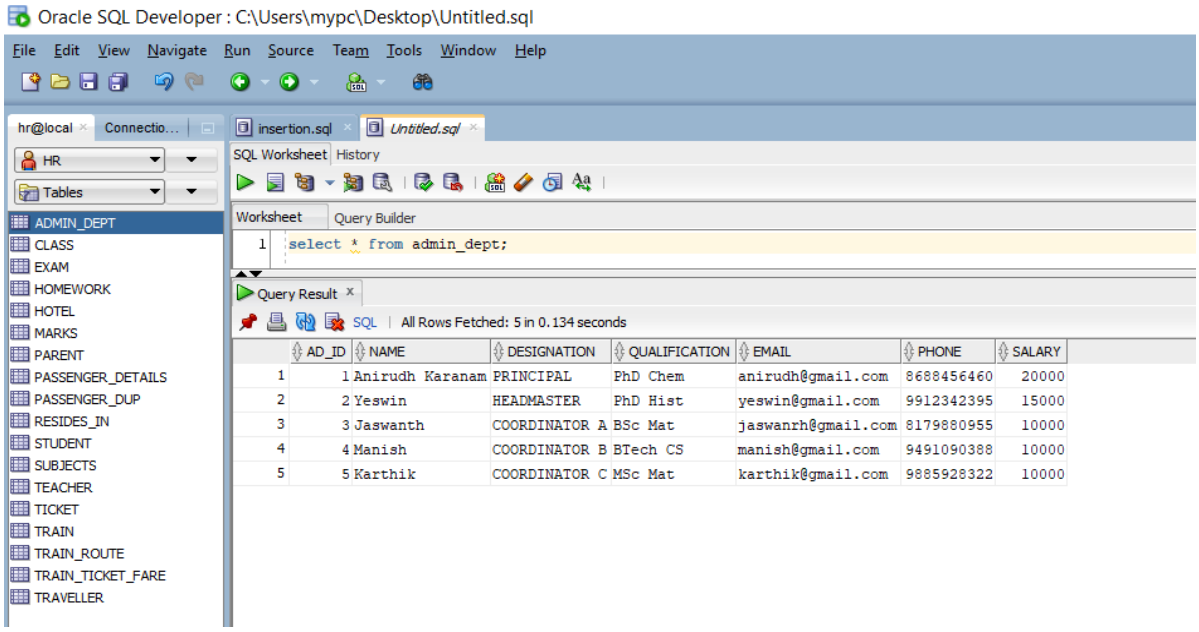
```
Create table Exam(
Exam_code varchar(10) constraint ex_cd_pk primary key,
Exam_sub varchar(10) constraint ex_sb_nnull not null,
Exam_date date,
Exam_qpaper varchar(20),
T_ID varchar(10),
Constraint ex_fk foreign key(T_ID) references teacher(T_ID));
```

INSERTION OF DATA:

'Admin_dept':

```
INSERT INTO ADMIN_DEPT values(00001,'Anirudh Karanam','PRINCIPAL','PhD Chem','anirudh@gmail.com',8688456460,20000);

INSERT INTO ADMIN_DEPT VALUES(00002,'Yeswin','HEADMASTER','PhD Hist','yeswin@gmail.com',9912342395,15000);

INSERT INTO ADMIN_DEPT VALUES(00003,'Jaswanth','COORDINATOR A','BSc Mat','jaswanrh@gmail.com',8179880955,10000);

INSERT INTO ADMIN_DEPT VALUES(00004,'Manish','COORDINATOR B','BTech CS','manish@gmail.com',9491090388,10000);

INSERT INTO ADMIN_DEPT VALUES(00005,'Karthik','COORDINATOR C','MSc Mat','karthik@gmail.com',9885928322,10000);

commit;
```

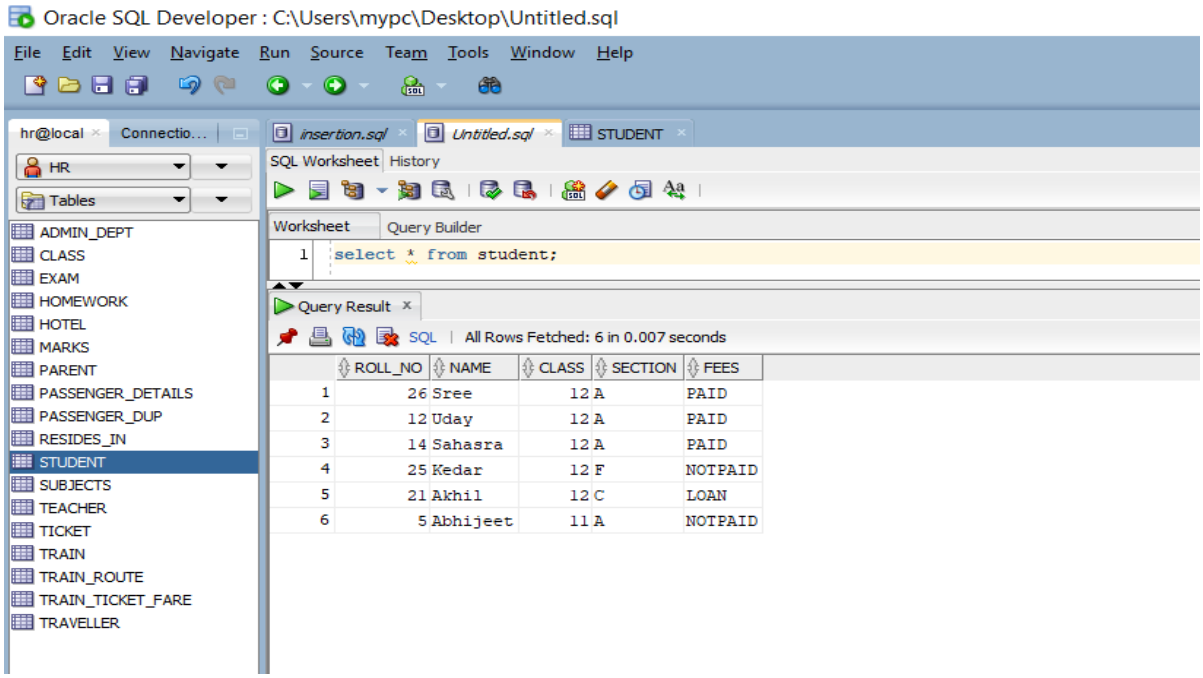
OUTPUT



'Student':

```
INSERT INTO STUDENT VALUES(26,'Sree',12,'A','PAID');
INSERT INTO STUDENT VALUES(12,'Uday',12,'A','PAID');
INSERT INTO STUDENT VALUES(14,'Sahasra',12,'A','PAID');
INSERT INTO STUDENT VALUES(25,'Kedar',12,'F','NOTPAID');
INSERT INTO STUDENT VALUES(21,'Akhil',12,'C','LOAN');
INSERT INTO STUDENT VALUES(05,'Abhijeet',11,'A','NOTPAID');
commit;
```

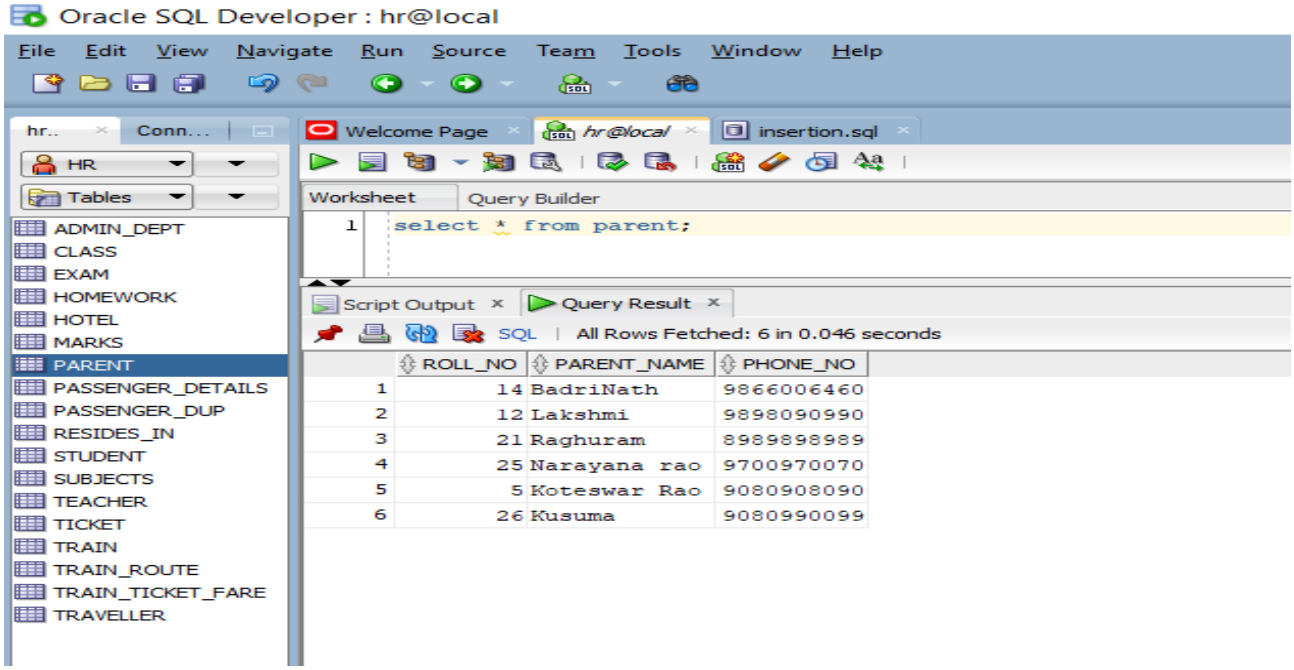
OUTPUT



'Parent':

```
INSERT INTO PARENT VALUES(14,'BadriNath',9866006460);
INSERT INTO PARENT VALUES(12,'Lakshmi',9898090990);
INSERT INTO PARENT VALUES(21,'Raghuram',8989898989);
INSERT INTO PARENT VALUES(25,'Narayana rao',9700970070);
INSERT INTO PARENT VALUES(05,'Koteswar Rao',9080908090);
INSERT INTO PARENT VALUES(26,'Kusuma',9080990099);
commit;
```

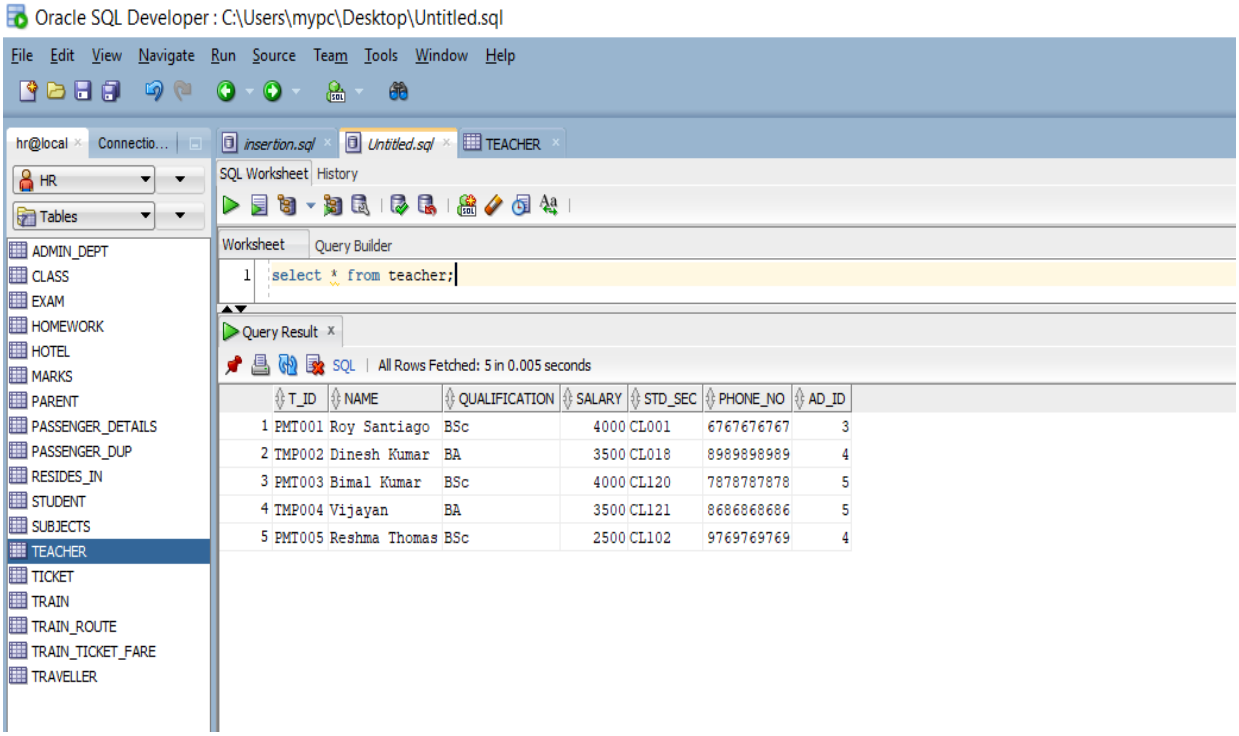
OUTPUT



'Teacher':

```
INSERT INTO TEACHER VALUES('PMT001','Roy Santiago','BSc',4000,'CL001',6767676767,3);

INSERT INTO TEACHER VALUES('TMP002','Dinesh Kumar','BA',3500,'CL018',8989898989,4);

INSERT INTO TEACHER VALUES('PMT003','Bimal Kumar','BSc',4000,'CL120',7878787878,5);

INSERT INTO TEACHER VALUES('TMP004','Vijayan','BA',3500,'CL121',8686868686,5);

INSERT INTO TEACHER VALUES('PMT005','Reshma Thomas','BSc',2500,'CL102',9769769769,4);

commit;
```

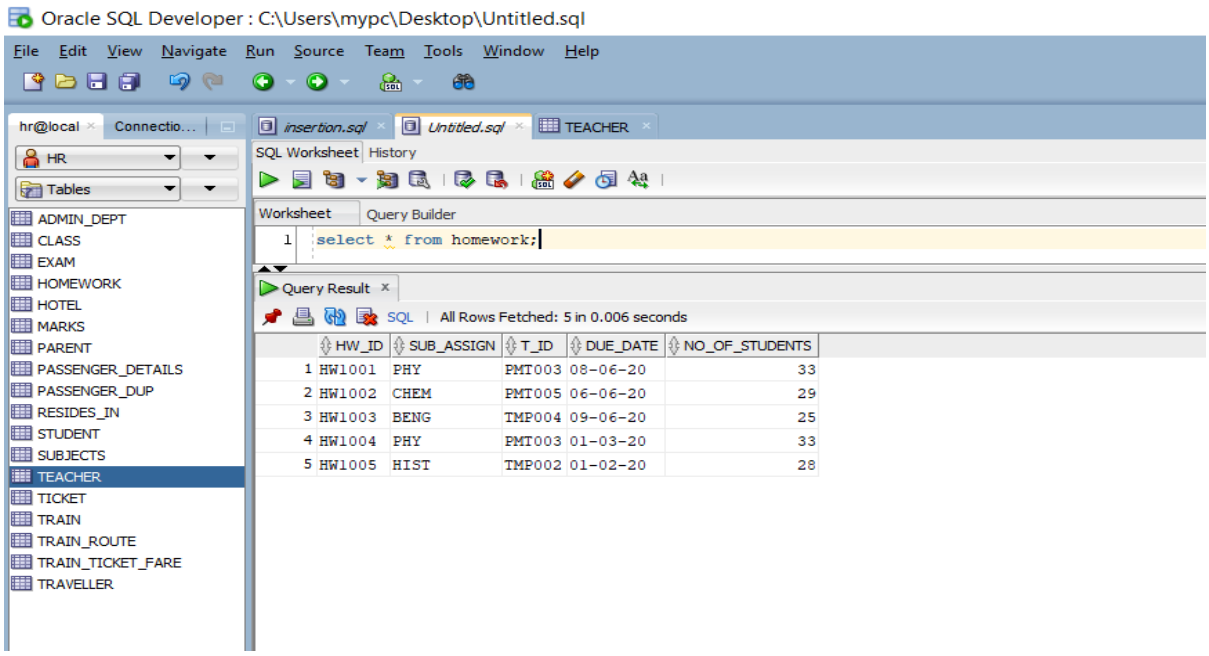
OUTPUT



'Homework':

```
INSERT INTO HOMEWORK VALUES('HW1001','PHY','PMT003',DATE'2020-06-08',33);

INSERT INTO HOMEWORK VALUES('HW1002','CHEM','PMT005',DATE'2020-06-06',29);

INSERT INTO HOMEWORK VALUES('HW1003','BENG','TMP004',DATE'2020-06-09',25);

INSERT INTO HOMEWORK VALUES('HW1004','PHY','PMT003',DATE'2020-03-01',33);

INSERT INTO HOMEWORK VALUES('HW1005','HIST','TMP002',DATE'2020-02-01',28);

commit;
```

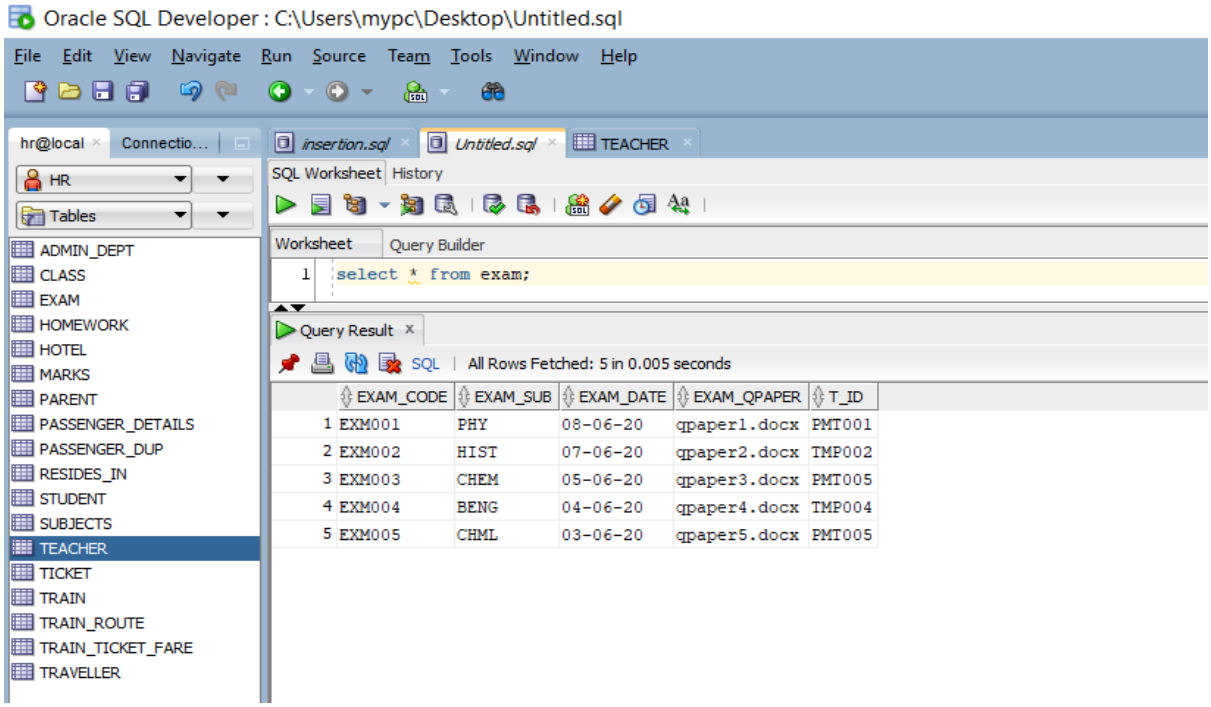
OUTPUT



'Exam'

```
INSERT INTO EXAM VALUES('EXM001','PHY',DATE'2020-06-08','qpaper1.docx','PMT001');

INSERT INTO EXAM VALUES('EXM002','HIST',DATE'2020-06-07','qpaper2.docx','TMP002');

INSERT INTO EXAM VALUES('EXM003','CHEM',DATE'2020-06-05','qpaper3.docx','PMT005');

INSERT INTO EXAM VALUES('EXM004','BENG',DATE'2020-06-04','qpaper4.docx','TMP004');

INSERT INTO EXAM VALUES('EXM005','CHML',DATE'2020-06-03','qpaper5.docx','PMT005');

commit;
```

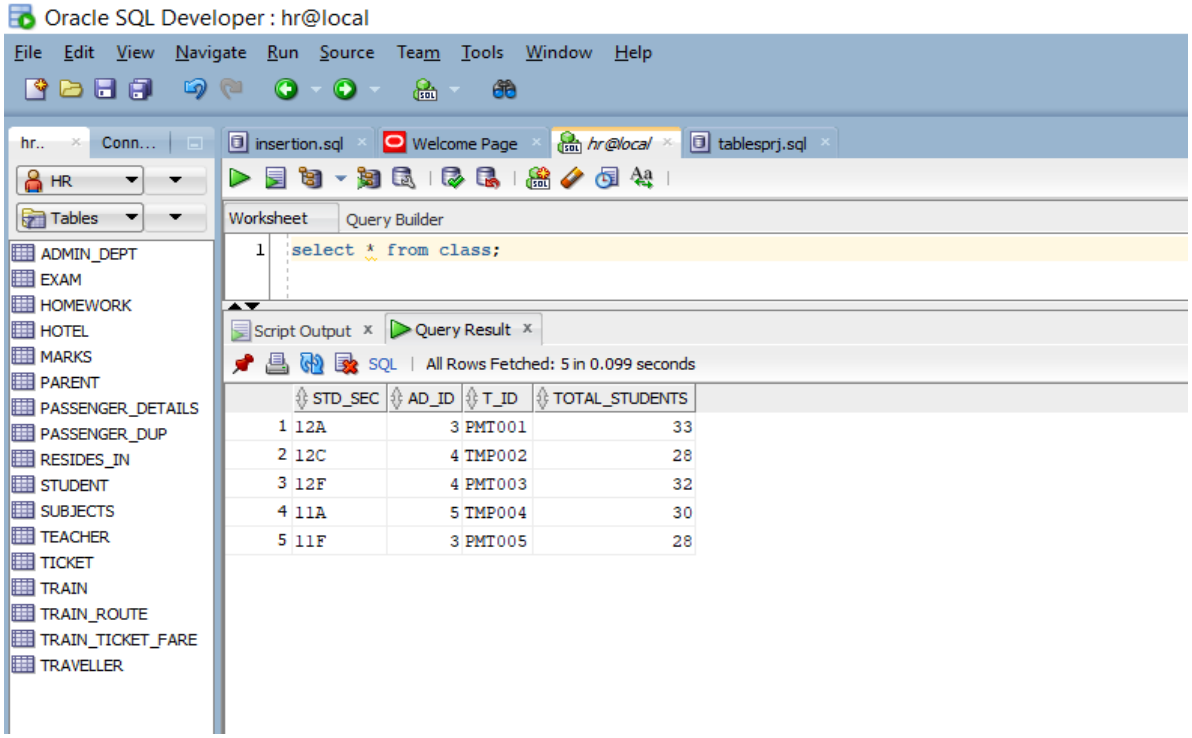
OUTPUT



'Class'

```
INSERT INTO CLASS VALUES('12A',3,'PMT001',33);
INSERT INTO CLASS VALUES('12C',4,'TMP002',28);
INSERT INTO CLASS VALUES('12F',4,'PMT003',32);
INSERT INTO CLASS VALUES('11A',5,'TMP004',30);
INSERT INTO CLASS VALUES('11F',3,'PMT005',28);
Commit;
```

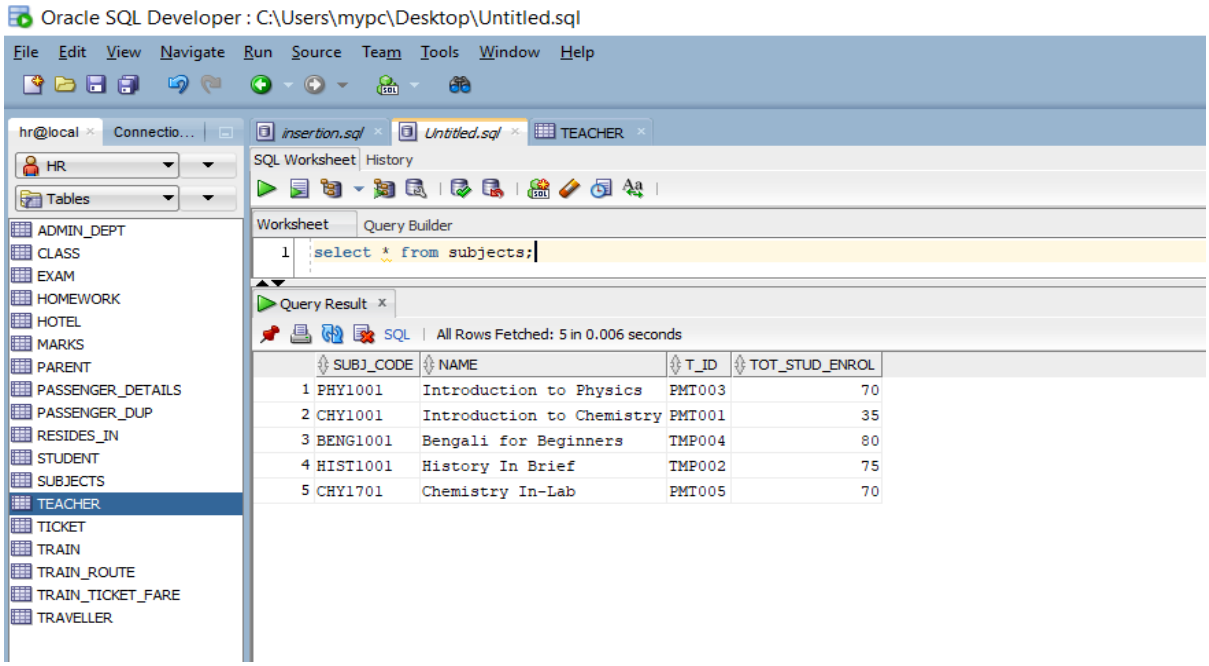
OUTPUT



'Subjects'

```
INSERT INTO SUBJECTS VALUES('PHY1001','Introduction to Physics','PMT003',70);
INSERT INTO SUBJECTS VALUES('CHY1001','Introduction to Chemistry','PMT001',35);
INSERT INTO SUBJECTS VALUES('BENG1001','Bengali for Beginners','TMP004',80);
INSERT INTO SUBJECTS VALUES('HIST1001','History In Brief','TMP002',75);
INSERT INTO SUBJECTS VALUES('CHY1701','Chemistry In-Lab','PMT005',70);
Commit;
```

OUTPUT



'Marks'

```
INSERT INTO MARKS VALUES('SL001',98,100,'PHYSICS','Sahasra','PMT003',14);
INSERT INTO MARKS VALUES('SL0001',97,100,'PHYSICS','Sree','PMT003',26);
INSERT INTO MARKS VALUES('SL003',94,100,'PHYSICS','Uday','PMT003',12);
INSERT INTO MARKS VALUES('SL004',92,100,'CHEMISTRY','Sahasra','PMT001',14);
INSERT INTO MARKS VALUES('SL005',90,100,'CHEMISTRY','Sree','PMT001',26);
commit;
```

OUTPUT