

SCHOOL MANAGEMENT SYSTEM

Views:

View-1). View which shows what all students are taught by teachers and what subjects these teachers are teaching. By this view parents can check which teachers are teaching their children and what subject do they teach.

```
CREATE view students_under_teacher AS
SELECT r1.efname,
       r1.elname,
       r1.qualification,
       r1.experience,
       r1.class_id,
       r1.student_id,
       r1.sfname,
       r1.slname,
       subject.sub_name
FROM school_management.subject
JOIN ( SELECT employee.fname AS efname,
              employee.lname AS elname,
              employee.qualification,
              employee.experience,
              r.class_id,
              r.student_id,
              r.fname AS sfname,
              r.lname AS slname,
              r.subject_id
FROM school_management.employee
JOIN ( SELECT student.class_id,
              student.student_id,
              student.fname,
              student.lname,
              teaches.teacher_id,
              teaches.subject_id
FROM school_management.student
      JOIN school_management.teaches USING (class_id)) r ON
employee.teacher_id = r.teacher_id) r1 USING (subject_id);
```

Output:

Dashboard Properties SQL Statistics Dependencies Dependents school/postgres@PostgreSQL 13 *

school/postgres@PostgreSQL 13

Query Editor Query History

```
1 SELECT * FROM students_under_teacher;
```

Data Output Explain Messages Notifications

	efname character varying (20)	ename character varying (20)	qualification character varying (30)	experience integer	class_id character varying (9)	student_id numeric (12)	sfname character varying (20)	sname character varying (20)	sub_name character varying
1	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100018	Mihir	Raval	Mathematics
2	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100017	Mihir	Akbari	Mathematics
3	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100016	Krish	Gudhka	Mathematics
4	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100015	Krish	Pandya	Mathematics
5	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100014	Karan	Gosai	Mathematics
6	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100013	Daksh	Rathod	Mathematics
7	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100012	Aayush	Chavda	Mathematics
8	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100011	Rohan	Limbasiya	Mathematics
9	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100010	Krishna	Modi	Mathematics
10	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100009	Krishna	Mendapara	Mathematics
11	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100008	Dhyey	Makawana	Mathematics
12	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100007	Harsh	Pithadiya	Mathematics
13	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100006	Gaurang	Limbasiya	Mathematics
14	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100005	Zarna	Mungra	Mathematics
15	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100004	Bhavaya	Josh	Mathematics
16	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100003	Vishwa	Sonagar	Mathematics
17	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100002	Arjun	Sanariya	Mathematics
18	Nalin	Kumar	B.Sc Mathematics, B.Ed		3 20210110A	202101100001	Aditi	Mankad	Mathematics
19	Madhukant	Sharma	B.Sc Science, B.Ed		5 20210110A	202101100018	Mihir	Raval	Science
20	Madhukant	Sharma	B.Sc Science, B.Ed		5 20210110A	202101100017	Mihir	Akbari	Science
21	Madhukant	Sharma	B.Sc Science, B.Ed		5 20210110A	202101100016	Krish	Gudhka	Science
22	Madhukant	Sharma	B.Sc Science, B.Ed		5 20210110A	202101100015	Krish	Pandya	Science
23	Madhukant	Sharma	B.Sc Science, B.Ed		5 20210110A	202101100014	Karan	Gosai	Science
24	Madhukant	Sharma	B.Sc Science, B.Ed		5 20210110A	202101100013	Daksh	Rathod	Science

View-2).View which shows marks obtained by students in some examination of some subject. This view can be used by manager to check what all exams of different subjects are taken and what marks students scored in it.

CREATE VIEW student_exams as

SELECT student_id,fname || ' ' || lname as

sname,exam_id,subject_id,sub_name,date_exam,total_marks,mark_scored FROM student

NATURAL JOIN

(SELECT * FROM result_exam NATURAL JOIN (SELECT * FROM examination

NATURAL JOIN subject) as R) as R1

Output:

school/postgres@PostgreSQL 13

Query Editor Query History Scratch Pad

```
1 SELECT * FROM student_exams;
```

Data Output Explain Messages Notifications

	student_id numeric (12)	sname text	exam_id character varying (10)	subject_id character varying (4)	sub_name character varying (20)	date_exam date	total_marks integer	mark_scored integer
1	202101100001	Aditi Mank...	2021100201	1002	Science	2021-06-30	20	20
2	202101100001	Aditi Mank...	2021100102	1001	Mathematics	2021-08-15	50	20
3	202101100001	Aditi Mank...	2021100101	1001	Mathematics	2021-06-30	25	20
4	202101100002	Arjun Sana...	2021100201	1002	Science	2021-06-30	20	7
5	202101100002	Arjun Sana...	2021100102	1001	Mathematics	2021-08-15	50	35
6	202101100002	Arjun Sana...	2021100101	1001	Mathematics	2021-06-30	25	16
7	202101100003	Vishwa So...	2021100201	1002	Science	2021-06-30	20	13
8	202101100003	Vishwa So...	2021100102	1001	Mathematics	2021-08-15	50	40
9	202101100003	Vishwa So...	2021100101	1001	Mathematics	2021-06-30	25	12
0	202101100004	Bhavaya J...	2021100201	1002	Science	2021-06-30	20	15
1	202101100004	Bhavaya J...	2021100102	1001	Mathematics	2021-08-15	50	45
2	202101100004	Bhavaya J...	2021100101	1001	Mathematics	2021-06-30	25	22
3	202101100005	Zarna Mun...	2021100201	1002	Science	2021-06-30	20	7
4	202101100005	Zarna Mun...	2021100102	1001	Mathematics	2021-08-15	50	32

View-3). View which shows marks obtained by students in some examination of some subject and maximum, minimum, average marks of that examination. This view can be used by manager to check what all exams of different subjects are taken and what is the performance of students. This view can also be used by parents to check where their child lies and what progress is required. This view can also be used by teachers to each child's weak points.

```
CREATE VIEW student_mark_comparison as
SELECT * FROM student_exams NATURAL JOIN
(SELECT exam_id,MIN(mark_scored),MAX(mark_scored),round(AVG(mark_scored),2)
from student_exams GROUP BY exam_id) as R ;
```

Output:

exam_id	student_id	sname	subject_id	sub_name	date_exam	total_marks	mark_scored	min	max	round
1	202101100001	Aditi Mank...	1001	Mathematics	2021-06-30	25	20	10	25	18.22
2	202101100002	Arjun Sana...	1001	Mathematics	2021-06-30	25	16	10	25	18.22
3	202101100003	Vishwa So...	1001	Mathematics	2021-06-30	25	12	10	25	18.22
4	202101100004	Bhavaya J...	1001	Mathematics	2021-06-30	25	22	10	25	18.22
5	202101100005	Zarna Mun...	1001	Mathematics	2021-06-30	25	25	10	25	18.22
6	202101100006	Gaurang Li...	1001	Mathematics	2021-06-30	25	12	10	25	18.22
7	202101100007	Harsh Pith...	1001	Mathematics	2021-06-30	25	15	10	25	18.22
8	202101100008	Dhyey Mak...	1001	Mathematics	2021-06-30	25	20	10	25	18.22
9	202101100009	Krishna Me...	1001	Mathematics	2021-06-30	25	23	10	25	18.22
10	202101100010	Krishna Modi	1001	Mathematics	2021-06-30	25	25	10	25	18.22
11	202101100011	Rohan Lim...	1001	Mathematics	2021-06-30	25	17	10	25	18.22
12	202101100012	Aayush Ch...	1001	Mathematics	2021-06-30	25	19	10	25	18.22
13	202101100013	Daksh Rat...	1001	Mathematics	2021-06-30	25	20	10	25	18.22
14	202101100014	Karan Gosai	1001	Mathematics	2021-06-30	25	17	10	25	18.22
15	202101100015	Krish Pand...	1001	Mathematics	2021-06-30	25	10	10	25	18.22
16	202101100016	Krish Gudh...	1001	Mathematics	2021-06-30	25	22	10	25	18.22
17	202101100017	Mihir Akbari	1001	Mathematics	2021-06-30	25	12	10	25	18.22
18	202101100018	Mihir Raval	1001	Mathematics	2021-06-30	25	21	10	25	18.22
19	202101100020	Aditi Mank...	1001	Mathematics	2021-08-15	50	20	20	49	35.06
20	202101100002	Arjun Sana...	1001	Mathematics	2021-08-15	50	35	20	49	35.06
21	202101100003	Vishwa So...	1001	Mathematics	2021-08-15	50	40	20	49	35.06
22	202101100004	Bhavaya J...	1001	Mathematics	2021-08-15	50	45	20	49	35.06
23	202101100005	Zarna Mun...	1001	Mathematics	2021-08-15	50	32	20	49	35.06
24	202101100006	Gaurang Li...	1001	Mathematics	2021-08-15	50	49	20	49	35.06

View-4). View which shows examinations taken for different subjects by teachers . This view can be used by manager to check what all exams of different subjects are taken and what all subjects a teacher teaches in which class.

```
CREATE VIEW teacher_subjects_exams as
SELECT teacher_id,pan_no,fname || ' ' || lname as
sname,subject_id,class_id,exam_id,sub_name,date_exam FROM employee NATURAL
JOIN (SELECT * from teaches NATURAL JOIN
(SELECT subject.subject_id,sub_name,exam_id,date_exam FROM subject LEFT OUTER
JOIN examination ON (subject.subject_id=examination.subject_id)) as R) as R1
```

Output:

Dashboard Properties SQL Statistics Dependencies Dependents school/postgres@PostgreSQL 13 *

View-5). View which shows instruments which are not in good condition. Through this view manager can enlist list of instruments or chemicals which are either to be repaired or new ones are to be brought for a particular library. Also one can find the name of incharge who is handling all these records.

```
CREATE VIEW manage_labs as
SELECT incharge_id, fname||' '||lname as
iname, lab_code, instrument_id, lab_type, instrument_name, chemical_name, condition_status
FROM employee NATURAL JOIN
(SELECT * FROM lab NATURAL JOIN (SELECT * FROM instruments
WHERE(condition_status!='Good'))) as R) as R1
```

Output:

Help

Dashboard

Properties

SQL

Statistics

Dependencies

Dependents

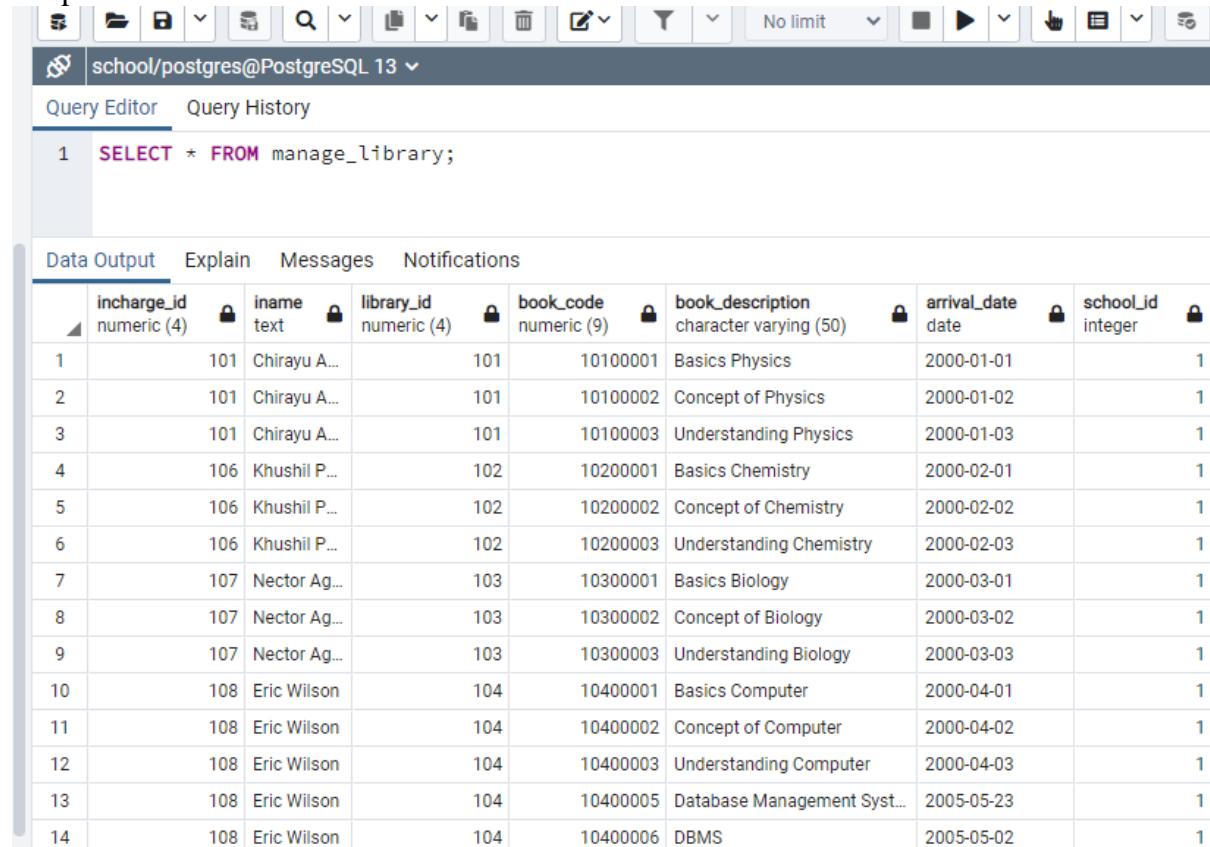
school/postgres@PostgreSQL 13 *

<

View-6). View which shows list of books present in different school libraries and their arrival dates. By this view library incharge or manager can make list of books which are not available in school library and are required.

```
CREATE VIEW manage_library as
SELECT R1.incharge_id,fname||' '||lname as
iname,library_id,book_code,book_description,arrival_date,R1.school_id FROM employee
JOIN
(SELECT * FROM library1 NATURAL JOIN books) as R1 ON
(employee.incharge_id=R1.incharge_id)
```

Output:



	incharge_id numeric (4)	iname text	library_id numeric (4)	book_code numeric (9)	book_description character varying (50)	arrival_date date	school_id integer
1	101	Chirayu A...	101	10100001	Basics Physics	2000-01-01	1
2	101	Chirayu A...	101	10100002	Concept of Physics	2000-01-02	1
3	101	Chirayu A...	101	10100003	Understanding Physics	2000-01-03	1
4	106	Khushil P...	102	10200001	Basics Chemistry	2000-02-01	1
5	106	Khushil P...	102	10200002	Concept of Chemistry	2000-02-02	1
6	106	Khushil P...	102	10200003	Understanding Chemistry	2000-02-03	1
7	107	Nector Ag...	103	10300001	Basics Biology	2000-03-01	1
8	107	Nector Ag...	103	10300002	Concept of Biology	2000-03-02	1
9	107	Nector Ag...	103	10300003	Understanding Biology	2000-03-03	1
10	108	Eric Wilson	104	10400001	Basics Computer	2000-04-01	1
11	108	Eric Wilson	104	10400002	Concept of Computer	2000-04-02	1
12	108	Eric Wilson	104	10400003	Understanding Computer	2000-04-03	1
13	108	Eric Wilson	104	10400005	Database Management Syst...	2005-05-23	1
14	108	Eric Wilson	104	10400006	DBMS	2005-05-02	1

View-7). View which shows student contact details and parent details of that student. This view can be used by manager to get contact details of a student.

```
CREATE VIEW student_parent_detail as
SELECT student_id,fname||' '||lname as sname,student.contactno as
s_contact,class_id,totalfees,feespaid,parent,pname,contact_no FROM student JOIN
(SELECT parent,student,fname ||' '|| lname as pname,contact_no FROM relation JOIN
(SELECT * FROM parent JOIN parent_contact ON(pan_no=parent_id)) as R ON
(parent=parent_id)) as R1 ON (student_id=student);
```

Output:

help

Dashboard Properties SQL Statistics Dependencies Dependents school/postgres@PostgreSQL 13 *

Query Editor Query History

```
1 SELECT * FROM student_parent_detail;
```

Data Output Explain Messages Notifications

	student_id numeric (12)	sname text	s_contact numeric (10)	class_id character varying (9)	totalfees numeric (6)	feespaid numeric (6)	parent character varying (10)	pname text	contact_no numeric (10)
1	202101100001	Aditi Mank...	9276215690	20210110A	50000	25000	1111111110	Usha Patel	9999999999
2	202101100002	Arjun Sana...	9687123498	20210110A	45000	35000	1111111110	Usha Patel	9999999999
3	202101100001	Aditi Mank...	9276215690	20210110A	50000	25000	1111111111	Nilam Cha...	9999999998
4	202101100002	Arjun Sana...	9687123498	20210110A	45000	35000	1111111111	Nilam Cha...	9999999998
5	202101100003	Vishwa So...	9607183498	20210110A	50000	35000	1111111112	Bharat Patel	9999999997
6	202101100004	Bhavaya J...	9481123498	20210110A	50000	50000	1111111113	Gopal Cho...	9999999996
7	202101100005	Zarna Mun...	9684283498	20210110A	45000	5000	1111111114	Ramesh S...	9999999995
8	202101100006	Gaurang Li...	9687123702	20210110A	45000	0	1111111115	Nita Patel	9999999994
9	202101100007	Harsh Pitth...	9687159198	20210110A	0	0	1111111116	Om Patel	9999999993
10	202101100008	Dhyey Mak...	9457183498	20210110A	50000	15000	1111111117	Abhi Chau...	9999999992
11	202101100009	Krishna Me...	9481123498	20210110A	50000	50000	1111111118	Raman Patel	9999999991
12	202101100010	Krishna Modi	9684283498	20210110A	45000	10000	1111111119	Surya Cho...	9999999990
13	202101100011	Rohan Lim...	9687123702	20210110A	45000	0	1111111119	Surya Cho...	9999999990
14	202101100012	Aayush Ch...	9687123702	20210110A	50000	5000	1111111120	Geeta Shah	9999999990

Functions

Function).Find information of employees whose salary is more than 30000 by creating a function.

CREATE OR REPLACE function more_salary(sal integer)

RETURNS SETOF employee AS \$BODY\$

DECLARE

e employee%rowtype;

BEGIN

FOR e IN SELECT * FROM employee

LOOP

IF e.salary > sal THEN -- could be more complex filter

RETURN NEXT e;

-- return basically appends a row to result-set, and builds it for return

END IF;

END LOOP;

RAISE NOTICE '%',e;

RETURN; -- this is actual return

END

\$BODY\$ LANGUAGE plpgsql;

SELECT * FROM more_salary(30000);

Output:

Dashboard Properties SQL Statistics Dependencies Dependencies school/postgres@PostgreSQL 13 *

Query Editor Query History

```
1 SELECT * FROM more_salary(30000);
```

Data Output Explain Messages Notifications

	par_no character varying (10)	fname character varying (20)	lname character varying (20)	designation character varying (50)	address character varying (150)	contact_no numeric (10)	salary integer
1	0000000008	Nalin	Kumar	Teacher	Ahemdabad	7777252228	35
2	0000000009	Madhukant	Sharma	Teacher	Ahemdabad	7777252308	35
3	0000000010	Jenish	Patel	Teacher	Ahemdabad	9910254118	35
4	0000000006	Ketan	Shah	Principal	Ahemdabad	7887333118	33

Data Output Explain Messages Notifications

	contact_no numeric (10)	salary integer	dob date	gender character (1)	school_id integer	teacher_id numeric (5)	qualification character varying (30)	experience integer	incharge_id numeric (4)
	7777252228	35000	1990-12-...	M	1	1001	B.Sc Mathematics, B.Ed	3	[null]
	7777252308	35000	1890-03-...	M	1	1002	B.Sc Science, B.Ed	5	[null]
	9910254118	35000	1890-03-...	M	1	1003	B.Ed	5	[null]
	7887333118	33000	1989-02-...	M	1	[null] [null]		[null]	[null]

Output:

school/postgres@PostgreSQL 13

Query Editor Query History

```

1 --select increment_sal(0.1, '000000006'::VARCHAR(10));
2 SELECT * FROM employee;

```

Data Output Explain Messages Notifications

	pan_no [PK] character varying (10)	fname character varying (20)	lname character varying (20)	designation character varying (50)	address character varying (150)	contact_no numeric (10)	salary integer	dob date
1	0000000001	Raju	Choksi	Security Guard	Ahemdabad	7887254118	9000	1991
2	0000000002	Ram	Choksi	Clerk	Ahemdabad	9997254118	11000	1987
3	0000000003	Shyam	Choksi	Peoun	Ahemdabad	7909254118	5000	1997
4	0000000004	Geeta	Patel	Receptionist	Ahemdabad	7882754300	9000	1991
5	0000000005	Kalgi	Shukla	Accountant	Ahemdabad	9999254118	10000	1991
6	0000000007	Sangeeta	Choksi	Vice Principal	Ahemdabad	7777254118	20000	1891
7	0000000008	Nalin	Kumar	Teacher	Ahemdabad	7777252228	35000	1991
8	0000000009	Madhukant	Sharma	Teacher	Ahemdabad	7777252308	35000	1891
9	0000000010	Jenish	Patel	Teacher	Ahemdabad	9910254118	35000	1891
10	0000000011	Chaitri	Gudhka	Teacher	Ahemdabad	8888254118	30000	1891
11	0000000012	Vraj	Chaudhari	Teacher	Ahemdabad	8888254118	30000	1891
12	0000000013	Pathik	Patel	Teacher	Ahemdabad	7777754118	25000	1891
13	0000000014	Prayag	Patel	Teacher	Ahemdabad	8877754118	25000	1891
14	0000000015	Manan	Parikh	Teacher	Ahemdabad	7777766118	28000	1891
15	0000000016	Malhar	Nimavat	Teacher	Ahemdabad	7777754998	28000	1891
16	0000000017	Chirayu	Agrawal	Incharge	Ahemdabad	7667754118	20000	1891
17	0000000018	Devarshi	Joshi	Incharge	Ahemdabad	7669954118	18000	1891
18	0000000020	Ishan	Raval	Incharge	Ahemdabad	7889954118	18000	1891

After calling Function:

school/postgres@PostgreSQL 13

Query Editor Query History

```

1 select increment_sal(0.1, '0000000001'::VARCHAR(10));
2 SELECT * FROM employee;

```

Data Output Explain Messages Notifications

	pan_no [PK] character varying (10)	fname character varying (20)	lname character varying (20)	designation character varying (50)	address character varying (150)	contact_no numeric (10)	salary integer	dob date
10	0000000012	Vraj	Chaudhari	Teacher	Ahemdabad	8888254118	30000	1891
11	0000000013	Pathik	Patel	Teacher	Ahemdabad	7777754118	25000	1891
12	0000000014	Prayag	Patel	Teacher	Ahemdabad	8877754118	25000	1891
13	0000000015	Manan	Parikh	Teacher	Ahemdabad	7777766118	28000	1891
14	0000000016	Malhar	Nimavat	Teacher	Ahemdabad	7777754998	28000	1891
15	0000000017	Chirayu	Agrawal	Incharge	Ahemdabad	7667754118	20000	1891
16	0000000018	Devarshi	Joshi	Incharge	Ahemdabad	7669954118	18000	1891
17	0000000020	Ishan	Raval	Incharge	Ahemdabad	7889954118	18000	1891
18	0000000021	Shreyansh	Kunjera	Incharge	Ahemdabad	7660054118	18000	1891
19	0000000022	Khushil	Patel	Incharge	Ahemdabad	7667754000	20000	1891
20	0000000023	Nector	Agrawal	Incharge	Ahemdabad	7667014118	20000	1891
21	0000000024	Eric	Wilson	Incharge	Ahemdabad	7611754118	18000	1891
22	0000000019	Soudamini	Kidambi	Incharge	Ahemdabad	7611754118	18000	1891
23	0000000025	Ramnath	Kohli	Teacher	Ahemdabad	7887254118	30000	1891
24	0000000026	Maresh	Mishra	Teacher	Ahemdabad	7897254118	30000	1891
25	0000000027	Pooja	Kant	Teacher	Ahemdabad	7887254118	30000	1891
26	0000000006	Ketan	Shah	Principal	Ahemdabad	7887333118	33000	1891
27	0000000001	Raju	Choksi	Security Guard	Ahemdabad	7887254118	9900	1891

Triggers

Trigger) To derive age attribute of employee from dob attribute whenever record inserted or updated.

create or replace function add_age_employee()

returns trigger as \$emp_audit\$

begin

New.age=(CURRENT_DATE-new.dob)/365;

```

return new;
end;
$emp_audit$ language plpgsql;

```

```

create trigger add_age
before insert or update on employee
for each row execute procedure add_age_employee();

```

Output:

school/postgres@PostgreSQL 13										
Query Editor Query History										
<pre> 1 SET SEARCH_PATH TO school_management; 2 INSERT INTO employee(3 pan_no, fname, lname, designation, address, contact_no, salary, dob, gender, school_id) 4 VALUES ('0000000029', 'Anil', 'Singh', 'Assitant', 'Ahemdabad', 9871243210, 25000, '1995-3-21', 'M', 1); 5 SELECT * FROM employee </pre>										
Data Output Explain Messages Notifications										
	pan_no [PK] character varying (10)	fname character varying (20)	lname character varying (20)	designation character varying (50)	address character varying (150)	contact_no numeric (10)	salary integer	dob date	gender character (1)	school_id integer
17	0000000020	Ishan	Raval	Incharge	Rajnagar	7889954118	18000			
18	0000000021	Shreyansh	Kunjera	Incharge	Rajnagar	7660054118	18000			
19	0000000022	Khushil	Patel	Incharge	Rajnagar	7667754000	20000			
20	0000000023	Nector	Agrawal	Incharge	Rajnagar	7667014118	20000			
21	0000000024	Eric	Wilson	Incharge	Rajnagar	7611754118	18000			
22	0000000019	Soudamini	Kidambi	Incharge	Rajnagar	7611754118	18000			
23	0000000025	Ramnath	Kohli	Teacher	Rajnagar	7887254118	30000			
24	0000000026	Mahesh	Mishra	Teacher	Rajnagar	7897254118	30000			
25	0000000027	Pooja	Kant	Teacher	Rajnagar	7887254118	30000			
26	0000000006	Ketan	Shah	Principal	Rajnagar	7887333118	33000			
27	0000000001	Raju	Choksi	Security Guard	Rajnagar	7887254118	9000			
28	0000000028	Malini	Singh	Assitant	Ahemdabad	9876543210	20000			
29	0000000029	Anil	Singh	Assitant	Ahemdabad	9871243210	25000			

school/postgres@PostgreSQL 13 *										
Query Editor Query History										
<pre> 1 SET SEARCH_PATH TO school_management; 2 INSERT INTO employee(3 pan_no, fname, lname, designation, address, contact_no, salary, dob, gender, school_id) 4 VALUES ('0000000029', 'Anil', 'Singh', 'Assitant', 'Ahemdabad', 9871243210, 25000, '1995-3-21', 'M', 1); 5 SELECT * FROM employee </pre>										
Data Output Explain Messages Notifications										
	contact_no numeric (10)	salary integer	dob date	gender character (1)	school_id integer	teacher_id numeric (5)	qualification character varying (30)	experience integer	incharge_id numeric (4)	age integer
	7889954118	18000	1890-09...	M		[null]	[null]	[null]	104	131
	7660054118	18000	1890-01...	M	1	[null]	[null]	[null]	105	131
	7667754000	20000	1990-01...	M	1	[null]	[null]	[null]	106	31
	7667014118	20000	1990-08...	M	1	[null]	[null]	[null]	107	31
	7611754118	18000	1990-08...	M	1	[null]	[null]	[null]	108	31
	7611754118	18000	1990-08...	M	1	[null]	[null]	[null]	103	31
	7887254118	30000	1990-04...	M	1	1010	B. Ed, M.Ed	15	[null]	31
	7897254118	30000	1990-05...	M	1	1011	B.Sc	7	[null]	31
	7887254118	30000	1990-04...	F	1	1012	B.Sc, M.Sc	10	[null]	31
	7887333118	33000	1989-02...	M	1	[null]	[null]	[null]	[null]	32
	7887254118	9000	1990-04...	M	1	[null]	[null]	[null]	[null]	31
	9876543210	20000	1998-03...	F	1	[null]	[null]	[null]	[null]	23
	9871243210	25000	1995-03...	M	1	[null]	[null]	[null]	[null]	26