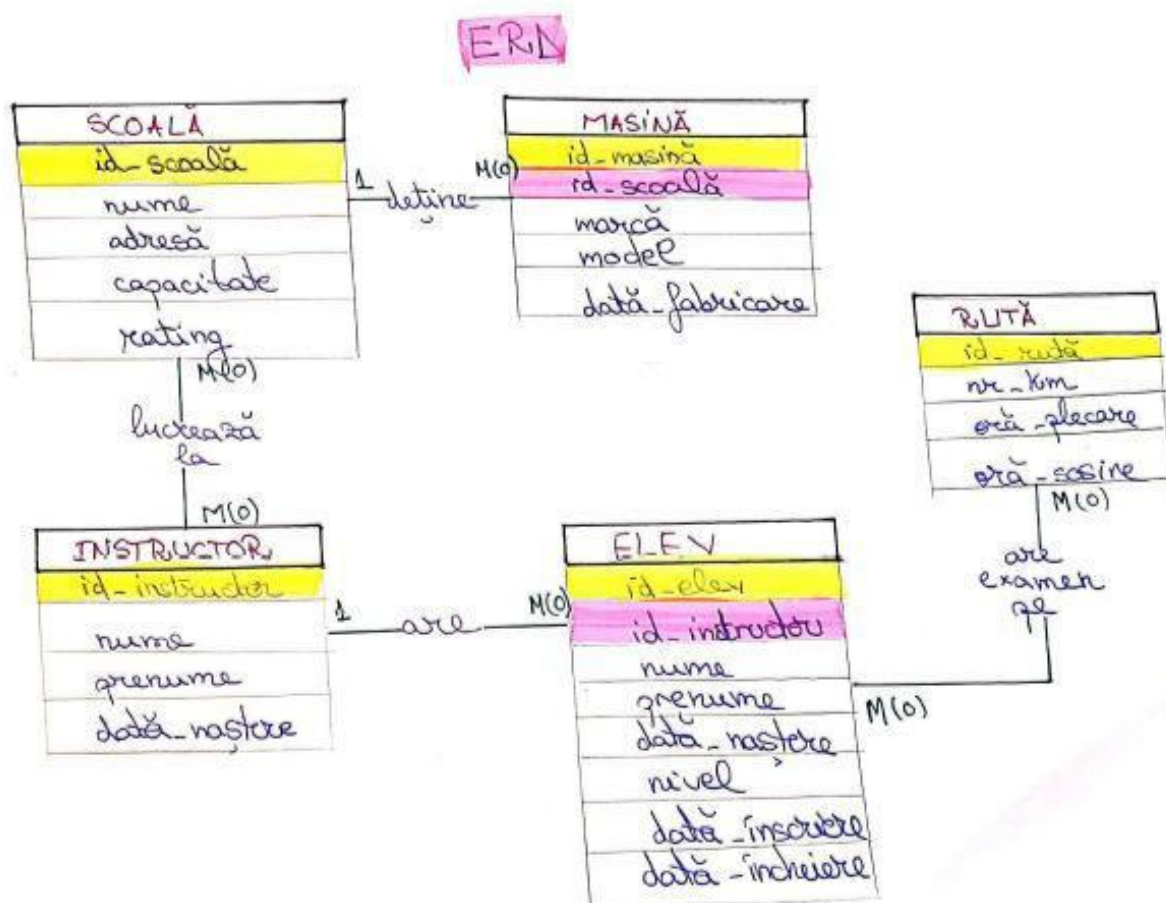


SCOALĂ DE ȘOFERI

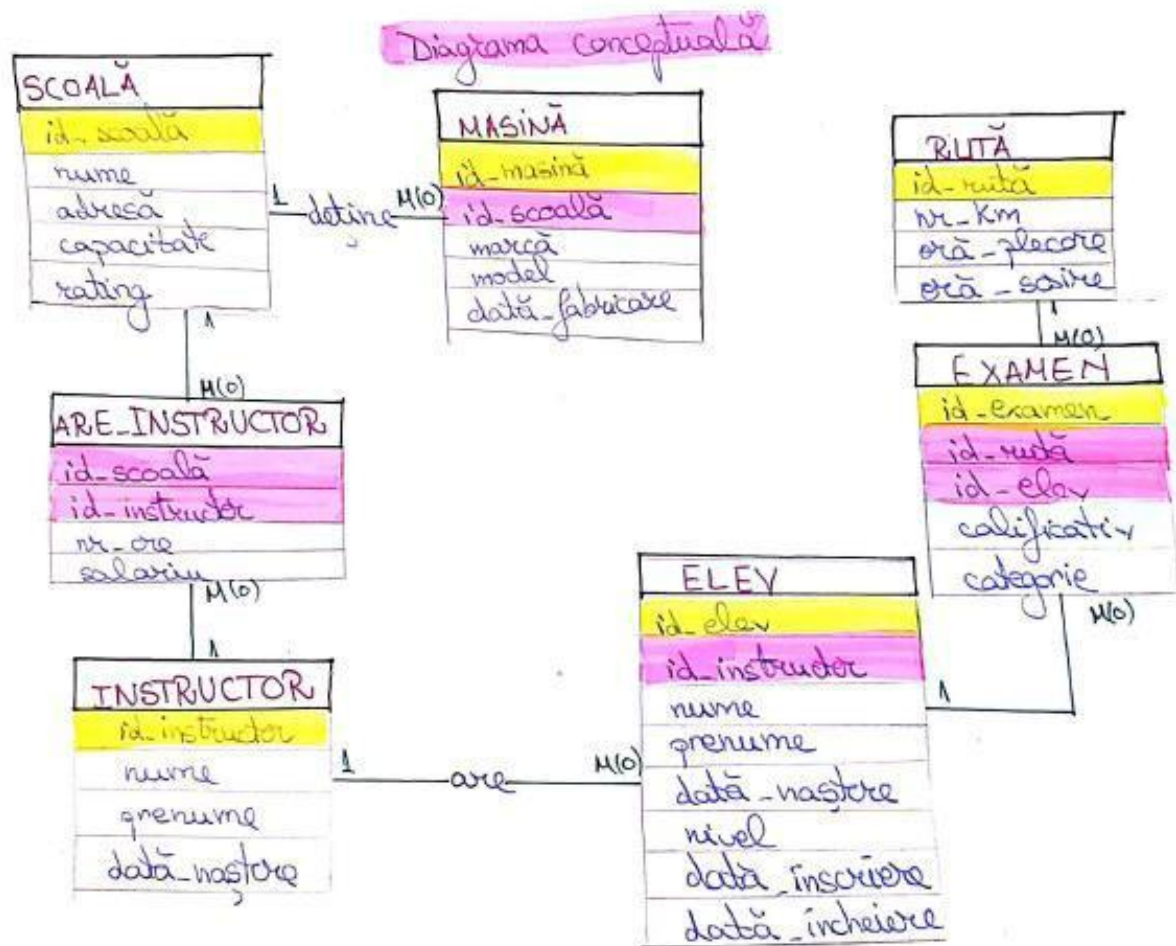
- Scurtă prezentare a bazei de date

Aceasta este o bază de date pentru gestionarea școlilor de șoferi, conținând date despre acestea, precum și despre elevi, instructori și examene. De asemenea aceasta păstrează date despre mașinile deținute de fiecare școală, dar și despre rute predefinite pentru examenul practice.

- Diagrama entitate-relație



- Diagrama conceptuală



- Definirea tabelelor in Oracle si implementarea constrângerilor de integritate

The screenshot displays the Oracle SQL Developer environment. On the left, the 'Connections' pane shows a connection to 'exam52' and a tree view of the 'grupa44proiect' schema. The 'Tables (Filtered)' list includes tables such as ADMISSION_CPR, APLICATII_ADO, ARE_INSTRUCTOR_BPA, ARTIST_OTO, ATTENDS_VST, BAC_CPR, BACALAUREAT_ADO, BILETE_EMU, CANDIDATE_CPR, CANDIDATI_ADO, CATEGORY_OTO, CLIENT_OTO, CLIENTI_EMU, COURSES_VST, CURSE_EMU, DEPARTMENTS_VST, DISTRIBUIE_ABU, DISTRIBUITORI_ABU, ELEV_BPA, EVENT_OTO, EXAMEN_ADMITERE_ADO, EXAMEN_BPA, EXAMS_VST, FACULTATI_ADO, FACULTY_CPR, INSTRUCTOR_BPA, LOCATII_ABU, LOCATION_CPR, LOCATION_OTO, MAGAZINE_ABU, MASINA_BPA, PARTICIPATE_OTO, PRODUSE_ABU, RUTA_BPA, RUTE_EMU, SCOALA_BPA, SOFERI_EMU, SPECIALIZARI_ADO, SPECIALIZATION_CPR, STOCURI_ABU, STUDENTS_VST, TEACHERS_VST, TEACHES_VST, TICKET_OTO, and VEHICULE_EMU.

The main window shows a SQL Worksheet with the following code:

```
1 CREATE TABLE "SCOALA_BPA" (  
2   "ID_SCOALA" INT NOT NULL,  
3   "NUME" VARCHAR2(100) NOT NULL,  
4   "ADRESA" VARCHAR2(200) NOT NULL,  
5   "CAPACITATE" INT NOT NULL,  
6   "RATING" INT NOT NULL,  
7   constraint SCOALA_BPA_PK PRIMARY KEY ("ID_SCOALA"));  
8  
9 CREATE sequence "SCOALA_BPA_ID_SCOALA_SEQ";  
10  
11 CREATE trigger "BI_SCOALA_BPA_ID_SCOALA"  
12   before insert on "SCOALA_BPA"  
13   for each row  
14   begin  
15     select "SCOALA_BPA_ID_SCOALA_SEQ".nextval into :NEW."ID_SCOALA" from dual;  
16   end;  
17  
18  
19 CREATE TABLE "RUTA_BPA" (  
20   "ID_RUTA" INT NOT NULL,  
21   "NR_KM" INT NOT NULL,  
22   "ORA_PLECARA" TIMESTAMP WITH LOCAL TIME ZONE NOT NULL,  
23   "ORA_SOSIRE" TIMESTAMP WITH LOCAL TIME ZONE NOT NULL,  
24   constraint RUTA_BPA_PK PRIMARY KEY ("ID_RUTA"));  
25  
26 CREATE sequence "RUTA_BPA_ID_RUTA_SEQ";  
27  
28 CREATE trigger "BI_RUTA_BPA_ID_RUTA"  
29   before insert on "RUTA_BPA"  
30   for each row  
31   begin  
32     select "RUTA_BPA_ID_RUTA_SEQ".nextval into :NEW."ID_RUTA" from dual;  
33   end;  
34  
35  
36 CREATE TABLE "INSTRUCTOR_BPA" (  
37   "ID_INSTRUCTOR" INT NOT NULL,  
38   "NUME" VARCHAR2(50) NOT NULL,  
39   "PRENUME" VARCHAR2(50) NOT NULL,  
40   "DATA_NASTERE" DATE NOT NULL,  
41   constraint INSTRUCTOR_BPA_PK PRIMARY KEY ("ID_INSTRUCTOR"));  
42  
43 CREATE sequence "INST_BPA_ID_INST_SEQ";  
44  
45 CREATE trigger "BI_INST_BPA_ID_INST"  
46   before insert on "INSTRUCTOR_BPA"  
47   for each row  
48   begin  
49     select "INST_BPA_ID_INST_SEQ".nextval into :NEW."ID_INSTRUCTOR" from dual;  
50   end;  
51  
52  
53 CREATE TABLE "ELEV_BPA" (  
54   "ID_ELEV" INT NOT NULL,  
55   "ID_INSTRUCTOR" INT NOT NULL,  
56   "NUME" VARCHAR2(50) NOT NULL,  
57   "PRENUME" VARCHAR2(50) NOT NULL,  
58   "DATA_NASTERE" DATE NOT NULL,  
59   "NIVEL" VARCHAR2(20) NOT NULL,  
60   "DATA_INSCRIERE" DATE NOT NULL,  
61   "DATA_INCHEIERE" DATE NOT NULL,  
62   constraint ELEV_BPA_PK PRIMARY KEY ("ID_ELEV"));  
63  
64 CREATE sequence "ELEV_BPA_ID_ELEV_SEQ";  
65  
66 CREATE trigger "BI_ELEV_BPA_ID_ELEV"  
67   before insert on "ELEV_BPA"  
68   for each row  
69   begin  
70     select "ELEV_BPA_ID_ELEV_SEQ".nextval into :NEW."ID_ELEV" from dual;  
71   end;
```

Tables (Filtered)	SQL Code
ADMISSION_CPR	74 CREATE TABLE "ARE_INSTRUCTOR_BPA" (
APLICATII_ADO	75 "ID_SCOALA" INT NOT NULL,
ARE_INSTRUCTOR_BPA	76 "ID_INSTRUCTOR" INT NOT NULL,
ARTIST_OTO	77 "NR_ORE" INT NOT NULL,
ATTENDS_VST	78 "SALARIU" FLOAT NOT NULL);
BAC_CPR	79
BACALAUREAT_ADO	80
BILETE_EMU	81
CANDIDATE_CPR	82 CREATE TABLE "EXAMEN_BPA" (
CANDIDATI_ADO	83 "ID_EXAMEN" INT NOT NULL,
CATEGORY_OTO	84 "ID_RUTA" INT NOT NULL,
CLIENT_OTO	85 "ID_ELEV" INT NOT NULL,
CLIENTI_EMU	86 "CALIFICATIV" VARCHAR2(50) NOT NULL,
COURSES_VST	87 "CATEGORIE" CHAR(255) NOT NULL,
CURSE_EMU	88 constraint EXAMEN_BPA_PK PRIMARY KEY ("ID_EXAMEN"));
DEPARTMENTS_VST	89
DISTRIBUIE_ABU	90 CREATE sequence "EXAMEN_BPA_ID_EXAMEN_SEQ";
DISTRIBUTORI_ABU	91
ELEV_BPA	92 CREATE trigger "BI_EXAMEN_BPA_ID_EXAMEN"
EVENT_OTO	93 before insert on "EXAMEN_BPA"
EXAMEN_ADMITERE_ADO	94 for each row
EXAMEN_BPA	95 begin
EXAMS_VST	96 select "EXAMEN_BPA_ID_EXAMEN_SEQ".nextval into :NEW."ID_EXAMEN" from dual;
FACULTATI_ADO	97 end;
FACULTY_CPR	98
INSTRUCTOR_BPA	99
LOCATII_ABU	100
LOCATION_CPR	101 CREATE TABLE "MASINA_BPA" (
LOCATION_OTO	102 "ID_MASINA" INT NOT NULL,
MAGAZINE_ABU	103 "ID_SCOALA" INT NOT NULL,
MASINA_BPA	104 "MARCA" VARCHAR2(50) NOT NULL,
PARTICIPATE_OTO	105 "MODEL" VARCHAR2(50) NOT NULL,
PRODUCE_ABU	106 "DATA_FABRICARE" DATE NOT NULL,
RUTA_BPA	107 constraint MASINA_BPA_PK PRIMARY KEY ("ID_MASINA"));
RUTE_EMU	108
SCOALA_BPA	109 CREATE sequence "MASINA_BPA_ID_MASINA_SEQ";
SOFERI_EMU	110
SPECIALIZARI_ADO	111 CREATE trigger "BI_MASINA_BPA_ID_MASINA"
SPECIALIZATION_CPR	112 before insert on "MASINA_BPA"
STOCURI_ABU	113 for each row
STUDENTS_VST	114 begin
TEACHERS_VST	115 select "MASINA_BPA_ID_MASINA_SEQ".nextval into :NEW."ID_MASINA" from dual;
	116 end;
	117
CATEGORY_OTO	121 ALTER TABLE "MASINA_BPA" ADD CONSTRAINT "MASINA_BPA_fk0" FOREIGN KEY ("ID_SCOALA")
CLIENT_OTO	122 REFERENCES "SCOALA_BPA" ("ID_SCOALA");
CLIENTI_EMU	123
COURSES_VST	124
CURSE_EMU	125 ALTER TABLE "ELEV_BPA" ADD CONSTRAINT "ELEV_BPA_fk0" FOREIGN KEY ("ID_INSTRUCTOR")
DEPARTMENTS_VST	126 REFERENCES "INSTRUCTOR_BPA" ("ID_INSTRUCTOR");
DISTRIBUIE_ABU	127
DISTRIBUTORI_ABU	128 ALTER TABLE "ARE_INSTRUCTOR_BPA" ADD CONSTRAINT "ARE_INSTRUCTOR_BPA_fk0" FOREIGN KEY ("ID_SCOALA")
ELEV_BPA	129 REFERENCES "SCOALA_BPA" ("ID_SCOALA");
EVENT_OTO	130 ALTER TABLE "ARE_INSTRUCTOR_BPA" ADD CONSTRAINT "ARE_INSTRUCTOR_BPA_fk1" FOREIGN KEY ("ID_INSTRUCTOR")
EXAMEN_ADMITERE_ADO	131 REFERENCES "INSTRUCTOR_BPA" ("ID_INSTRUCTOR");
EXAMEN_BPA	132
EXAMS_VST	133 ALTER TABLE "EXAMEN_BPA" ADD CONSTRAINT "EXAMEN_BPA_fk0" FOREIGN KEY ("ID_RUTA")
FACULTATI_ADO	134 REFERENCES "RUTA_BPA" ("ID_RUTA");
FACULTY_CPR	135 ALTER TABLE "EXAMEN_BPA" ADD CONSTRAINT "EXAMEN_BPA_fk1" FOREIGN KEY ("ID_ELEV")
INSTRUCTOR_BPA	136 REFERENCES "ELEV_BPA" ("ID_ELEV");

Pentru realizarea auto-incrementării au fost necesare crearea unui sequence si a unui trigger conform surselor următoare, valabile pentru bazele de date Oracle:

2 https://www.w3schools.com/sql/sql_autoincrement.asp

2 <http://www.java2s.com/Code/Oracle/Trigger/Usetriggertocreateautoincrementcolumn.htm>

- Adăugarea informațiilor in tabele

~ INSTRUCTOR

```
INSERT INTO instructor_bpa (nume,prenume,data_nastere)
VALUES ('Ionescu','Ionel',TO_DATE('1989-12-09','YYYY-MM-DD'));

INSERT INTO instructor_bpa (nume,prenume,data_nastere)
VALUES ('Marinescu','Monica',TO_DATE('1972-02-12','YYYY-MM-DD'));

INSERT INTO instructor_bpa (nume,prenume,data_nastere)
VALUES ('Mihnea','Aurel',TO_DATE('1977-11-29','YYYY-MM-DD'));

INSERT INTO instructor_bpa (nume,prenume,data_nastere)
VALUES ('Mateescu','Alexandru',TO_DATE('1990-01-21','YYYY-MM-DD'));

INSERT INTO instructor_bpa (nume,prenume,data_nastere)
VALUES ('Popa','Irina',TO_DATE('1982-12-29','YYYY-MM-DD'));

INSERT INTO instructor_bpa (nume,prenume,data_nastere)
VALUES ('Baniciu','Andrei',TO_DATE('1989-03-05','YYYY-MM-DD'));
```

~ RUTA

```
INSERT INTO ruta_bpa (nr_km,ora_plecare,ora_sosire)
VALUES (10, TO_TIMESTAMP ('10-Sep-20 14:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'),
        TO_TIMESTAMP ('10-Sep-20 16:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'));

INSERT INTO ruta_bpa (nr_km,ora_plecare,ora_sosire)
VALUES (15, TO_TIMESTAMP ('22-JUN-21 12:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'),
        TO_TIMESTAMP ('22-JUN-21 14:30:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'));

INSERT INTO ruta_bpa (nr_km,ora_plecare,ora_sosire)
VALUES (20, TO_TIMESTAMP ('30-AUG-18 11:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'),
        TO_TIMESTAMP ('30-AUG-18 12:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'));

INSERT INTO ruta_bpa (nr_km,ora_plecare,ora_sosire)
VALUES (13, TO_TIMESTAMP ('30-AUG-20 16:30:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'),
        TO_TIMESTAMP ('30-AUG-20 19:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'));

INSERT INTO ruta_bpa (nr_km,ora_plecare,ora_sosire)
VALUES (5, TO_TIMESTAMP ('14-Feb-19 15:50:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'),
        TO_TIMESTAMP ('14-Feb-19 16:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'));

INSERT INTO ruta_bpa (nr_km,ora_plecare,ora_sosire)
VALUES (21, TO_TIMESTAMP ('20-FEB-20 14:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'),
        TO_TIMESTAMP ('20-Feb-20 16:10:10.123000', 'DD-Mon-RR HH24:MI:SS.FF'));
```

~ *ELEV*

```
INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (2,'Gheorghe','Ionut',TO_DATE('1999-12-09','YYYY-MM-DD'),'Inceptor',TO_DATE('2020-12-09','YYYY-MM-DD'),
        TO_DATE('2021-02-09','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (2,'Ghimis','Cristian',TO_DATE('2000-02-10','YYYY-MM-DD'),'Mediu',TO_DATE('2020-02-10','YYYY-MM-DD'),
        TO_DATE('2020-04-10','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (2,'Proteasa','Vlad',TO_DATE('1999-12-09','YYYY-MM-DD'),'Avansat',TO_DATE('2019-03-05','YYYY-MM-DD'),
        TO_DATE('2019-05-05','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (1,'Barda','Monica',TO_DATE('1999-07-12','YYYY-MM-DD'),'Inceptor',TO_DATE('2018-08-25','YYYY-MM-DD'),
        TO_DATE('2018-10-25','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (5,'Barda','Monica',TO_DATE('1999-07-12','YYYY-MM-DD'),'Avansat',TO_DATE('2020-09-25','YYYY-MM-DD'),
        TO_DATE('2020-11-25','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (1,'Sima','Ion',TO_DATE('1999-06-09','YYYY-MM-DD'),'Mediu',TO_DATE('2020-12-09','YYYY-MM-DD'),
        TO_DATE('2021-02-09','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (1,'Dumitrana','Daniel',TO_DATE('1997-10-16','YYYY-MM-DD'),'Mediu',TO_DATE('2018-08-28','YYYY-MM-DD'),
        TO_DATE('2018-10-28','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (3,'Vasilescu','Dariana',TO_DATE('1998-04-19','YYYY-MM-DD'),'Inceptor',TO_DATE('2018-12-09','YYYY-MM-DD'),
        TO_DATE('2019-02-09','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (4,'Bratu','Valerian',TO_DATE('1996-04-08','YYYY-MM-DD'),'Avansat',TO_DATE('2020-04-08','YYYY-MM-DD'),
        TO_DATE('2020-06-08','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (4,'Mario','Marcel',TO_DATE('1998-07-09','YYYY-MM-DD'),'Inceptor',TO_DATE('2020-06-09','YYYY-MM-DD'),
        TO_DATE('2021-08-09','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (4,'Petre','Maria',TO_DATE('1999-03-09','YYYY-MM-DD'),'Mediu',TO_DATE('2018-07-09','YYYY-MM-DD'),
        TO_DATE('2018-09-09','YYYY-MM-DD'));

INSERT INTO elev_bpa (id_instructor,nume,prenume,data_nastere,nivel,data_inscriere,data_incheiere)
VALUES (5,'Apetrei','Mara',TO_DATE('1996-04-13','YYYY-MM-DD'),'Mediu',TO_DATE('2019-02-12','YYYY-MM-DD'),
        TO_DATE('2021-04-12','YYYY-MM-DD'));
```


~ *MASINA*

```
INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (1, 'Audi', 'A5', to_date('2007-06-09','YYYY-MM-DD'));

INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (1, 'Audi', 'A4', to_date('2002-06-09','YYYY-MM-DD'));

INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (2, 'Dacia', 'Logan', to_date('2005-07-01','YYYY-MM-DD'));

INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (2, 'Opel', 'Astra', to_date('2000-04-03','YYYY-MM-DD'));

INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (2, 'Volkswagen', 'Polo', to_date('2004-12-10','YYYY-MM-DD'));

INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (3, 'Audi', 'A5', to_date('2010-07-09','YYYY-MM-DD'));

INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (3, 'Volkswagen', 'Golf', to_date('2001-09-24','YYYY-MM-DD'));

INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (4, 'Mercedes-Benz', 'Q7', to_date('2012-04-09','YYYY-MM-DD'));

INSERT INTO masina_bpa( id_scoala,marca,model,data_fabricare)
VALUES (5, 'Mercedes-Benz', 'Q1', to_date('2003-12-09','YYYY-MM-DD'));
```

~ *EXAMEN*

```
INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (4,2,'Admis','C');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (6,2,'Respins','C');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (7,2,'Respins','C');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (5,1,'Admis','C');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (4,1,'Respins','C');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (6,9,'Respins','C');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (7,11,'Respins','D');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (8,8,'Admis','E');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (9,3,'Respins','E');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (7,3,'Admis','E');

INSERT INTO examen_bpa(id_ruta, id_elev,calificativ,categorie)
VALUES (4,3,'Respins','E');
select * from examen_bpa;
```

~ ARE_INSTRUCTOR

```
INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (1,1,12,1200);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (3,1,20,1400);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (2,1,3,700);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (3,4,20,2200);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (4,3,10,1100);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (5,6,3,700);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (4,5,25,3500);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (1,5,5,500);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (3,6,13,1800);

INSERT INTO are_instructor_bpa(id_scoala,id_instructor,nr_ore,salariu)
VALUES (5,2,3,700);
```

- Interogări

```
388 --1. Sa se afiseze instructorii care au cel mult 2 elevi. Sa se sorteze alfabetic dupa nume.
389
390 SELECT e.id_instructor as "Id instructor", i.numa as "Nume instructor", i.prenume as "Prenume instructor",
391        count(e.id_elev) as "Numar elevi"
392 FROM elev_bpa e
393 JOIN instructor_bpa i ON (e.id_instructor=i.id_instructor)
394 GROUP BY e.id_instructor, i.numa, i.prenume
395 HAVING count(e.id_elev)<=2
396 ORDER BY 2;
397
```

Script Output x Query Result x

SQL | All Rows Fetched: 3 in 0.008 seconds

	Id instructor	Nume instructor	Prenume instructor	Numar elevi
1	1	Ionescu	Ionel	2
2	3	Mihnea	Aurel	1
3	5	Popa	Irina	2


```
398 --2. Sa se afiseze codul instructorilor care nu lucreaza la aceasi scoala cu instructorul care are nr_ore = 25.
399
400 SELECT id_instructor
401 FROM are_instructor_bpa
402 MINUS
403 SELECT id_instructor
404 FROM are_instructor_bpa
405 WHERE id_scoala IN (SELECT id_scoala
406                     FROM are_instructor_bpa
407                     WHERE nr_ore = 25);
408
```

Script Output x Query Result x
SQL | All Rows Fetched: 4 in 0.008 seconds

ID_INSTRUCTOR
1
2
3
4

```
409 --3. Sa se afiseze elevii care au inceput si incheiat scoala in 2019.
410
411 SELECT *
412 FROM elev_bpa
413 WHERE TO_CHAR(data_inscriere, 'YYYY') = '2019' and TO_CHAR(data_incheiere, 'YYYY') = '2019' ;
414
```

Script Output x Query Result x
SQL | All Rows Fetched: 1 in 0.016 seconds

ID_ELEV	ID_INSTRUCTOR	NUME	PRENUME	DATA_NASTERE	NIVEL	DATA_INSCRIERE	DATA_INCHEIERE
1	3	2	Proteasa Vlad	09-DEC-99	Avansat	05-MAR-19	05-MAY-19

```
415 --4. Sa se afiseze elevul care a incheiat scoala de soferi dupa cel mai mare numar de zile.
416
417 SELECT id_elev,nume,prenume,data_inscriere,data_incheiere,data_incheiere-data_inscriere as "Nr zile"
418 FROM elev_bpa
419 WHERE data_incheiere-data_inscriere =
420     (
421         SELECT MAX(data_incheiere-data_inscriere)
422         FROM elev_bpa
423     );
424
```

Script Output x Query Result x
SQL | All Rows Fetched: 1 in 0.008 seconds

ID_ELEV	NUME	PRENUME	DATA_INSCRIERE	DATA_INCHEIERE	Nr zile
1	12	Apetrei Mara	12-FEB-19	12-APR-21	790

```
426 --5. Sa se afiseze instructorii elevilor ce au fost admisi dupa examen.
427
428 SELECT DISTINCT i.id_instructor, i.numa, i.prenume
429 FROM examen_bpa ex
430 JOIN elev_bpa e ON (e.id_elev=ex.id_elev)
431 JOIN instructor_bpa i ON (i.id_instructor=e.id_instructor)
432 WHERE LOWER(ex.calificativ)='admis';
433
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0.013 seconds

	ID_INSTRUCTOR	NUMA	PRENUME
1	3	Mihnea	Aurel
2	2	Marinescu	Monica

```
434 --6 Sa se afiseze scoala de la care provin elevii care au categoria C la examen.
435
436 SELECT DISTINCT s.id_scoala,s.numa
437 FROM examen_bpa ex
438 JOIN elev_bpa e ON (e.id_elev=ex.id_elev)
439 JOIN instructor_bpa i ON (i.id_instructor=e.id_instructor)
440 JOIN are_instructor_bpa ai ON (ai.id_instructor=i.id_instructor)
441 JOIN scoala_bpa s ON (s.id_scoala=ai.id_scoala)
442 WHERE LOWER(ex.categorie)='c';
443
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0.021 seconds

	ID_SCOALA	NUMA
1	3	Drive School
2	5	Mara

```
444 --7. Sa se afiseze elevii cu durata examenului cea mai mare.
445
446 SELECT e.id_elev,e.numa,e.prenume
447 FROM ruta_bpa r
448 JOIN examen_bpa ex ON (ex.id_ruta = r.id_ruta)
449 JOIN elev_bpa e ON (e.id_elev = ex.id_elev)
450 WHERE r.ora_sosire-r.ora_plecare =
451 (
452     SELECT MAX(ruta_bpa.ora_sosire-ruta_bpa.ora_plecare)
453     FROM ruta_bpa
454 );
455
456
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0.009 seconds

	ID_ELEV	NUMA	PRENUME
1	9	Bratu	Valerian
2	2	Ghimis	Cristian

(Se întâmplă să fie mai mulți elevi cu aceeași durată a examenului, deoarece susțin examenul pe aceeași rută prestabilită)

```
457 --8. Sa se afiseze instructorul si nr de elevi ai sai. Sa se sorteze descrescator dupa numarul de elevi, iar in caz de egalitate
458 --alfabetic dupa nume.
459
460 SELECT i.id_instructor,i.nume as "Nume instructor",count(e.id_elev) as "Nr elevi"
461 FROM elev_bpa e
462 JOIN instructor_bpa i ON (e.id_instructor=i.id_instructor)
463 GROUP BY e.id_instructor, i.nume, i.id_instructor
464 ORDER BY 3 DESC, 2 ASC;
465
```

Script Output x Query Result x

SQL | All Rows Fetched: 5 in 0.012 seconds

ID_INSTRUCTOR	Nume instructor	Nr elevi
1	2 Marinescu	3
2	4 Mateescu	3
3	1 Ionescu	2
4	5 Popa	2
5	3 Mihnea	1

```
466 --9. Sa se afiseze scolile care au cei mai multi elevi inscrisi.
467
468 SELECT x.id_scoala, x.nume, MAX("Nr elevi") as "Numarul elevilor" from (SELECT s.id_scoala,s.nume,count(e.id_elev) as "Nr elevi"
469 FROM examen_bpa ex
470 JOIN elev_bpa e ON (e.id_elev=ex.id_elev)
471 JOIN instructor_bpa i ON (i.id_instructor=e.id_instructor)
472 JOIN are_instructor_bpa ai ON (ai.id_instructor=i.id_instructor)
473 JOIN scoala_bpa s ON (s.id_scoala=ai.id_scoala)
474 GROUP BY s.id_scoala, s.nume) x
475 GROUP BY x.id_scoala, x.nume
476 HAVING MAX("Nr elevi") = (
477 SELECT MAX(count(e2.id_elev))
478 FROM examen_bpa ex2
479 JOIN elev_bpa e2 ON (e2.id_elev=ex2.id_elev)
480 JOIN instructor_bpa i2 ON (i2.id_instructor=e2.id_instructor)
481 JOIN are_instructor_bpa ai2 ON (ai2.id_instructor=i2.id_instructor)
482 JOIN scoala_bpa s2 ON (s2.id_scoala=ai2.id_scoala)
483 GROUP BY s2.id_scoala, s2.nume
484 );
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.026 seconds

ID_SCOALA	NUME	Numarul elevilor
1	5 Mara	8

```
487 --10. Sa se afiseze instructorii care au elevi cu nivelul incepator si care au fost admisi la examen.
488
489 SELECT DISTINCT i.id_instructor, i.nume, i.preume,e.nivel
490 FROM examen_bpa ex
491 JOIN elev_bpa e ON (e.id_elev=ex.id_elev)
492 JOIN instructor_bpa i ON (i.id_instructor=e.id_instructor)
493 WHERE LOWER(ex.calificativ)='admis' and LOWER(e.nivel) = 'incepator';
494
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0.012 seconds

ID_INSTRUCTOR	NUME	PRENUME	NIVEL
1	2 Marinescu	Monica	Incepator
2	3 Mihnea	Aurel	Incepator


```

495 --11. Sa se afiseze toate persoanele(instructori+elevi) care au anul nasterii dupa anul 1987.
496
497 SELECT nume, prenume, data_nastere
498 FROM instructor_bpa
499 WHERE TO_CHAR(data_nastere, 'YYYY') > '1987'
500 UNION
501 SELECT nume, prenume, data_nastere
502 FROM elev_bpa
503 WHERE TO_CHAR(data_nastere, 'YYYY') > '1987' ;

```

Script Output x Query Result x			
SQL All Rows Fetched: 14 in 0.012 seconds			
	NUME	PRENUME	DATA_NASTERE
1	Apetrei	Mara	13-APR-96
2	Baniciu	Andrei	05-MAR-89
3	Barda	Monica	12-JUL-99
4	Bratu	Valerian	08-APR-96
5	Dumitrana	Daniel	16-OCT-97
6	Gheorghe	Ionut	09-DEC-99
7	Ghimis	Cristian	10-FEB-00
8	Ionescu	Ionel	09-DEC-89
9	Mario	Marcel	09-JUL-98
10	Mateescu	Alexandru	21-JAN-90
11	Petre	Maria	09-MAR-99
12	Proteasa	Vlad	09-DEC-99
13	Sima	Ion	09-JUN-99
14	Vasilescu	Dariana	19-APR-98

```

505 --12. Sa se afiseze pentru salariile instructorilor cuprinse intre [0,800) "MIC" , [800,1300) "MEDIU",
506 --[1300,2500] "MARE" , peste 2500 "FOARTE MARE".
507
508 SELECT salariu,
509 CASE
510     WHEN (salariu < 800) THEN 'MIC'
511     WHEN (salariu >= 800 AND salariu < 1300) THEN 'MEDIU'
512     WHEN (salariu >= 1300 AND salariu < 2500) THEN 'MARE'
513     ELSE 'FOARTE MARE'
514     END as "TIP SALARIU"
515 FROM are_instructor_bpa;

```

Script Output x Query Result x		
SQL All Rows Fetched: 10 in 0.009 seconds		
	SALARIU	TIP SALARIU
1	1200	MEDIU
2	2200	MARE
3	700	MIC
4	1100	MEDIU
5	700	MIC
6	3500	FOARTE MARE
7	1800	MARE
8	700	MIC
9	1400	MARE
10	500	MIC

Pascu Andreea Bianca
Grupa 244

```

517 --13. Sa se afiseze elevii al caror nume contine doua litere 'a' sau al caror prenumele are exact 3 litere si
518 --prenumele instructorului acestora incepe cu litera 'i'.
519
520 SELECT e.id_elev as "Id elev",e.numa as "Nume elev",e.prenume as "Prenume elev", i.numa as "Nume instructor",
521        i.prenume as "Prenume instructor"
522 FROM elev_bpa e
523 JOIN instructor_bpa i ON(e.id_instructor = i.id_instructor)
524 WHERE (LOWER(e.numa) LIKE '%a%a%' OR LENGTH(e.prenume) = 3) AND (LOWER(i.numa) LIKE 'i%');
525

```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0.008 seconds

Id elev	Nume elev	Prenume elev	Nume instructor	Prenume instructor
1	6 Sima	Ion	Ionescu	Ionel
2	7 Dumitrana	Daniel	Ionescu	Ionel

```

526 --14. Sa se afiseze toti instructorii si elevii lor, inclusiv instructorii care nu au elevi.
527
528 SELECT i.id_instructor, i.numa as "Nume instructor", i.prenume as "Prenume instructor", e.id_elev as "Id elev",
529        e.numa as "Nume elev",e.prenume as "Prenume elev"
530 FROM instructor_bpa i
531 LEFT JOIN elev_bpa e ON (i.id_instructor = e.id_instructor);
532

```

Script Output x Query Result x

SQL | All Rows Fetched: 12 in 0.011 seconds

ID_INSTRUCTOR	Nume instructor	Prenume instructor	Id elev	Nume elev	Prenume elev
1	2 Marinescu	Monica	1 Gheorghe	Ionut	
2	2 Marinescu	Monica	2 Ghimis	Cristian	
3	2 Marinescu	Monica	3 Proteasa	Vlad	
4	5 Popa	Irina	5 Barda	Monica	
5	1 Ionescu	Ionel	6 Sima	Ion	
6	1 Ionescu	Ionel	7 Dumitrana	Daniel	
7	3 Mihnea	Aurel	8 Vasilescu	Dariana	
8	4 Mateescu	Alexandru	9 Bratu	Valerian	
9	4 Mateescu	Alexandru	10 Mario	Marcel	
10	4 Mateescu	Alexandru	11 Petre	Maria	
11	5 Popa	Irina	12 Apetrei	Mara	
12	6 Baniciu	Andrei	(null)	(null)	(null)

```

534 --15. Sa se afiseze elevii care au dat examen de mai multe ori.
535
536 SELECT e.numa as "Nume",e.prenume as "Prenume", count(e.id_elev) as "Numar sustineri examen"
537 FROM elev_bpa e
538 JOIN examen_bpa ex ON (ex.id_elev = e.id_elev)
539 GROUP BY e.numa, e.prenume
540 HAVING COUNT(e.id_elev) > 1;

```

Script Output x Query Result x

SQL | All Rows Fetched: 3 in 0.011 seconds

Nume	Prenume	Numar sustineri examen
1 Proteasa	Vlad	3
2 Ghimis	Cristian	3
3 Gheorghe	Ionut	2

```
544 --16. Sa se afiseze toate id-urile instructorilor care lucreaza la scoli cu rating = 7.
545
546 SELECT DISTINCT id_instructor
547 FROM are_instructor_bpa ail
548 WHERE NOT EXISTS
549     (SELECT 1
550      FROM scoala_bpa s
551      WHERE rating = 7
552      AND NOT EXISTS
553          (SELECT 2
554           FROM are_instructor_bpa ai2
555           WHERE s.id_scoala = ai2.id_scoala
556           AND ai2.id_instructor=ail.id_instructor
557          ))
558 ORDER BY 1;
```

Script Output x Query Result x

SQL | All Rows Fetched: 3 in 0.009 seconds

ID_INSTRUCTOR
1
2
3

- Tabelele populate cu date

INSTRUCTOR_BPA				
Columns Data Model Constraints Grants Statistics Triggers Flashback De				
Sort.. Filter:				
ID_INSTRUCTOR	NUME	PRENUME	DATA_NASTERE	
1	1 Ionescu	Ionel	09-DEC-89	
2	2 Marinescu	Monica	12-FEB-72	
3	3 Mihnea	Aurel	29-NOV-77	
4	4 Mateescu	Alexandru	21-JAN-90	
5	5 Popa	Irina	29-DEC-82	
6	6 Baniciu	Andrei	05-MAR-89	

ARE_INSTRUCTOR_BPA				
Columns Data Model Constraints Grants Statistics Triggers Flashba				
	ID_SCOALA	ID_INSTRUCTOR	NR_ORE	SALARIU
1	1	1	12	1200
2	3	4	20	2200
3	2	1	3	700
4	4	3	10	1100
5	5	6	3	700
6	4	5	25	3500
7	3	6	13	1800
8	5	2	3	700
9	3	1	20	1400
10	1	5	5	500

SCOALA_BPA					
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Part					
	ID_SCOALA	NUME	ADRESA	CAPACITATE	RATING
1	1	Cama	Strada Maria Tanase 3	12	8
2	2	TOP	Calea Mosilor 84	5	9
3	3	Drive School	Strada Olteniei 35-37	20	7
4	4	Teo	Strada Florilor 12	15	5
5	5	Mara	Strada Ion Creanga 45	20	10

MASINA_BPA					
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Det					
	ID_MASINA	ID_SCOALA	MARCA	MODEL	DATA_FABRICARE
1	1	1	Audi	A5	09-JUN-07
2	2	2	Audi	A4	09-JUN-02
3	3	2	Dacia	Logan	01-JUL-05
4	4	2	Opel	Astra	03-APR-00
5	5	2	Volkswagen	Polo	10-DEC-04
6	6	3	Audi	A5	09-JUL-10
7	7	3	Volkswagen	Golf	24-SEP-01
8	8	4	Mercedes-Benz	Q7	09-APR-12
9	9	5	Mercedes-Benz	Q1	09-DEC-03

ELEV_BPA								
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL								
	ID_ELEV	ID_INSTRUCTOR	NUME	PRENUME	DATA_NASTERE	NIVEL	DATA_INSCRIERE	DATA_INCHEIERE
1	1	2	Gheorghe	Ionut	09-DEC-99	Inceptor	09-DEC-20	09-FEB-21
2	2	2	Ghimis	Cristian	10-FEB-00	Mediu	10-FEB-20	10-APR-20
3	3	2	Proteasa	Vlad	09-DEC-99	Avansat	05-MAR-19	05-MAY-19
4	5	5	Barda	Monica	12-JUL-99	Avansat	25-SEP-20	25-NOV-20
5	6	1	Sima	Ion	09-JUN-99	Mediu	09-DEC-20	09-FEB-21
6	7	1	Dumitrana	Daniel	16-OCT-97	Mediu	28-AUG-18	28-OCT-18
7	8	3	Vasilescu	Dariana	19-APR-98	Inceptor	09-DEC-18	09-FEB-19
8	9	4	Bratu	Valerian	08-APR-96	Avansat	08-APR-20	08-JUN-20
9	10	4	Mario	Marcel	09-JUL-98	Inceptor	09-JUN-20	09-AUG-21
10	11	4	Petre	Maria	09-MAR-99	Mediu	09-JUL-18	09-SEP-18
11	12	5	Apetrei	Mara	13-APR-96	Mediu	12-FEB-19	12-APR-21

EXAMEN_BPA					
Columns Data Model Constraints Grants Statistics Triggers Flashback Depend					
	ID_EXAMEN	ID_RUTA	ID_ELEV	CALIFICATIV	CATEGORIE
1	1	4	2	Admis	C
2	2	2	5	1	Admis
3	3	6	9	Respins	C
4	4	7	11	Respins	D
5	5	8	8	Admis	E
6	6	9	3	Respins	E
7	7	6	2	Respins	C
8	8	7	3	Admis	E
9	9	7	2	Respins	C
10	10	4	3	Respins	E
11	11	4	1	Respins	C

RUTA_BPA

ColumnsDataModelConstraintsGrantsStatisticsTriggersFlashbackDependenciesDetailsPartitionsIndexesSQL

Sort..Filter:

	ID_RUTA	NR_KM	ORA_PLECAR	ORA_SOSIRE
1	4	10	10-SEP-20 02.10.10.1230000000 PM	10-SEP-20 04.10.10.1230000000 PM
2	5	15	22-JUN-21 01.10.10.1230000000 PM	22-JUN-21 03.30.10.1230000000 PM
3	6	13	30-AUG-20 04.30.10.1230000000 PM	30-AUG-20 07.10.10.1230000000 PM
4	7	20	30-AUG-18 11.10.10.1230000000 AM	30-AUG-18 12.10.10.1230000000 PM
5	8	5	14-FEB-19 03.50.10.1230000000 PM	14-FEB-19 04.10.10.1230000000 PM
6	9	21	20-FEB-20 02.10.10.1230000000 PM	20-FEB-20 04.10.10.1230000000 PM