Assumptions

1. A father doesn't exist unless with a mother.
2. An institute has a unique code just like study institute has Dlis code.
3. An institute may have no midwives in extreme cases.
4. One father and one mother form a couple.
5. We can make artificial keys for sessions, eg, 2022012903, 20220129 for date and time, 03 for the third sessions on that date.
6. Times of pregnancy relates to the couple not wife.
7. An appointment is only set up for one couple and one midwife.
8. A test cannot be prescribed without an appointment.
9. Notes can only be written by one midwife at one appointment.
10. A test must and only conducted by one technician.
11. Assume babies' gender and blood type can only be found out by tests.

Artificial keys:
Institute Code;
Baby ID;
SessionID;
CoupleID;
AppointmentID;
NotesID;
TestID;

Restrictions

1. In my model, couple is determined by mother since a mother have or don't have a husband. Thus, I find it hard to achieve that a mother can find another husband to make a couple. But since the law indicates one to one marriage, I think we just break the couple if they divorce and then make a new couple.
2. I don't know how to restrict a test only to a mother or child, but since it's related to appointment, it really doesn't make sense to add that restriction or make it visualized in the model I guess.
3. Since we can assign a primary and a back up midwife to a couple at the same time, I don't think we can restrict two or even more back up midwives.
4. Notes can only be written by one midwife at one appointment and if backup midwife want to add something, the notes cannot be related to backup midwife.
5. In the case where several test for expected birth date are conducted and the results differs, midwife and mother determines the date. I set up relationship between tests and mother to keep data of the date, but I am not 100% able to depict when the midwife will determine the date like after one or two tests on ER model.
6. If mother and father's blood type are found out not by tests, it's redundant to depict them in

ER model as we may just update mother and father entity.
7. I am not sure about depicting how couple assigns name to babies on ER; I suppose we first have baby ID available and assign name to them and this need not be shown on ER.
8. I don't think it's necessary to depict restriction of birth place on ER, since we can just set the birth place to choose from home or birthing clinic in the table.

Relational Translation

Midwife(prac-id,name, email, phone number,institute code)
Institute code is Foreign key referencing Institute.

Institute(institute code,name)
Community Clinic(institute code,name)
Birthing center((institute code,name,birthing facility)
We can combine institute into a large relation.
That is: Institute(institute code,name,birthing facility)
In this case community clinic will have NULL for ther birthing facility.

Mother(HCardID,email,expected time of birth,date of birth,name,phone#,profession,blood type,address)
Father(HCardID,name,FatherHCardID,email,expectedtimeofbirth,dateofbirth,phone#,profession, blood type,address)
HCardID is foreign key referencing Mother.
I don't think there is a way to reduce redundency of information about father and mother,since father is dependent on mother's identity, no father exists without mother.
Couple(CoupleID,Times of Pregnancy)

Info Session(SessionID,lang,date,time,capacity,CoupleID,Prac-ID)
CoupleIDs is a collection of CoupleID which is Foreign key referencing Couple.
Prac-ID is Foreign key referencing Midwife.
We don't track if the couple attends the info session for simplicity.
Selection(CoupleID,SessionID,Institute Code)
CoupleID is Foreign key referencing Couple.
SessionID is Foreign key referencing session.
Institue Code is Foreign key referencing institute.
Assignment(Prac-ID,CoupleID,Primary/BackUp)
CoupleID is Foreign key referencing Couple.
Prac-ID is Foreign key referencing Midwife.

Appointment(AppointmentID,Prac-ID,CoupleID,Notes IDs,Date,Time,TestID)
CoupleID is Foreign key referencing Couple.
Prac-ID is Foreign key referencing Midwife.
Notes IDs is a collection of Notes ID which is Foreign key referencing Note.
TestID is Foreign key referencing Test.

Notes(Notes ID, Date,Time,Observation,AppointmentID)
AppointmentID is Foreign key referencing Appointment.

Test(TestID,Type,DateScheduled,DateTaked,Result,AppointmentID,DateDoneAtLab,TechID)
AppointmentID is Foreign key referencing Appointment.
TechID is Foreign key referencing Technician.
Technician(TechID,Name,Phone#)
TestUpdateBirthDate(TestsID,HCardID,Date)
TestID is Foreign key referencing Tests.
AppointUpdateBirthDate(AppointmentID,HCardID,Date)
AppointmentID is Foreign key referencing Appointment.
discussToUpdateBirthDate(Prac-ID,HCardID,updatedDate)
HCardID is foreign key referencing Mother.
Prac-ID is Foreign key referencing Midwife.
Update blood type(TestID,HCardID,bloodType)
TestID is Foreign key referencing Test.
HCardID is foreign key referencing Mother.
I found it hard to build models for updating father's blood type, since this information is vague and I think we can just look up father in the table and change the blood type.

BabiesDetermines(TestID,HCardID,babies #,babyInfo)
TestID is Foreign key referencing Test.
HCardID is foreign key referencing Mother.
BabyInfo is a collection of info of babies(id,blood type,Date of Birth,Gender) in which BabyID is a foreign key referencing Baby.
Baby(BabyID,blood type,DateOfB,Name,Gender,HCardID,location)
HCardID is foreign key referencing Mother.
GiveBirthTo(CoupleID,location,BabyID)
CoupleID is foreign key referencing Couple.
BabyID is foreign key referencing Baby.