Table Scripts:

create table mountain\_huts ( id integer not null, name varchar(40) not null, altitude integer not null, unique(name) , unique(id) );

create table trails ( hut1 integer not null, hut2 integer not null );

insert into mountain\_huts values (1,'Dakonat',1900) ,(2,'Natisa',2100) ,(3,'Gajantut',1600) ,(4,'Rifat',782) ,(5,'tupur',1370) ; insert into trails values (1,3) ,(3,2) ,(3,5) ,(4,5) ,(1,5);

**Question:**

A ski company is planning to construct a new ski slope using a pre-existing network of mountain huts and trails between them. A new slope has to begin at one of the mountain huts, have a middle station at another hut connected with the first one by a direct trail, and end at the third mountain hut which is also connected by a direct trail to the second hut. The altitude of the three huts chosen for constructing the ski slope has to be strictly decreasing.

You are given two SQL tables, mountain\_huts and trails, with the following structure each entry table trails represents a direct connection between huts with IDs hut1 and hut2.

Note that all trails are bidirectional.

Create a query that finds all triplets(startpt, middlept, endpt) representing that mountain huts that may be used for construction of a ski slope. Output returned by the query can be ordered in any way.