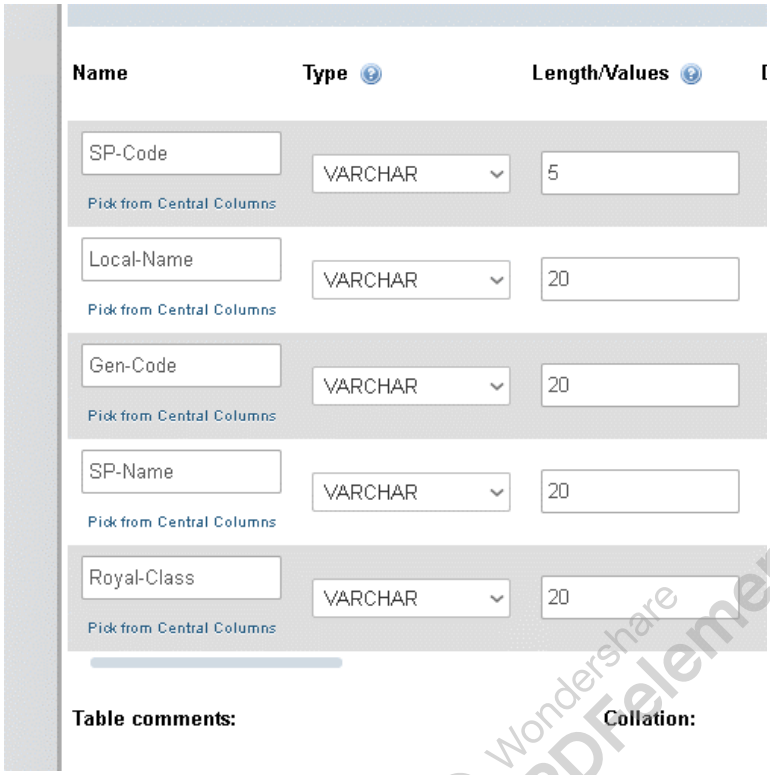
Step 1 : create database

create database call tree

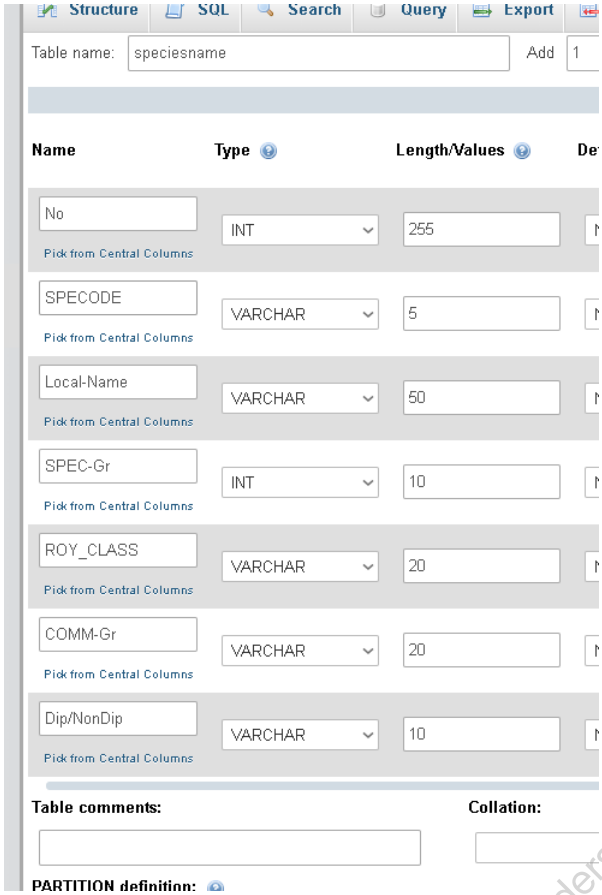
create table named :

a) scientificname :



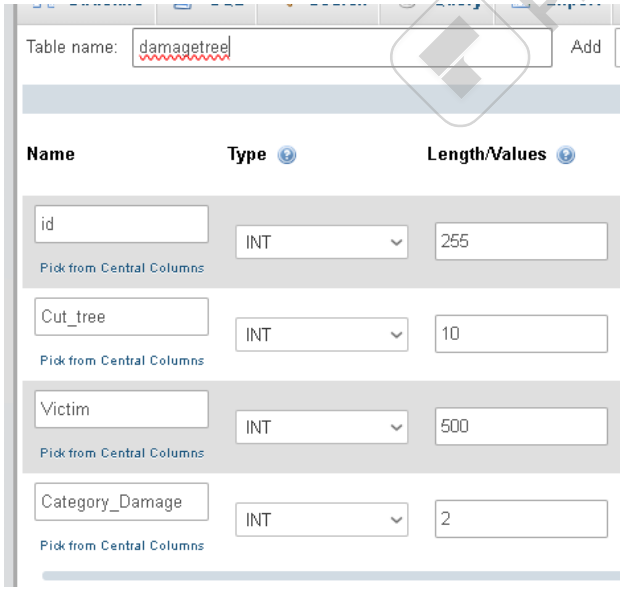
import scientifname. Csv

b) speciesname :

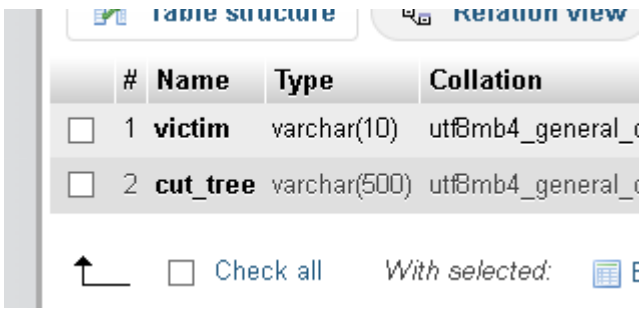


import species.csv

c) damagetree (change int for cut_tree and victim to varchar)



d) victim



#	Name	Type	Collation
1	victim	varchar(10)	utf8mb4_general_
2	cut_tree	varchar(500)	utf8mb4_general_

Check all With selected:

e) newforestori (set coordinate as varchar sorry mb)

Table structure | Relation view

#	Name	Type	Collation	Attributes
1	Id	int(11)		
2	BlockX	int(10)		
3	BlockY	int(10)		
4	x	int(100)		
5	y	int(100)		
6	coordinate	int(10)		
7	Treenum	varchar(9)	utf8mb4_general_ci	
8	species	varchar(4)	utf8mb4_general_ci	
9	spgroup	int(7)		
10	Diameter	float		
11	DiameterClass	int(10)		
12	StemHeight	float		
13	volume	float		
14	status_tree	varchar(10)	utf8mb4_general_ci	
15	Cut_Angle	int(11)		
16	Damage	int(11)		

Check all With selected: Browse Chan

Step 2 : Run create_forest.php

- Run it for 4 time to get 10 000 ++ data

Step 3 Run Sql Code in the new forest ori

1. UPDATE newforestori INNER JOIN speciesname ON newforestori.species = speciesname.No SET newforestori.species = speciesname.No
2. UPDATE newforestori SET Volume = 3.142 * POW((Diameter / 200), 2) * StemHeight * 0.50
3. UPDATE newforestori SET TreeNum = CONCAT('T', LPAD(BlockX, 2, '0'), LPAD(BlockY, 2, '0'), LPAD(x, 2, '0'), LPAD(y, 2, '0'))
4. UPDATE newforestori SET status_tree = CASE WHEN spgroup IN (1, 2, 3, 5) AND Diameter > 45 THEN 'Cut' WHEN spgroup IN (1, 2, 3, 5) AND Diameter <= 45 THEN 'Keep' ELSE status_tree END
5. UPDATE newforestori SET Cut_Angle = CASE WHEN status_tree = 'Cut' THEN FLOOR(RAND() * 360) + 1 ELSE NULL EN
6. UPDATE newforestori INNER JOIN speciesname ON speciesname.No = newforestori.species SET newforestori.species = speciesname.SPECODE
7. INSERT INTO victim (Victim, cut_tree, Category_damage) SELECT Victim, cut_tree, Category_damage FROM damagetree;
8. UPDATE newforestori nf
INNER JOIN (
SELECT Cut_tree,
SUM(CASE WHEN Category_Damage = 1 THEN 1 ELSE 0 END) AS stem_victim_count,
SUM(CASE WHEN Category_Damage = 2 THEN 1 ELSE 0 END) AS crown_victim_count
FROM damagetree
GROUP BY Cut_tree
) AS dt ON nf.cut_tree = dt.Cut_tree
SET nf.Damage_stem = dt.stem_victim_count,
nf.Damage_crown = dt.crown_victim_count;

Step 4 : run find_damage.php

- For produce the victim table

Step 5 : run find_victim.php

- For produce the cut table