

Step 1 : create database

create database call tree

create table named :

a) scientificname :

Name	Type ?	Length/Values ?	I
<input type="text" value="SP-Code"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="5"/>	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="Local-Name"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="20"/>	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="Gen-Code"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="20"/>	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="SP-Name"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="20"/>	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="Royal-Class"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="20"/>	
<a href="#">Pick from Central Columns</a>			

**Table comments:**

**Collation:**

import scientificname. Csv

b) speciesname :

Structure SQL Search Query Export

Table name:  Add 1

Name	Type	Length/Values	Default
<input type="text" value="No"/>	INT	255	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="SPECODE"/>	VARCHAR	5	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="Local-Name"/>	VARCHAR	50	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="SPEC-Gr"/>	INT	10	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="ROY_CLASS"/>	VARCHAR	20	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="COMM-Gr"/>	VARCHAR	20	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="Dip/NonDip"/>	VARCHAR	10	
<a href="#">Pick from Central Columns</a>			

Table comments:

Collation:

PARTITION definition: [?](#)

import species.csv

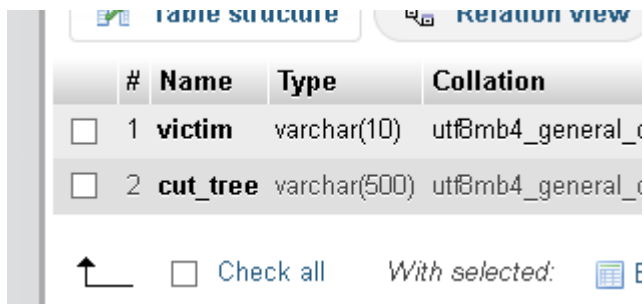
c) damagetree (change int for cut\_tree and victim to varchar)

Structure SQL Search Query Export

Table name:  Add 1


Name	Type	Length/Values	Default
<input type="text" value="id"/>	INT	255	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="Cut_tree"/>	INT	10	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="Victim"/>	INT	500	
<a href="#">Pick from Central Columns</a>			
<input type="text" value="Category_Damage"/>	INT	2	
<a href="#">Pick from Central Columns</a>			

d) victim



The screenshot shows the 'Table structure' tab for a table named 'victim'. The table has two columns: 'victim' (varchar(10)) and 'cut\_tree' (varchar(500)). Both columns have the 'utf8mb4\_general\_ci' collation. The interface includes a 'Check all' checkbox and a 'With selected:' dropdown menu.

#	Name	Type	Collation
<input type="checkbox"/> 1	<b>victim</b>	varchar(10)	utf8mb4_general_ci
<input type="checkbox"/> 2	<b>cut_tree</b>	varchar(500)	utf8mb4_general_ci


☐ Check all    With selected: 

e) newforestori (set coordinate as varchar sorry mb)



The screenshot shows the 'Table structure' tab for a table named 'newforestori'. The table has 16 columns: 'Id' (int(11)), 'BlockX' (int(10)), 'BlockY' (int(10)), 'x' (int(100)), 'y' (int(100)), 'coordinate' (int(10)), 'Treenum' (varchar(9)), 'species' (varchar(4)), 'spgroup' (int(7)), 'Diameter' (float), 'DiameterClass' (int(10)), 'StemHeight' (float), 'volume' (float), 'status\_tree' (varchar(10)), 'Cut\_Angle' (int(11)), and 'Damage' (int(11)). The 'Treenum', 'species', and 'status\_tree' columns have the 'utf8mb4\_general\_ci' collation. The interface includes a 'Check all' checkbox, a 'With selected:' dropdown menu, and buttons for 'Browse' and 'Change'.

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

☐ Check all    With selected:  Browse  Change

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 =  $3.142 * \text{POW}((\text{Diameter} / 200), 2) * \text{StemHeight} * 0.50$
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. UPDATE newforestori SET coordinate = CONCAT('T', LPAD(x, 2, '0'), LPAD(y, 2, '0')) }

Step 4 : run find\_damage.php

- For produce the victim table

Step 5 : run find\_victim.php

- For produce the cut table