Question: https://leetcode.com/problems/longest-common-prefix/

Initialize the common string as the first string of the list.

Now iterate through the rest of the list. And set the length of the common string as min of upcoming string length and common string length.

As common string cannot be bigger than the other strings.

Now check the substring of that length of common string and upcoming string matches or not, if matches then check the next upcoming string. Now if does not match then reduce the substring length and repeat the process until it matches.

If while doing it we get common string length as 0, then it means no common string exists so return an empty string.

Code:  
class Solution {

public String longestCommonPrefix(String[] strs) {

String prevCommon=strs[0];

for(int i = 0; i < strs.length-1; i++){

String upcomingstr=strs[i+1];

int j=Math.min(upcomingstr.length(), prevCommon.length());

while(j>0){

if(prevCommon.substring(0,j).equals(upcomingstr.substring(0,j))){

prevCommon=prevCommon.substring(0,j);

break;

}else{

j--;

}

}

if(j==0){

prevCommon="";

break;

}

}

return prevCommon;

}

}

Github Link :<https://lnkd.in/ecwtJeaz>