Answer the following questions using PHP/MySQL. Add comments where necessary.

Question 1

Two product classes, Line and Trunk, both require getServiceFee and setServiceFee functions (that return/manipulate their serviceFee member variable) as well as other functions unique to themselves. When completing the following tasks use type hinting where applicable. If you add any classes show only what is required of their interface/implementation to understand the rest of your code snippets.

a) Write the code to create a Line or Trunk in a situation where the type of object is only known at runtime.
b) Show the code for a function called increaseProductFee which is passed the object to be updated as well as the increase percentage. The function should assume we do not know whether the object passed to it is a Line or a Trunk.

Question 2

Fix any problems you can find with the following tables. **Show the SQL queries** needed to create your improved/new tables and explain why you have made the changes:

Student Table

name(varchar, PK)	**age**(int)	**grade**(int)	**classTeacher**(varchar)	**subjects**(varchar)
Johnny Jones	17	12	Robin Smith	"Eng/Afr/Math/Sci/Hist/Comp"
Susan King	17	12	Robin Smith	"Afr/Eng/Math/Sci/Geo/Hist"
…	…	…	…	…

Teacher Table

name(varchar, PK)	**subjects**(varchar)	**headOfGrade**(varchar)	**salary**(int)
Robin Smith	"Hist"	Yes	10000
John Doe	"Hist/Math"	No	5000
…	…	…	…

Question 3

A table called prefixMap stores all the prefixes required to correctly route a number to the correct network operator.

prefix	**destination**
011	ZA_Telkom
012	ZA_Telkom
0111234	ZA_Switch
013	ZA_Telkom
0131234567	ZA_Neotel
…	
…	

This means that when a client makes a call to 0111234567 we can tell from the prefix 0111234 that the operator is ZA_Switch as it overrides the rule for 011 (ZA_Telkom) as it is more specific. Note that this table will contain more then **two-and-a-half million entries** and be read **many times a second** so efficiency is important.

a) Write a function that will retrieve the correct destination when given a telephone number.
b) Write a function that will change the destination of a single number or a block of 10 or 100 numbers (The first number in a block always ends with 0 or 00 and the last number ends with 9 or 99. For example: 0111234560 to 0111234569, 0111234500 to 0111234599) to a given destination. A block can only be updated if all the numbers in that block are currently linked to the same destination (i.e. There are no "more specific" prefixMaps for numbers within that range - throw an exception if this is not the case). For example a block of 100 starting with 0111234500 should be given a prefixMap prefix of 01112345 as long as there are no prefixMap entries that match 01112345x or 01112345xx for different destinations.