USAGE NOTES for get_avail_trains and get_ticket

As I noted in class, get_avail_trains and get_ticket are not intended for stand-alone use but are meant to be called from a web page scripting language such as php, Python or node.js.

They both expect a variable to be set in the program to capture a status number from the stored procedure. In both cases, the stored procedures also write to tables, in the case of get_avail_trains the table avail_trains, and in the case of get_ticket Tickets and Passengers.

Thus, your program should check the return status integer from the sp and display the appropriate values in the tables in question.

You can test the programs stand-alone by setting a variable in the mysql shell and then referencing that variable in the OUT parameter when calling the stored procedure. You must also put a value of selected on one of the trains in avail_trains or else get_ticket will exit with a code of 2….no selection made. There is some sanity printing of variable values which you can remove.

I wrote both of the programs against railroad3 as it is a bit more developed than railroad1. Thus they refer to table names in railroad3. Watch out – the upper/lower case of the table names is not consistent. Thus most tables have a first capital letter or 2, as Trains, Passengers, Stations, Seats_Free but stops_at is not caps and neither is avail_trains. This makes for more buggy programming, sorry.

Also sorry but I lost the formatting in copying these sps back from Linux! Easily repaired…

I made some minor changes to the tables in railroad3, in addition to adding avail_trains. I added a method of payment field to Passenger. I populated the fares column of Segments using a random function. I had to change the precision of the fare fields as they were too small.

I posted a schema for avail_trains but you should look at the one in railroad3 and use that instead: because of the manner of updating I had to set specific default values for all fields.

Also, in testing the 2 sp's. you may want to change some of the Seats_Free entries for particular dates and segments to 0 so that those trains become unavailable for certain trips. You need to plan in advance the date, trains you expect to use, and seats free values to have a valid test.

Finally, as I emphasized in class, if you prefer, you can use the code in line in your application programs rather than using the stored procedures themselves.