2017

TPL: Table Programming Language

PROJECT PROPOSAL

HAMZA JAZMATI

Project Information:

As discussed over piazza with the TA, I will be doing this project alone.
Class Section: Hybrid Section.
Project Name: Table Programming Language.
Project ID: TPL
Student Uni: hj2441

Motivation:

I work at market risk technology department in a bank. I noticed that at my job most of the scripts are just processing different tables and producing new tables. These operations include a lot of joins (inner and outer), table concatenation (horizontal and vertical), column projection, lambda functions on some columns, and filtering data. Even further, a lot of people still use Microsoft Excel as a way of processing data, since it is more user friendly and can do the same functions mentioned earlier plus a few more like average, min, max, etc…

Types:

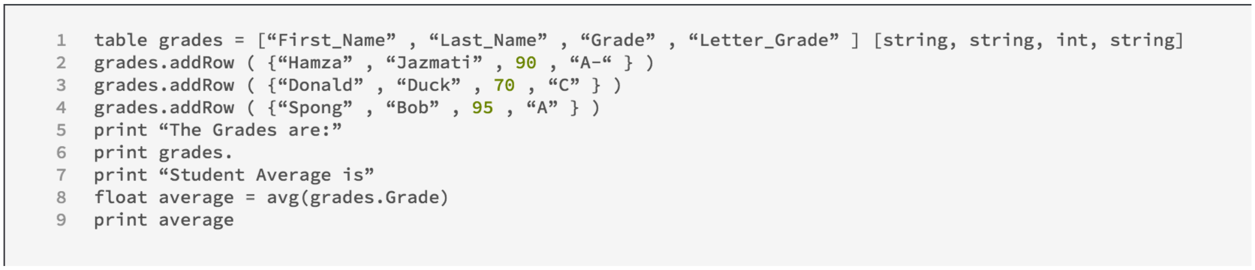
Table Programming Language (TPL) should support the following data types: *int*, *float*, *bool*, *string*, and *table*. Some of these data types might be missing from the final submission due to time limitation. Some other types might be added if seemed highly essential including *array*, *list*, or *dictionary (map)*. TPL will be a strong typed language. Developers looking to use *double* and *char* will have to use *float* and *string* respectively as an alternative.

Implementation:

For TPL, I will try to be as consistent as possible with most of the C programming language syntax. This includes variable initialization, comments, and the use of semicolons. Other similarities will be in the flow control, which will support at minimum *if/else* conditionals and *while* loops. Developers who need other flow control methods can use these two as a good substitute. At this point of the project, I am not sure yet if TPL final submission for the semester will focus on doing most of the data types and support arrays or will focus instead on adding most of the features to tables. Other features in C in like pointers will not be supported due time limitations.

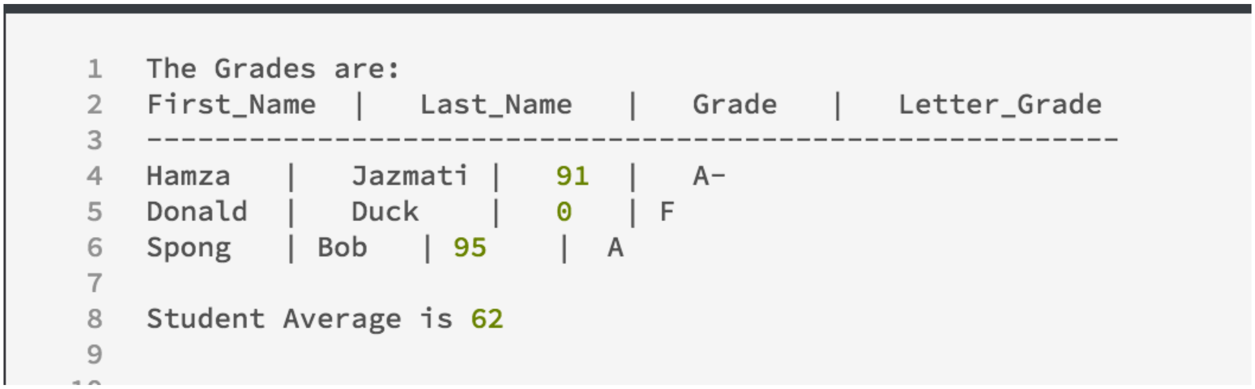
Source Code of a Sample Program:

In this program, we will have a table of grades for a couple of students. Then we will output the table with the data and output the average.



```
table grades = [“First_Name” , “Last_Name” , “Grade” , “Letter_Grade” ] [string, string, int, string]
grades.addRow ( {“Hamza” , “Jazmati” , 90 , “A-“ } )
grades.addRow ( {“Donald” , “Duck” , 70 , “C” } )
grades.addRow ( {“Spong” , “Bob” , 95 , “A” } )
print “The Grades are:”
print grades.
print “Student Average is”
float average = avg(grades.Grade)
print average
```

Output should be:



```
The Grades are:
First_Name  |   Last_Name   |   Grade   |   Letter_Grade
---------------------------------------------------------
Hamza   |   Jazmati |   91  |   A-
Donald  |   Duck    |   0   | F
Spong   | Bob   | 95    |  A

Student Average is 62

```