UML (Unified Modeling Language) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.

Program Project1: structure

student	
double quiz1	stores score from 0..10
double quiz2	stores score from 0..10
double midterm	stores score from 0..100
double final	stores score from 0..100
double average	stores average score
char letterGrade	stores letter grade

Program Project2: class

Student	
- quiz1 : double	stores score from 0..10
- quiz2 : double	stores score from 0..10
- midterm : double	stores score from 0..100
- final : double	stores score from 0..100
- average : double	stores average score
- letterGrade : char	stores letter grade
+ getQuiz1() : double const	retrieves score of quiz1
+ setQuiz1(newScore : double) void	sets new score of quiz1
+ getQuiz2() : double const	retrieves score of quiz2
+ setQuiz2(newScore : double) void	sets new score of quiz2
+ getMidterm() : double const	retrieves score of midterm
+ setMidterm(newScore : double) void	sets new score of midterm
+ getFinal() : double const	retrieves score of final
+ setFinal(newScore : double) void	sets new score of final
+ calculateAverage(): void	sets average score
+ getAverage() : double	retrieves average score
+ calculateLetterGrade(): void	sets letter grade
+ getLetterGrade() : char	retrieves letter grade

Program Project7: class

rational	
- numerator : int	stores integer value of numerator
- denominator : int	stores integer value of denominator
+ rational ()	no-arguments constructor
+ rational (newNumerator : int, newDenominator : int)	arguments constructor
+ rational (newNumerator : int)	argument constructor
+ add(rational* r2) : rational	returns a rational of the addition
+ sub(rational* r2) : rational	returns a rational of the subtraction
+ mul(rational* r2) : rational	returns a rational of the multiplication
+ div(rational* r2) : rational	returns a rational of the division
+ neg() : rational	returns a negative of the rational
+ input(ins : istream) : istream	fetches rational from the keyboard
+ output(outs : ostream) : ostream	write rational to the console

Program Project8: class

Odometer	
- milesDriven : double	stores value of miles driven
- fuelEfficiency : double	stores value of fuel efficiency
+ Odometer()	no-arguments constructor
+ reset() : void	sets all data members to zeros
+ setFuelEfficiency(newValue : double) void	sets new value of fuel efficiency
+ setMilesDriven(newValue : double) void	sets new value of miles driven
+ gallonsConsumed() : double	returns the consumed gallons

Program Project11: class

Movie	
- name : string	?
- MPAARating : string	??
- Number_1Terribles : int	???
- Number_2Bads : int	????
- Number_3Oks : int	?????
- Number_4Goods : int	??????
- Number_5Greats : int	???????
+ Movie(newName : string, newMPAARating : string)	?
+ setName(newName : string) void	?
+ getName() string	?
+ setMPAARating(newMPAARating : string) void	?
+ getMPAARating() string const	?
+ AddRating(rating : int) void	?
+ getAverage() : double const	?