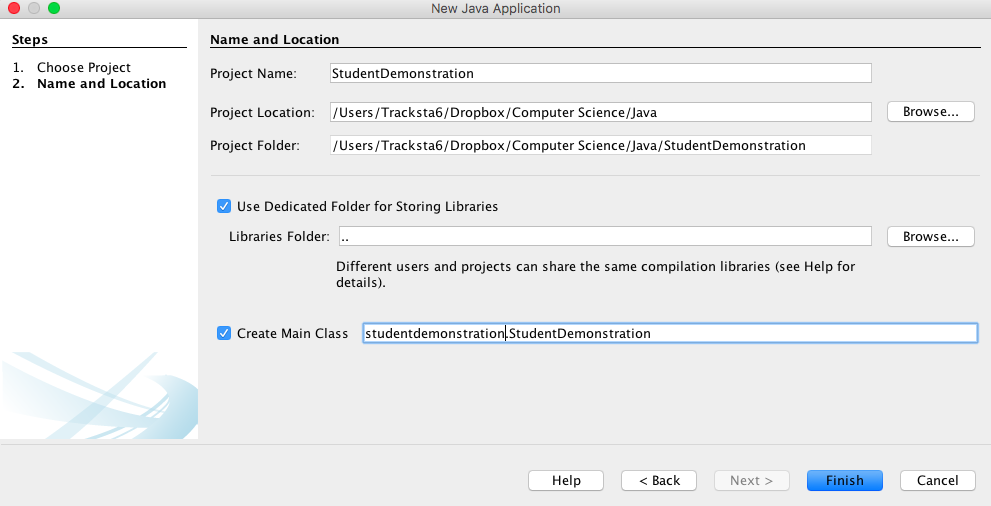
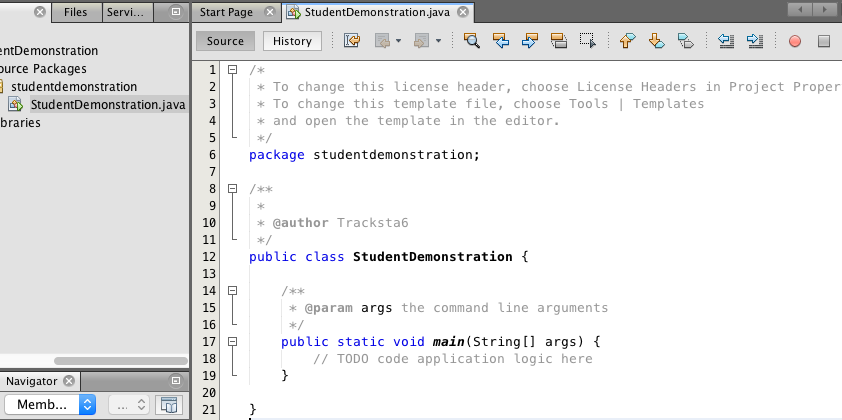
Java – Student Demonstration

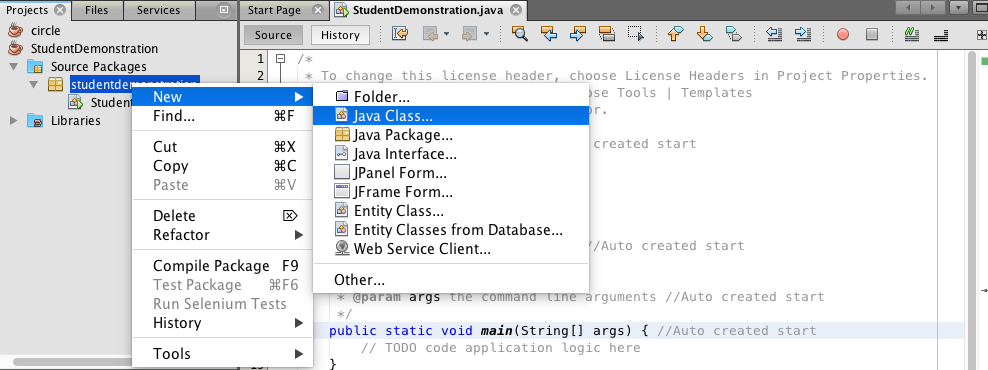
1. Open Netbeans
2. Select New Project
3. Name the project and select the directory.



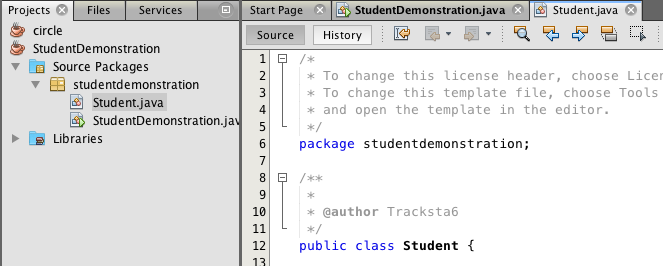
1. It will auto-create a new project with the very basic structure.



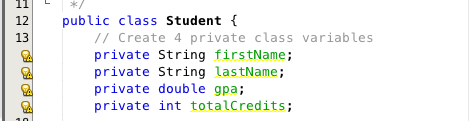
1. Now we will create a new class within this application.
   1. Select FileFolder top left: Select New: Click Java Class



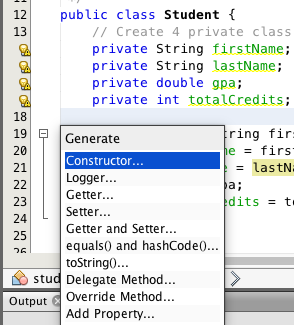
1. Name and place in directory as is appropriate.



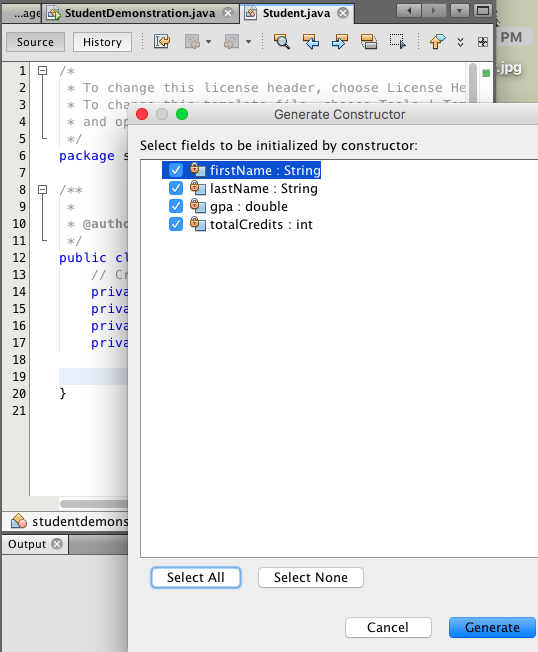
1. Now in your student class, create 4 class variables, firstname, lastname, gpa and credits.



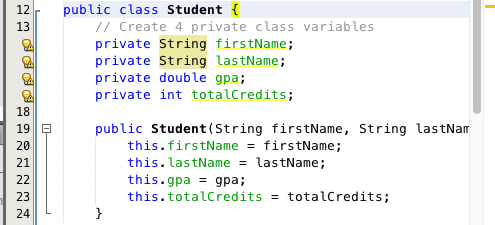
1. Now click below the class variable, anywhere in the class and right click. I believe alt+i will work too. This will auto include all 4 of the variables that we just created and let you select them in-order to potentially do many things include create constructors for them. Select “Constructor” on the list that pops up.



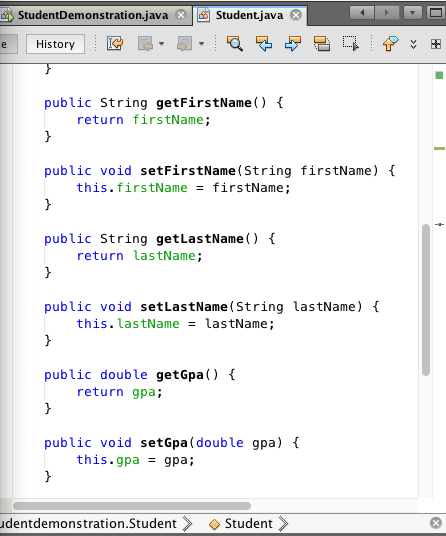
1. Now you can select the 4 variables that you want to create a constructors for on a new list.



1. Now it will auto create constructors for each.

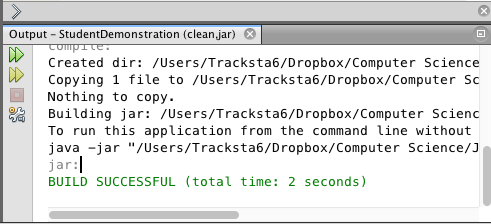


1. Now do the same things that we did to create the constructors but chose the seleciton for “Getters and Setters”

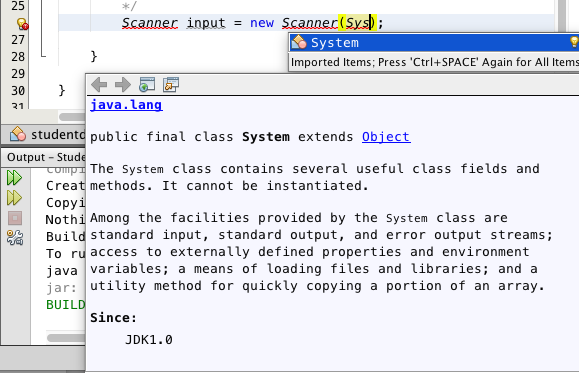


1. Same way with alt+i or right click insert
   1. Select the toString option
   2. This is stating that when we print a student it will print out these options.
      1. @Override
      2. public String toString() {
      3. return "Student{" + "firstName=" + firstName + ", lastName=" + lastName + ", gpa=" + gpa + ", totalCredits=" + totalCredits + '}';
      4. }

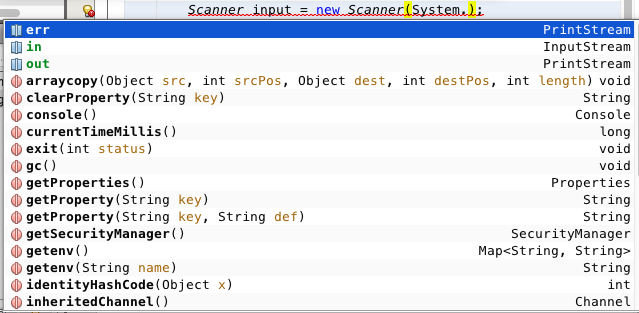
1. Now we have created a basic Student application with a class where we have created 4 variables, those variables constructors, the variable’s getters and setters, and the return format functions. Now we need to compile and clean the entire application.
   1. Press the top menu, ‘hammer and broom’ button.
   2. This should run through in the output section with many different messages that are not to understand but it should end with a “Build Successful”. That means our code has compiled cleanly without syntax errors. Clearly we are not done, but we are clean so far.



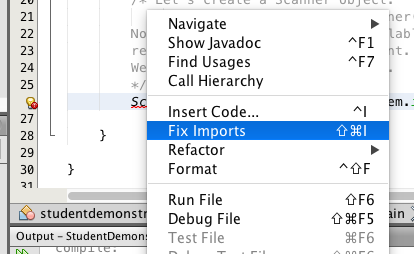
1. Now go to the main area in StudentDemonstration.java.
2. Create:
   1. Scanner input = new Scanner(System.in)
3. First: Note if you just type Sys and then hit alt+space it will give you a pop up with:



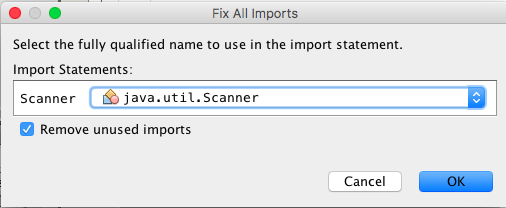
1. Double click System
2. Now, after System, type a period, and a new list will pop up:



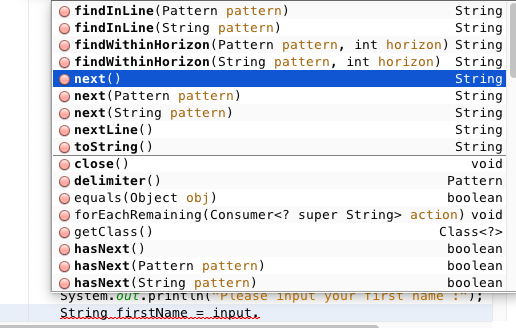
1. Here we will select ‘in’ (not sure why yet…)
   1. Scanner input = new Scanner(System.in);
2. BUT…Scanner will have a red underline since we have not imported yet.
3. Click once on Scanner, and right-click. Select ‘Fix Imports”



1. Now it will give us another selection that it is suggesting for us to import. This is not guaranteed, but this one is correct.



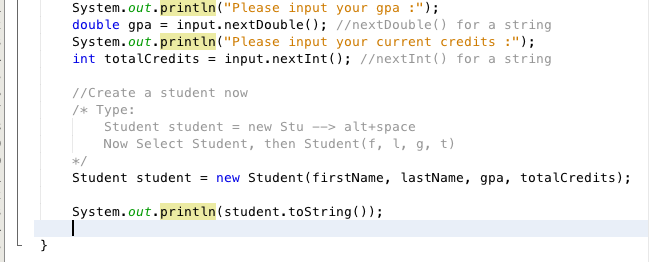
1. Now the Scanners in main will not have the red underline. And there will be a new line of code at the top: import java.util.Scanner;
2. Type 'sout + tab':
   1. System.out.println("");
   2. Then in the quotes, enter your phrase
   3. System.out.println("Please input your first name :");
   4. Clearly we want to get user input. Type:
   5. String firstName = input.
   6. After the dot, a popup will come with a list. Chose next()



* 1. String firstName = input.next()

1. Now we have:
   1. System.out.println("Please input your first name :");
   2. String firstName = input.next()
2. So we are asking for user input for a first name and getting it from the user. Repeat this for the lastname, gpa, and totalCredits.
   1. Know that gpa will use nextDouble() and totalCredits will use nextInt().

1. Now we have the full student set up, so lets create one.
2. Create a student now
   1. Type:
   2. Student student = new Stu --> alt+space to get a popup list
   3. Now Select Student, then Student(f, l, g, t) 🡪 now will have:
   4. Student student = new Student(firstName, lastName, gpa, totalCredits);

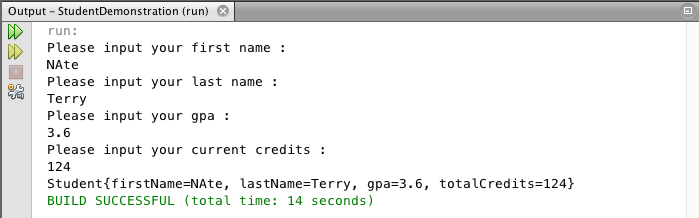


1. Now if we want to print out the student 🡪 sout + tab:

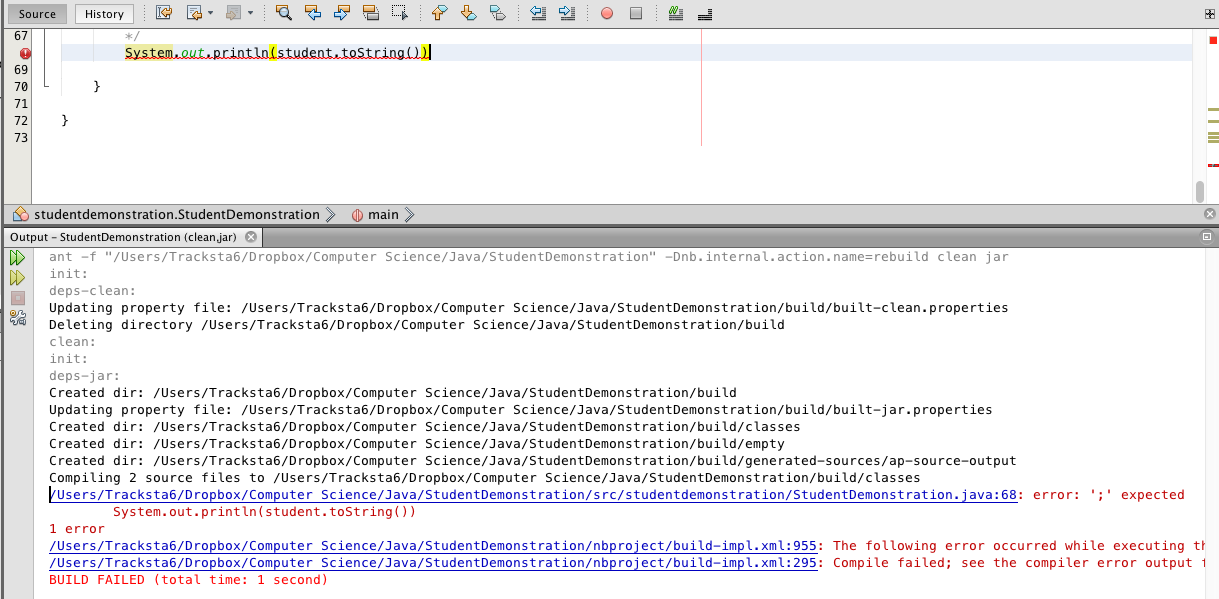
System.out.println("");

* 1. Change inside to student object. You can print specific studetn attributes or use the dot.toString() to print out the whole student.

1. Compile with the hammer+brush button. If successful…Run:



1. Note, if you have an error and your compile button tells you those errors, such as a missing semicolon.



1. It tells you line 68 for the error, but if you click on the hyperlink it will move your cursor to that line automatically to fix it.