

TASK	INSTRUCTIONS	verified
The simplest class	<p>1) open a text editor and type in the following class</p> <pre>public class GradeBook { }</pre> <p>2) save the file as GradeBook.java somewhere on your hard drive (remember where)</p> <p>3) issue the command javac GradeBook.java from the directory where you saved your source code</p> <p>4) display the directory and make sure you have GradeBook.java and GradeBook.class in the directory</p> <p>5) try and run GradeBook.java notice the error: Exception in thread "main" java.lang.NoClassDefFoundError: GradeBook</p> <p>you can only directly execute classes that have the public static void main(String args[]) method</p>	
Adding a method to our class	<p>6) add to your class by duplicating the following example</p> <pre>public class GradeBook { public void displayMessage() { System.out.println("Welcome to the Grade Book"); } }</pre> <p>7) save and compile the new file, if you received no errors move on.</p> <p>8) we still cannot directly run the java class it still has no main method</p>	
Creating a test client for our GradeBook class	<p>9) open another text editor or an additional file in the same text editor and type in the following class</p> <pre>public class GradeBookTest { public static void main(String args[]) { GradeBook cosc210GradeBook = new GradeBook(); cosc210GradeBook.displayMessage(); } }</pre> <p>10) save and compile this new java class source file</p> <p>If you see an error read the following If your file compiles skip to 13)</p> <pre>C:\\$downloads\iup\IntroToJava\ST0103_JavaBasics1> javac GradeBookTest.java GradeBookTest.java:3: cannot find symbol symbol : class GradeBook location: class GradeBookTest GradeBook cosc210GradeBook = new</pre>	

```
GradeBook();
```

```
^
```

```
GradeBookTest.java:3: cannot find symbol
```

```
symbol   : class GradeBook
```

```
location: class GradeBookTest
```

```
GradeBook cosc210GradeBook = new
```

```
GradeBook();
```

```
2 errors
```

The first time I compiled it it tells me I cannot find the symbol GradeBook.

so just like in class and the labs I set the CLASSPATH to tell the machine I am working on where to find my class files, my java executables.

11) tell your computer where to find the class files
(a) issued the set command to tell my command prompt where to find my class files, the period semicolon tells it to look in the current directory I am using.

```
set CLASSPATH=%CLASSPATH%;.;
```

or (b) I could have issued the following command:

```
set
```

```
CLASSPATH=%CLASSPATH%;C:\$downloads\iup\IntroToJava\STO103_JavaBasics1;
```

to receive the same results

or (c) I could have also set the CLASSPATH in control panel > system > advanced > environment variables

```
C:\$downloads\iup\IntroToJava\STO103_JavaBasics1>javac GradeBookTest.java
```

```
C:\$downloads\iup\IntroToJava\STO103_JavaBasics1>dir
```

```
Volume in drive C is Local Disk
```

```
Volume Serial Number is 8723-79C9
```

```
Directory of
```

```
C:\$downloads\iup\IntroToJava\STO103_JavaBasics1
```

```
01/28/2008 01:06 PM <DIR> .
```

```
01/28/2008 01:06 PM <DIR> ..
```

```
01/28/2008 12:58 PM 422 GradeBook.class
```

```
01/28/2008 12:58 PM 118 GradeBook.java
```

```
01/28/2008 01:06 PM 338 GradeBookTest.class
```

```
01/28/2008 01:03 PM 167 GradeBookTest.java
```

notice in the directory you are working on that you now have 2 java files and 2 class files

12) we have a GradeBook class and another that takes the GradeBook class and instantiates it into an object than sends a message to that object to invoke a behavior of that object using a method

run the test application to see the results of invoking the displayMessage() method java GradeBookTest

	Welcome to the Grade Book	
Alter the displayMessage() method to have a parameter	<p>13) alter GradeBook.java to look like the following:</p> <pre>public class GradeBook { public void displayMessage(String courseName) { System.out.printf("Welcome to the grade book for\n%s!\n", courseName); } }</pre> <p>14) save and compile GradeBook.java again we have added a parameter to the displayMessage method and used that parameter in the behavior of the method.</p> <p>15) we can't successfully invoke our test client again because the parameters on our method no longer match. If we were to invoke it as is we would see an error because it would not be able to find the method with the correct signature.</p> <p>you would see an error like the following:</p> <pre>java GradeBookTest Exception in thread "main" java.lang.NoSuchMethodError: GradeBook.displayMessage()V at GradeBookTest.main(GradeBookTest.java:4)</pre> <p>16) save GradeBookTest</p>	
Alter the GradeBook class to have an instance variable, a setter, and a getter	<p>17) edit GradeBook.java to look like the following:</p> <pre>public class GradeBook { private String courseName; public void setCourseName(String name) { courseName = name; } public String getCourseName() { return courseName; } public void displayMessage() { System.out.printf("Welcome to the grade book for\n%s!\n", getCourseName()); } }</pre> <p>18) save, compile GradeBook.java We cannot run the test client because it no longer matches the call in the GradeBook class</p>	
Edit GradeBookTest.java to take advantage	<p>19) edit GradeBookTest.java to take advantage of the getters and setters we just added in GradeBook.java as in the example below:</p> <pre>import java.util.Scanner; public class GradeBookTest { public static void main(String args[]) { Scanner input = new Scanner(System.in);</pre>	

Name _____

COSC 210 **LAB 2**

of the getters and setters we just added in GradeBook .java	<pre> GradeBook cosc210GradeBook = new GradeBook(); System.out.printf("Initial course name is: %s\n\n", cosc210GradeBook.getCourseName()); System.out.println("Please enter the course name:"); String theName = input.nextLine(); cosc210GradeBook.setCourseName(theName); System.out.println(); cosc210GradeBook.displayMessage(); } } </pre> <p>20) save GradeBookTest.java</p>	
Alter the GradeBook to initializ e with a construct or	<p>21) edit GradeBook.java to look like the following:</p> <pre> public class GradeBook { private String courseName; public GradeBook(String name){ courseName = name; } public void setCourseName(String name) { courseName = name; } public String getCourseName() { return courseName; } public void displayMessage() { System.out.printf("Welcome to the grade book for \n%s!\n", getCourseName()); } } </pre> <p>22) save, compile GradeBook.java</p>	
Edit GradeBook Test.java to use the construct or we just defined	<p>23) edit GradeBookTest.java to use the constructor we just defined</p> <pre> import java.util.Scanner; public class GradeBookTest { public static void main(String args[]) { Scanner input = new Scanner(System.in); GradeBook cosc210GradeBook = new GradeBook("COSC210 Java and GUI programming"); GradeBook cosc310GradeBook = new GradeBook("COSC310 Data Structures in Java"); System.out.printf("gradebook1 course name is: %s\n\n", cosc210GradeBook.getCourseName()); } } </pre>	

Name _____

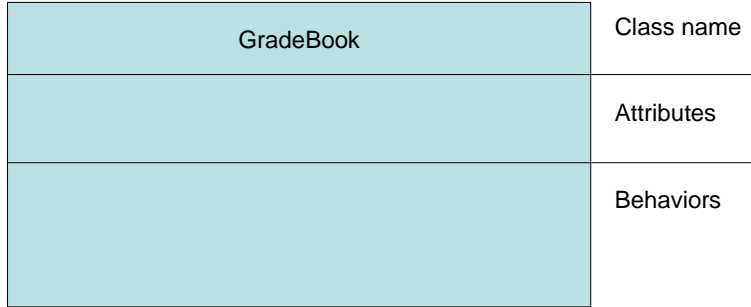
COS C 210 **LAB 2**

	<pre> System.out.printf("gradebook2 course name is: %s\n\n",cosc310GradeBook.getCourseName()); System.out.println("Please enter the course name for course 1:"); String theName = input.nextLine(); cosc210GradeBook.setCourseName(theName); System.out.println(); cosc210GradeBook.displayMessage(); } } </pre>	
	<p>24) save, compile and run GradeBookTest.java</p> <pre> java GradeBookTest gradebook1 course name is: COSC210 Java and GUI programming gradebook2 course name is: COSC310 Data Structures in Java Please enter the course name for course 1: cosc410 Welcome to the grade book for cosc410! </pre>	

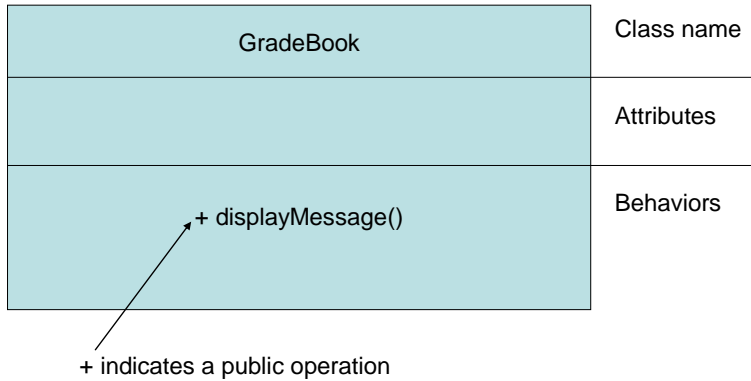
NOTES:

- declaring more than 1 public class in a single source file is a compilation error
- the type and number of parameters in the method call and the method declaration must match
- if we wanted to we could eliminate the import statement and fully qualify all references to Scanner with java.util.Scanner throughout the source code.
- Constructors cannot return values.
- If a class does not contain a constructor the instance variables are initialized to their default values.
- If a programmer declares a constructor java will not create a default Constructor for the class

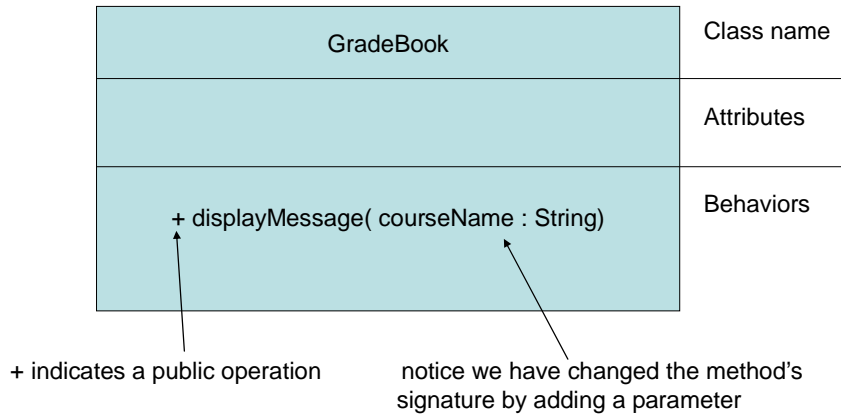
GradeBook UML diagrams



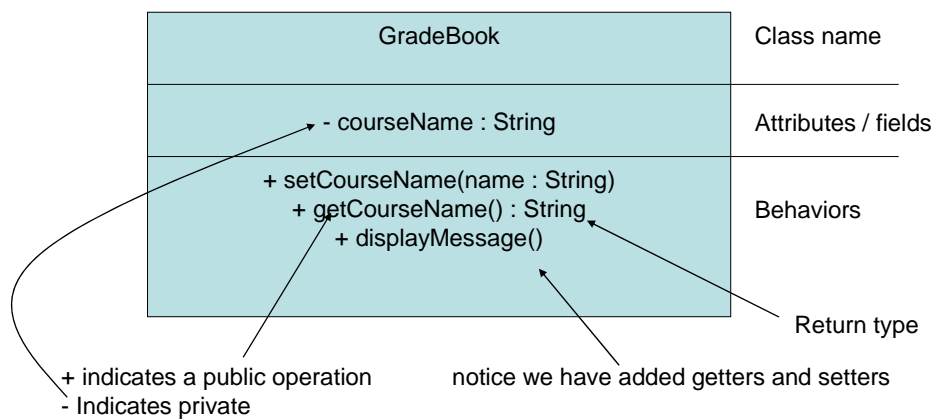
GradeBook UML diagrams



GradeBook UML diagrams



GradeBook UML diagrams



GradeBook UML diagrams

