

Project Submission Instructions

CS 2334 ~ Fall 2007

When you submit a project for grading, the following conditions must be met. ***If you fail to meet these conditions your project may not be graded and you may receive a grade of ZERO.***

1. Install the Java 2 Standard Edition 5.0 JDK Update 11+ from Sun Microsystems onto your laptop. See <http://java.sun.com/j2se/1.5.0/download.jsp>.

A set of instructions for installing Java in Windows can be found on in your first lab.

2. For this class, create a directory (folder) which contains your projects and other files from class. I recommend creating a `cs2334` folder for the class.
3. For each project, create a separate directory (folder) which contains only files which need to be handed in for the project. It is essential that you keep all the files that need to be submitted in a single directory. I recommend creating a separate folder each lab exercise and two folders for each project, one named `design1` and one named `project1`.

For each project you will make two electronic submissions. One for the detailed design and the second one for the final project. Both of these are discussed below.

Part 1: Detailed Design Submission

1. Revise your UML design from Lab. You must turn this in. Make sure to keep a copy of this.
2. Create the classes and methods specified in your design, but do not put code in the methods. Add the required documentation to your classes and methods as specified in the Documentation Requirements. This is called “stubbing” your classes and methods. A template (`ClassNameGoesHere.java`) for this file is found on the class web pages.
3. Run your “stubbed” Java files through Javadoc.
 - a) Download the `docs.opt` file from the class website and save it where your Java source code files for the project are located.
 - b) Run the following command at the command line. (First open a command windows and “cd” to the directory that contains your project files.)

```
javac @docs.opt *.java
```
 - c) This will create a set of HTML files in a directory named `javadococs` under your project directory.
 - d) Open the `index.html` file found in the `javadococs` directory in your favorite browser and examine the page for each class and inspect the contents for completeness.
4. Be sure to clearly write your name and “Project # Revised Design” on the top of the UML diagram along with your name. Where # is the number of the project, e.g. “Project 1 Revised Design”. Submit your revised UML design ***on engineering paper*** or ***printed from UML-layout software*** at the ***beginning of lab on the day specified in the project handout.***
5. Submit your “stubbed” Java files via the submit tool on `codd.cs.ou.edu`. I suggest you create a directory (folder) named `design#` to hold these source files on `codd.cs.ou.edu`, where # is the

project number, e.g. you would create a directory named *design2* for project #2. Transfer the files to be submitted to `codd.cs.ou.edu` using in the SSH Secure File Transfer Software. ***You only need to submit your .java source code files for the detailed design submission.***

Submit your project using the *submit* command on `codd.cs.ou.edu`. To do this you must log into `codd.cs.ou.edu` using the SSH Secure Shell Client software. Once you are logged into your account, use the *cd* command to change to the directory in which uploaded your project files. Finally, use the *submit* command to submit your project.

The form of the command is
`/opt/cs2334/bin/submit <course name> <assignment name> <extra files>`

For each project assignment, the command required for the submit tool including the *<course name>* and *<assignment name>* fields will be provided in the project assignment handout.

The *<course name>* field should be `cs2334-010`.

The *<assignment name>* field will be the name of the project you are submitting in the form of *project<#>-design*. For example, the *<assignment name>* field will be *project2-design* when submitting the detailed design for project #2.

Note, you must correctly specify the *<course name>* and *<assignment name>* fields in order to submit your lab or project. These fields are case sensitive.

The final field, <extra files>, contains the names of any source code files . If these files are not found in this directory, the submit tool will report an error and reject the submission.

Part 2: Final Project Submission

1. Implement the design you have developed by coding each method you have defined as well as any others you have left out of your design. As you do this make sure to modify and annotate the changes to your design on your UML and properly document all new code.
2. Test your program and fix any bugs.
3. Once you have completed the project and are ready to submit it for grading, generate Javadocs for the program again and inspect them for correctness and completeness.
4. For each project, create a text file named `COMPILATION.txt` (everything but the extension in all caps). This file should contain the exact command line you used to compile your project from the command line. If you used an Integrated Development Environment (IDE) to create your code, remember that the grader will not have access to your IDE. ***Your code needs to compile from the command line.***
5. For each project, create a text file named `EXECUTION.txt` (everything but the extension in all caps). This file should contain the exact command line you used to execute your project at the command line. If command line arguments are used, explain the command line arguments.
6. For each project, create a text file named `MILESTONES.txt` (everything but the extension in all

caps). This file will list the numbered objectives of the project, and list which objectives you believe that you have met. If you know of problems in meeting the objective, explain the problem, so that you may be considered for partial credit. A sample is given below.

Milestone #1:

I met this milestone. The test data which demonstrates this is in `milestone1.txt`

Milestone #2:

I did not meet this milestone.

Milestone #3:

I sort of met this milestone. My test data (included in `obj3.txt`) runs OK until the `remove` command is done. At this point the program gives a `NullPointerException`.

7. When you submit a project for grading, you need to transfer the files to be submitted to `codd.cs.ou.edu` using in the SSH Secure File Transfer Software. A list of the required files is given below.

- Java source files (***do not submit .class files; if you do not submit .java source files your project will not be graded and you will receive a grade of ZERO***).
- `COMPILATION.txt`
- `EXECUTION.txt`
- `OBJECTIVES.txt`
- Any test input files needed.

8. Submit your project using the `submit` command on `codd.cs.ou.edu`. To do this you must log into `codd.cs.ou.edu` using the SSH Secure Shell Client software. Once you are logged into your account, use the `cd` command to change to the directory in which uploaded your project files. Finally, use the `submit` command to submit your project.

The form of the command is

```
/opt/cs2334/bin/submit <course name> <assignment name> <extra files>
```

For each project assignment, the command required for the submit tool including the `<course name>` and `<assignment name>` fields will be provided in the project assignment handout.

The `<course name>` field should be `cs2334-010`.

The `<assignment name>` field will be the name of the project you are submitting in the form of `project<#>-final` where `<#>` is the project number. If you are submitting during the period of time allowed for Get Out of Jail Free, the `<assignment name>` field will be in the form of `project<#>-goojf`. For example if you submit Project #2 on time the `<assignment name>` field will be `project2-final`; if you submit Project #2 during the time allowed for Get Out of Jail Free, the `<assignment name>` field will be `project2-goojf`.

Note, you must correctly specify the `<course name>` and `<assignment name>` fields in order to submit your lab or project. These fields are case sensitive.

The final field, `<extra files>`, contains the names of any source code files or other files to be submitted other than the `COMPILATION.txt`, `EXECUTION.txt`, and `OBJECTIVES.txt` files. These three files are always required for projects. The submit tool will automatically retrieve these from the directory where you submit your project from. If these files are not found in this directory,

the submit tool will report an error and reject the submission.

You should NEVER submit .class files. These files will always be deleted before your project is graded.

When submitting a final project, the *submit* tool will also inform you of whether or not you have previously submitted a project using the Get Out of Jail Free provision. Once you have submitted a project using your Get Out of Jail Free, the *submit* tool will not allow you to submit another project using Get Out of Jail Free.

9. Once you have submitted your project, you should place a copy of it into a directory on your laptop that holds your submitted projects. ***Do not open, alter, edit, or execute this version of your project*** until after the grade for the project has been assigned and you are satisfied with the grade you have received.
10. Finally, you must submit your final UML design. This UML design must be annotated with any changes you have made since your submitted your “Revised UML Design”. Be sure to clearly write your name and “Project # Final Design” on the top of the UML diagram along with your name. Where # is the number of the project, e.g. “Project 1 Final Design”. Submit your final UML design ***on engineering paper or printed from UML-layout software*** at the ***beginning of class on the day specified in the project handout.***

Additional Project Submission Notes

- Make sure that you submit your Java source files (.java files) and not the java class files (.class files). If the source files are not submitted, the TAs will not be able to grade your project, and you will not receive a grade for the project.
- All projects should compile and execute successfully from the command line. ***If your projects do not compile or execute successfully, you will get a grade of ZERO on the project.*** It is wise to make sure that the program will compile and run even if it meets only a few objectives. This will make it possible for you to receive partial credit on the project.
- You are encouraged to ask the TA or instructor for help with specific problems or concepts on your project, but ***do NOT*** ask the TA to grade your project, even informally (e.g., “Is this okay?”), before you submit it. It is your responsibility to determine whether or not you have fully met the project objectives.
- On rare occasions, your project may execute differently for you than for the TA. If the TA/instructor is able to confirm the authenticity of the source files and the source files have not been modified since submission and your program correctly executes, your grade may be modified. For this purpose, ***you must follow Step 9 of these instructions.***
- It is highly recommended that before you dispute your project grade that you first check your project submission. More often than not, errors such as submission of the wrong files or making modifications to the files without testing them are to blame. ***For this purpose, you must follow Step 9 of these instructions.***
- Any questions or disagreements with the grading of projects or group work must be brought to the attention of the instructor within one week of when the item was returned to the class. ***Unless otherwise noted, lab and class meetings are NOT appropriate for discussing grading concerns. This needs to be handled during office hours.***