MAHARISHI INTERNATIONAL UNIVERSITY

CS 390
FUNDAMENTAL PROGRAMMING PRACTICES

Basic Principles and Practices Used in
Software Programming with Java Language

The Field of All Possibilities is the Source of All Solutions

Professor:

Siamak Tavakoli, PhD

December 2022

Maharishi International University is an Equal Opportunity Institution.

FUNDAMENTAL PROGRAMMING PRACTICES

"The Field of All Possibilities is the Source of All Solutions"

Basic Principles and Practices Used in Software Programming with Java Language

Siamak Tavakoli, PhD

SYLLABUS

"Know That, by knowing which, everything can be known."

—The Upanishads

"The regular practice of Transcendental Meditation is the direct way of rising to the state of transcendental Being and stabilizing it in the very nature of the mind, so that irrespective of the mind's engagements in the conflicts inherent in the diversities of life, the structure of Unity in eternal freedom is naturally maintained and life is not lost to itself"

—Maharishi's Commentary on the Bhagavad Gita

GOAL OF THE COURSE

The FPP course was created to fill gaps in the background of students when they first start their MSCS program; gaps of this kind have been classified into five areas. If you are in the FPP course, it means that the best first step you can take in this program is to strengthen your skills in these areas. The course will help you to:

- Develop skills as a Java developer [we will evaluate those skills on the Standard Exam and in labs]
- Develop facility in the object-oriented paradigm [we will evaluate this aspect of learning in the midterm, final, and quizzes]
- Understand the principles behind optimal use of data structures, together with key points about optimal implementation and use in the Java language [we will evaluate this aspect of learning in the midterm and final and in one question on the Standard Exam]
- Become skillful in using the technique of recursion [we will evaluate this skill on the final exam]
- Significantly enhance problem-solving skills [this educational outcome will be tested in labs, midterm, final, and Standard Exam]

COURSE OBJECTIVES, ACTIVITIES, AND ASSESSMENTS

This is what you'll learn to do	This is how you'll learn it	This is what will show you've learned it
Java Programming: **Tell** the computer how to distinguish between different types of data, and how to select the right instructions for a (sub) task (3,5)	**By writing** programs that use Data types: Primitive types and Object types **By writing** programs that use Flow Control: Selection, loops, and recursion	Results from Quizzes, and the Midterm examination
Organize data and functionality that belong together into distinct categories (useful for larger programs) (3,5)	**By writing** programs that use Basics of OO Programming: Objects, Classes, Inheritance and Polymorphism	Results from Quizzes, and the Midterm examination.
Create programs with Graphical User Interfaces (GUIs) (3,5)	**By writing** programs that use Swing: Components, Layout, and Event Handling	Results from Quizzes, and the Midterm examination.
Data Structures: **Organize** data into structures that are efficient for the task at hand (3,5)	**By explaining and applying** (in code) the principles behind the optimal use of data structures **By writing** implementations of: List, Stack, Queue, Trees and Hash tables.	Results from Quizzes, and the Final examination.
Science of Consciousness: **Explain** the connection between the Science of Consciousness and Programming. (2)	**By writing** appealing points (with a drawing) that have a Science of Consciousness connection.	Short Essays on Exams

*The numbers in parentheses refer to the MIU Essential Learning Outcomes that are best supported by this course objective; they appear in **boldface** in the list below. (highlight in bold those that best apply to your course objectives, activities and assessments)

1. Holistic development of consciousness and health
2. Consciousness-Based understanding (Knowledge)

3. Creative and critical thinking
4. Communication
5. Scientific and quantitative reasoning
6. Collaboration and leadership
7. Sustainable local and global citizenship

OFFICE HOURS, CONTACT INFORMATION AND BIOGRAPHICAL SKETCH

Siamak Tavakoli, PhD

Email: stavakoli@miu.edu

Office: Working from Home

Office hours: Email for appointment

Siamak is an experienced research engineer and software/hardware system analyst/developer with interests in design and development of state-of-the-art information technologies for sensor data acquisition / integration and computational efficiency problems, and in implementing robust user-oriented solutions. Due to his exposure to all stages of the software development lifecycle and hands on experience with a large variety of software technologies, he plays a major role in the architectural design as well as the implementation of the actual software solutions, test planning and documentation.

Siamak's main areas of qualification and experience include; Desktop, smartphone, and embedded system design and development, Sensor integration / data acquisition, Digital signal / image / voice processing, Non-destructive testing, Numerical Algorithm design and development, System Performance modelling, simulation and estimation, Information / distributed systems design and development, Dimensionality reduction, Computational complexity, Teaching degree / postgraduate course module, Supervising degree / postgrad./ PhD project, Lab development, lab infrastructure, system test.

EVALUATION PLAN

Grading components

Attendance and Contribution ..... 5 points
Labs ..... 30 points
Exam ..... 45 points
Project ..... 20 points
Total ..... 100 points

Note: *You can get up to 1.5% of extra credit for regular morning meditation attendance*

Meaning of grades

A (90–100)	Excellent — meets the course objectives at an exceptionally high level
B (80–89.9)	Good — meets the course objectives at the expected level
C (73–79.9)	Fair — meets the course objectives at a basic level
NC (below 73)	No credit — does not meet the course objectives

RECOMMENDED DAILY SCHEDULE

The daily schedule of all courses is designed to give students mastery of specific fields of knowledge and to cultivate higher states of consciousness for success and fulfillment in life. I recommend that you aim to be in bed by 10 PM, so that you are rested and fresh in the morning. If you have not finished your homework by then, then instead of staying up late to finish it, get a good night's rest and finish your homework in the morning before class.

MORNING	
10:00 AM – 12:00 PM	Class lecture, discussion, activities, labs
12:00 – 12:15 PM	Group meditation
12:15 – 1:15 PM	Lunch and walk
AFTERNOON	
1:15 – 3:00 PM	Continuation of morning class, projects, exercises in-class reading, labs
3:00 – 3:05 PM	Stretch break
3:05 – 3:30 PM	In-class group practice of the Transcendental Meditation program for Meditators and Rising Sidhas
	Group practice of the Transcendental Meditation and TM-Sidhi program for Citizen Sidhas and Governors
3:30 – 6:30 PM	Finish Labs, Review Course Material, Exercise, …
EVENING	
6:30 – 7:30 PM	Dinner
7:30 – 9:00 PM	Homework
9:30 PM	Rest

Eclipse Setup for FPP December 2022

This section covers the follow points:

1. Software setup
2. Configuring Eclipse.
3. Adding some Java projects to your workspace

Software Setup. For convenience in assessing your work, we ask you to use the versions of Java and Eclipse that we specify here. You will download and install these as part of your software setup.

Java. You will need to download and install Oracle jdk-16 which can be obtained by following this link:
https://www.oracle.com/java/technologies/javase-jdk16-downloads.html

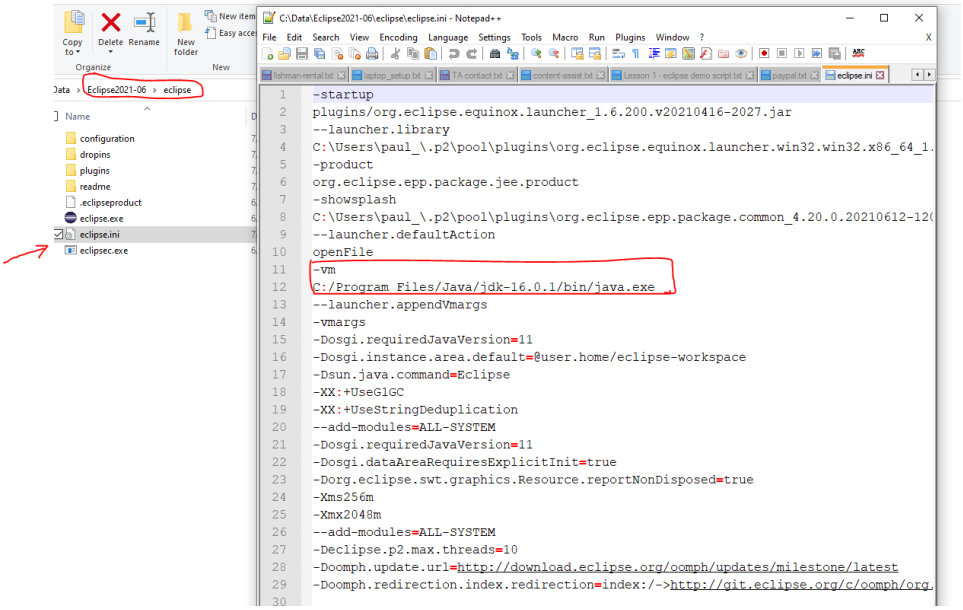
Use the Installer:

Windows x64 Installer	150.58 MB	jdk-16.0.2_windows-x64_bin.exe

Eclipse. Download Eclipse 2021-06; make use of the Installer. Go here:
https://www.eclipse.org/downloads/

Troubleshooting. When you attempt to run Eclipse, you may get an error message "Incompatible JVM". This can be solved by opening the text file eclipse.ini (in the folder where eclipse.exe is installed) and adding two lines:

-vm
C:\Program Files\Java\jdk-16.0.1\bin\java.exe

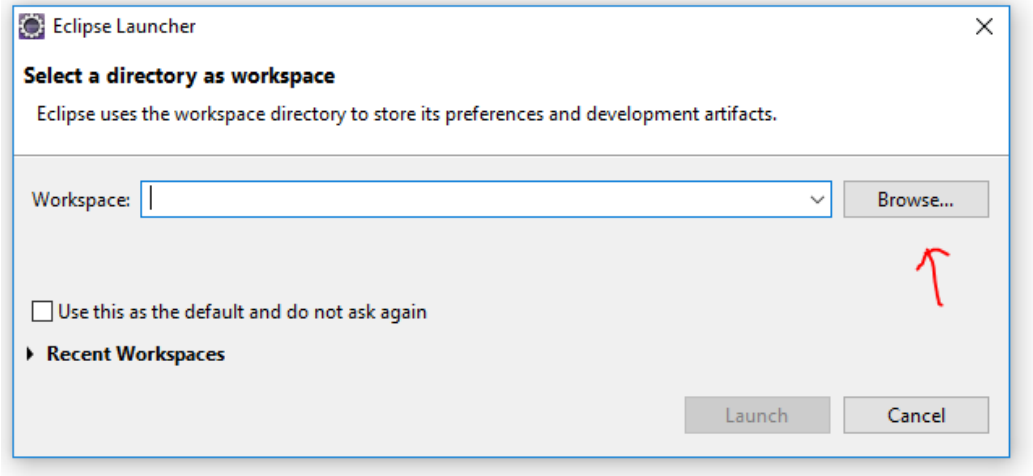


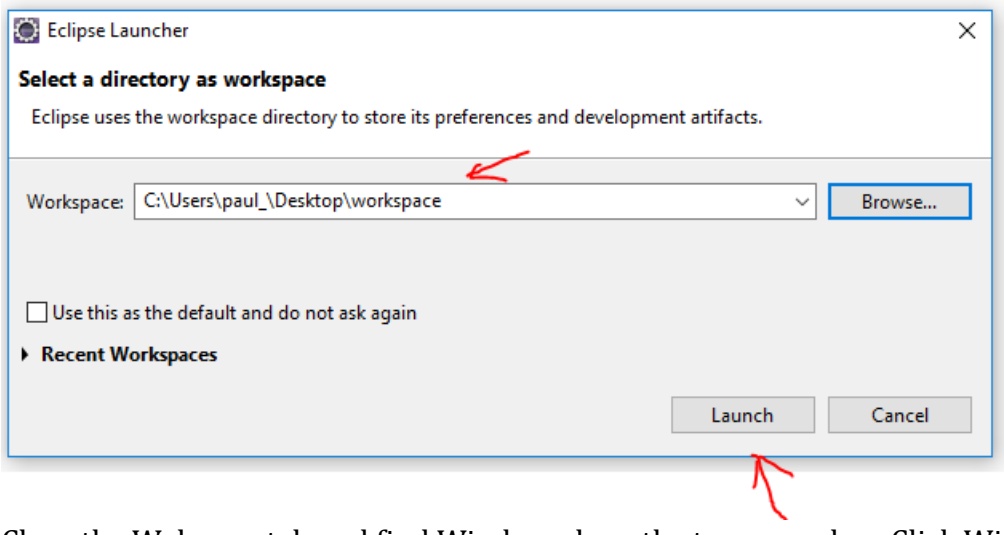
For more details about this, look at https://wiki.eclipse.org/Eclipse.ini#Specifying_the_JVM

Also: The character encoding in Eclipse needs to be changed to UTF-8:
In Eclipse, go to Windows > Preferences > General > Workspace
At the bottom left in this screen, you can change Text File Encoding to Other (and select UTF-8).

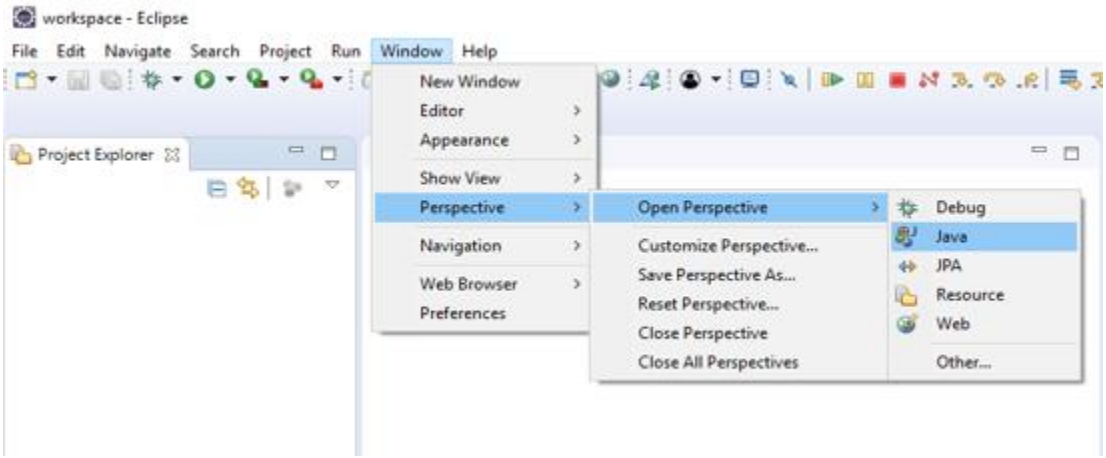
Configuring Eclipse.

1. Create a workspace that you will use for FPP. You can do this by creating a new called workspace. All the code that you write and use in the course will be in this folder.
2. Launch Eclipse; at startup, it will ask for your workspace location. Browse to the workspace folder that you just created and click the "Launch" button.



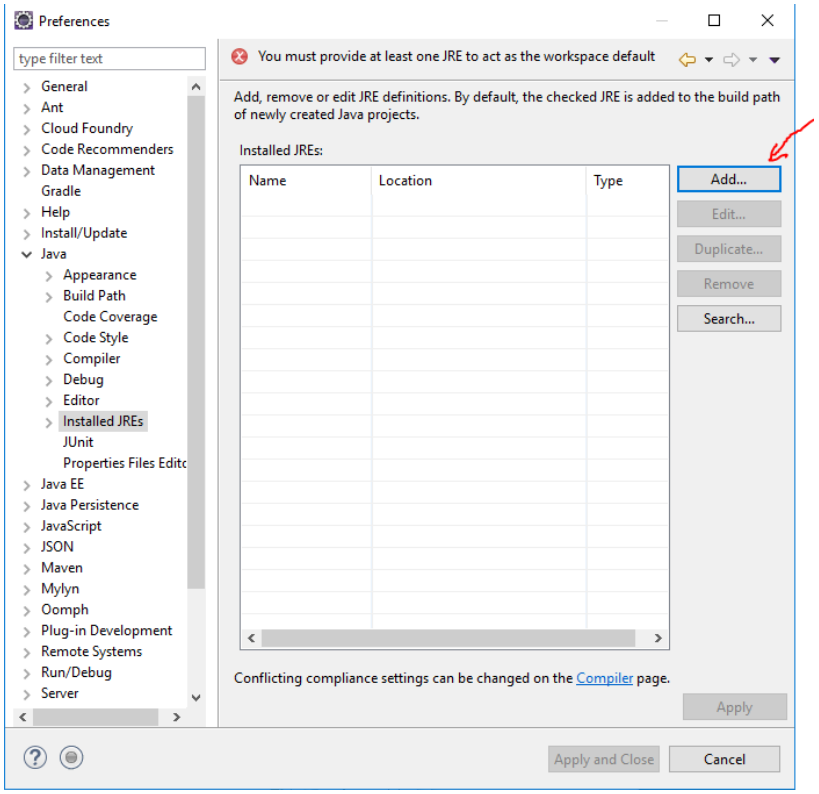


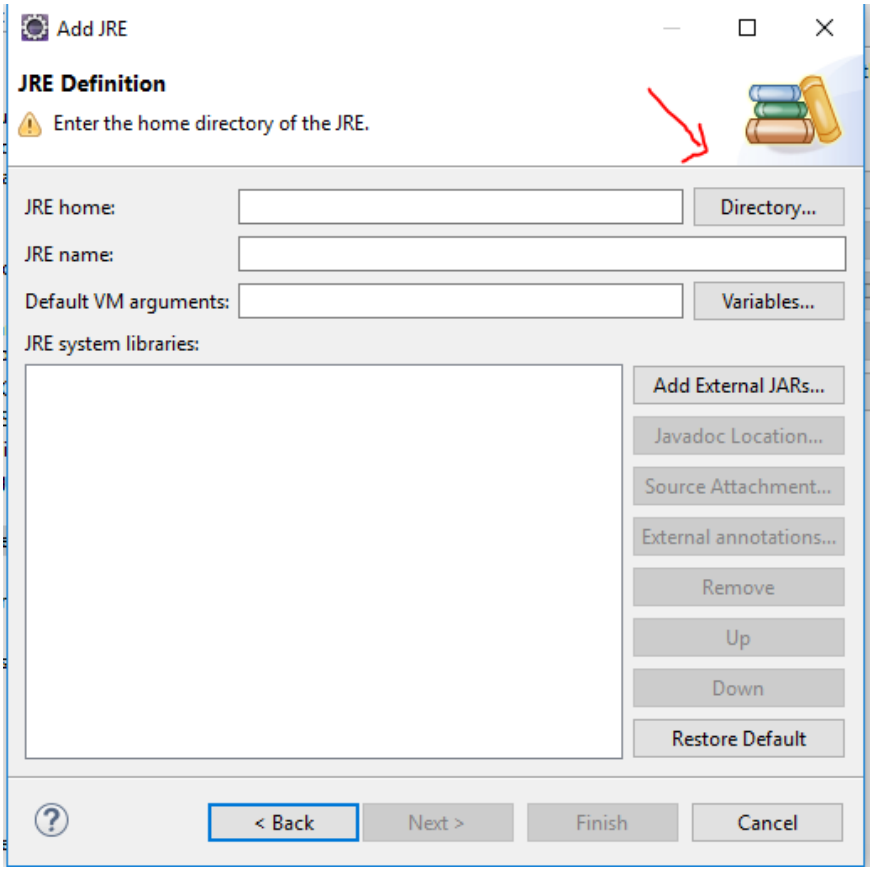
3. Close the Welcome tab and find Window along the top menu bar. Click Window > Perspective > Open Perspective > Java.

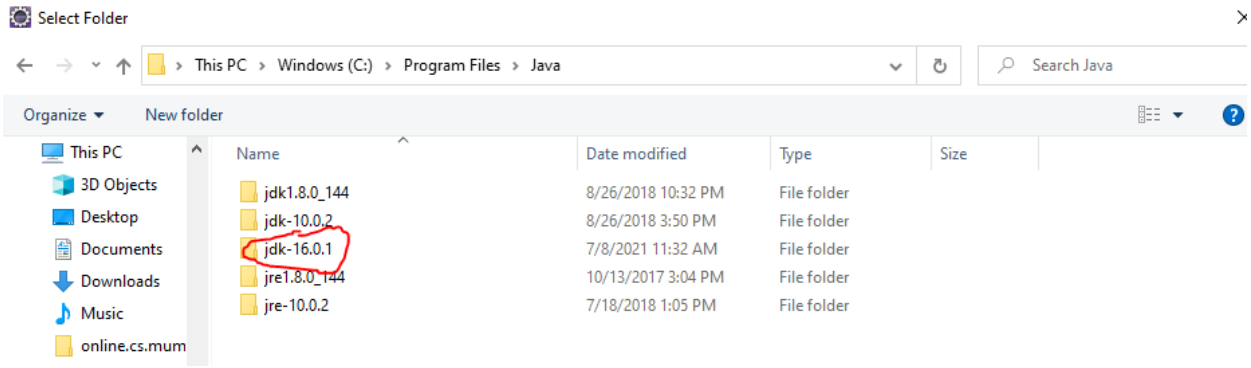


4. Point Eclipse to the jdk-16 distribution by doing the following.
   a. Go to Window > Preferences > Java > Installed JREs

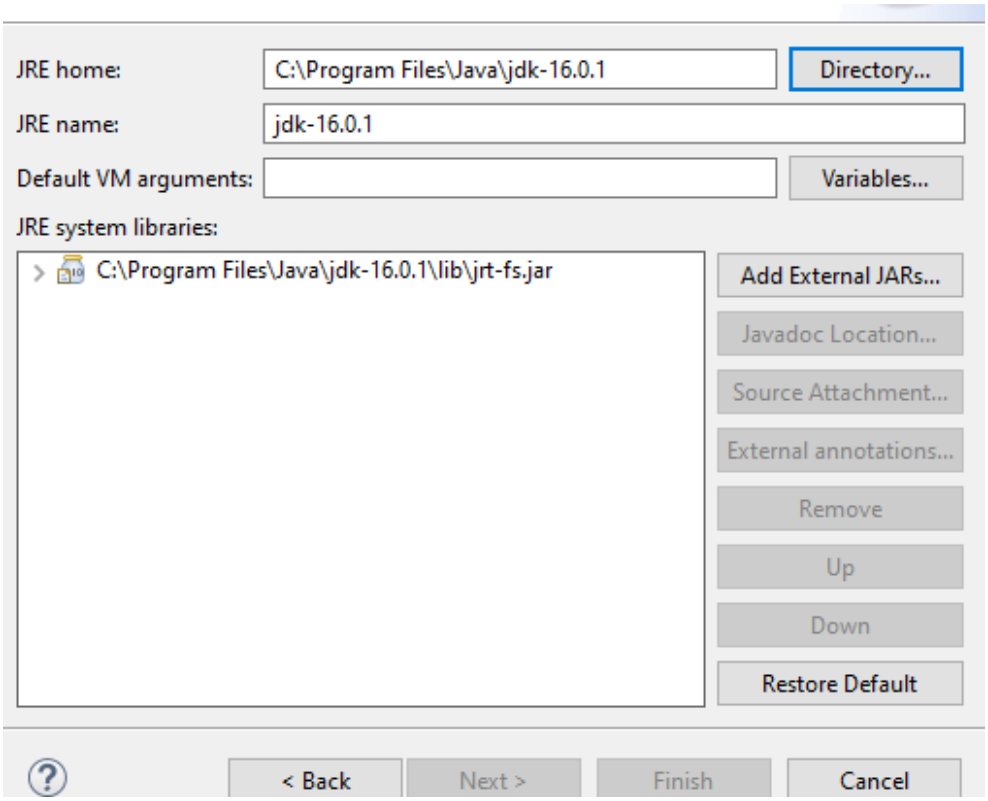
   b. If jdk-16 already appears in this window, be sure the box next to this java version is checked. Otherwise, if the jdk-16 version has not been installed, then click Add and navigate to this jdk distribution in your file system. Select Standard VM and click Next. Click the Directory button beside the JRE home field, and then navigate to your jdk distribution.



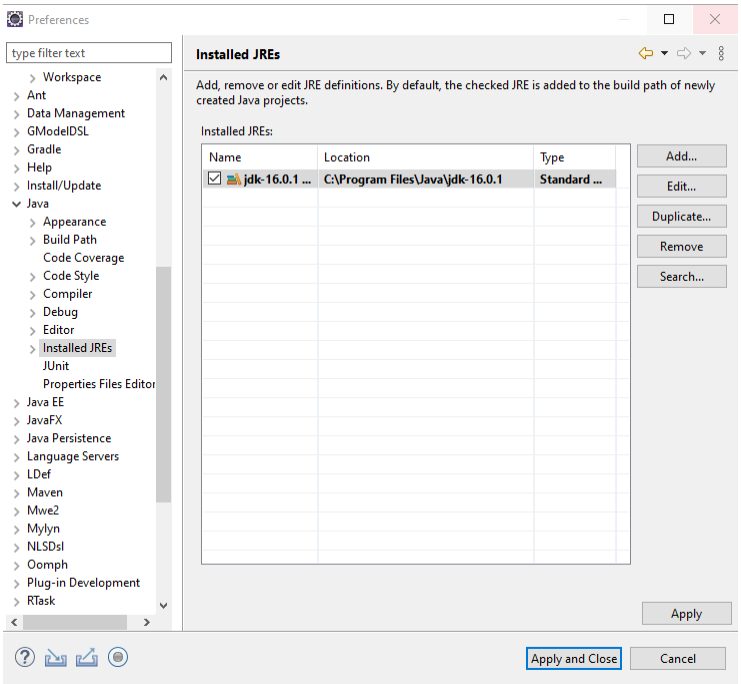




c. When you have highlighted the folder jdk-16.0.1, click OK and you will see the following window:

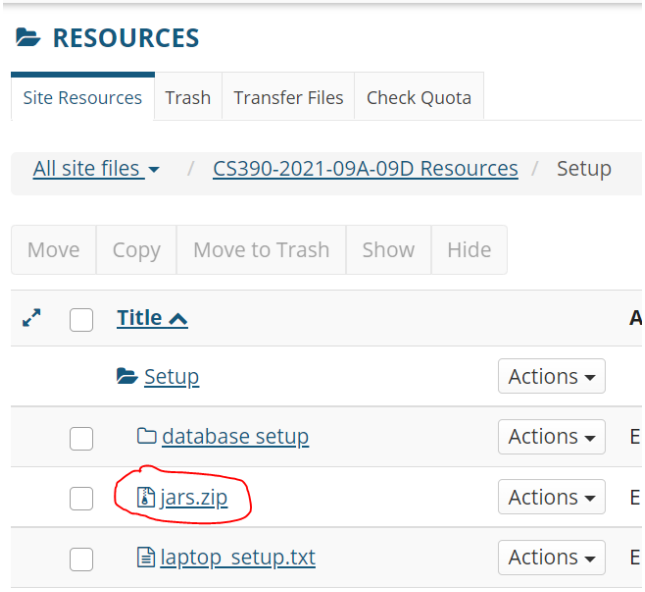


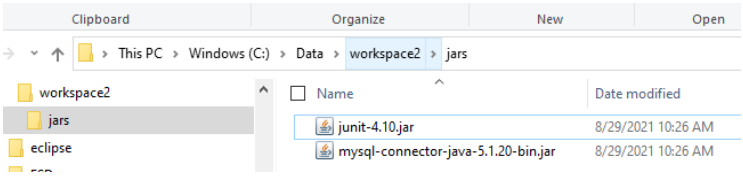
You must then check the box on the next window that asks you to specify jdk 16.0.1 as your default JRE. Then click the Apply and Close button.



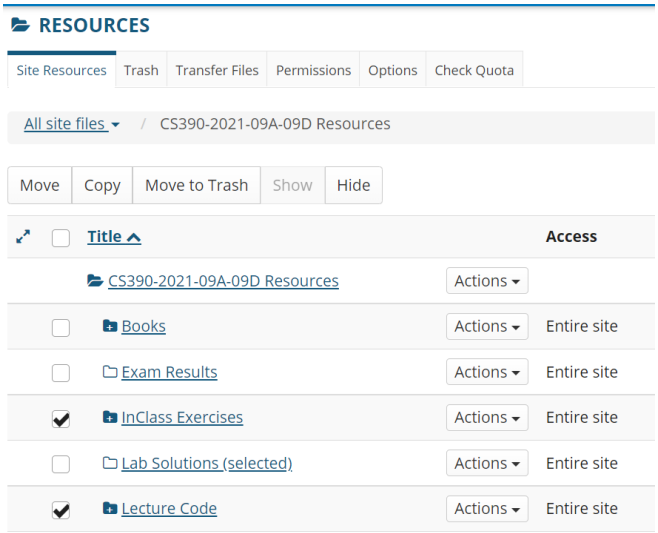
Load Java Projects

1. You will need to import two Java projects into your workspace before class begins. To make sure all the code in these projects compiles in your workspace, you will first need to download and install the jar files that will be used in the course.
2. *Jar Files.* These are in a zip file jars.zip in Resources > Setup. Download this zip file, unzip, and place the jars folder directly in your workspace.



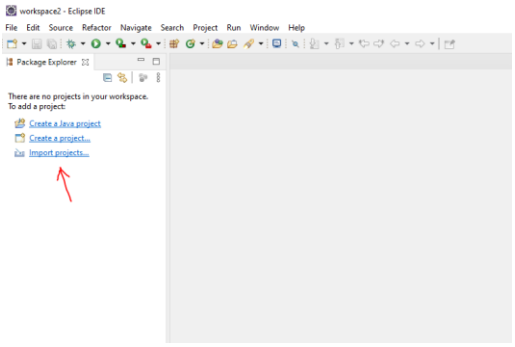


3. *Projects to download*. You will need the InClassExercises and LectureCode folders:

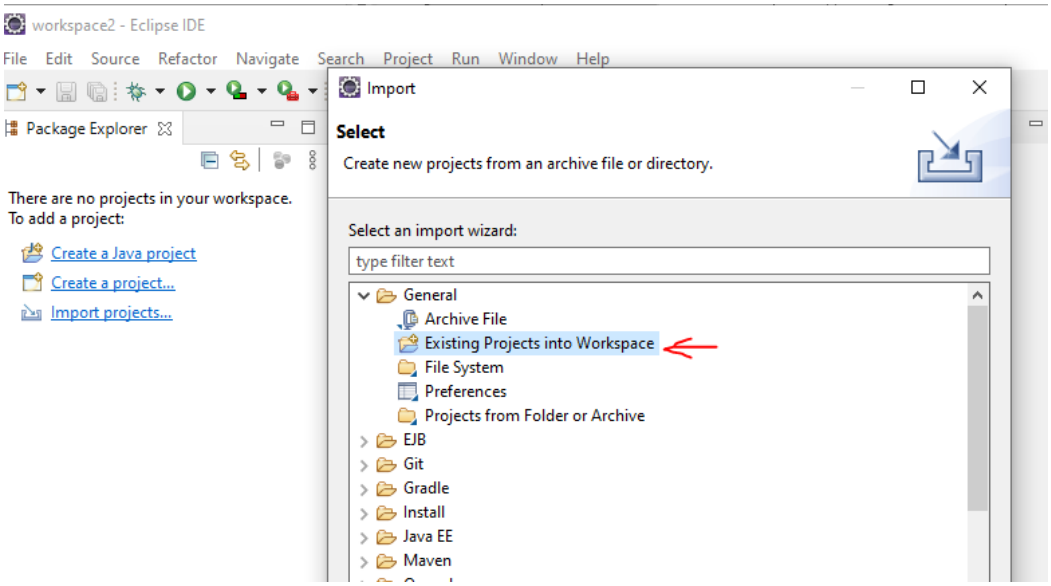


After you download the zip files, unzip and copy the folders ("InClassExercises" and "LectureCode") directly into your workspace. In the LectureCode zip file you will also see another folder "JdbcPractice" – copy this one also into your workspace. Once you have done this, you need to import all three projects into the workspace, in the following way:

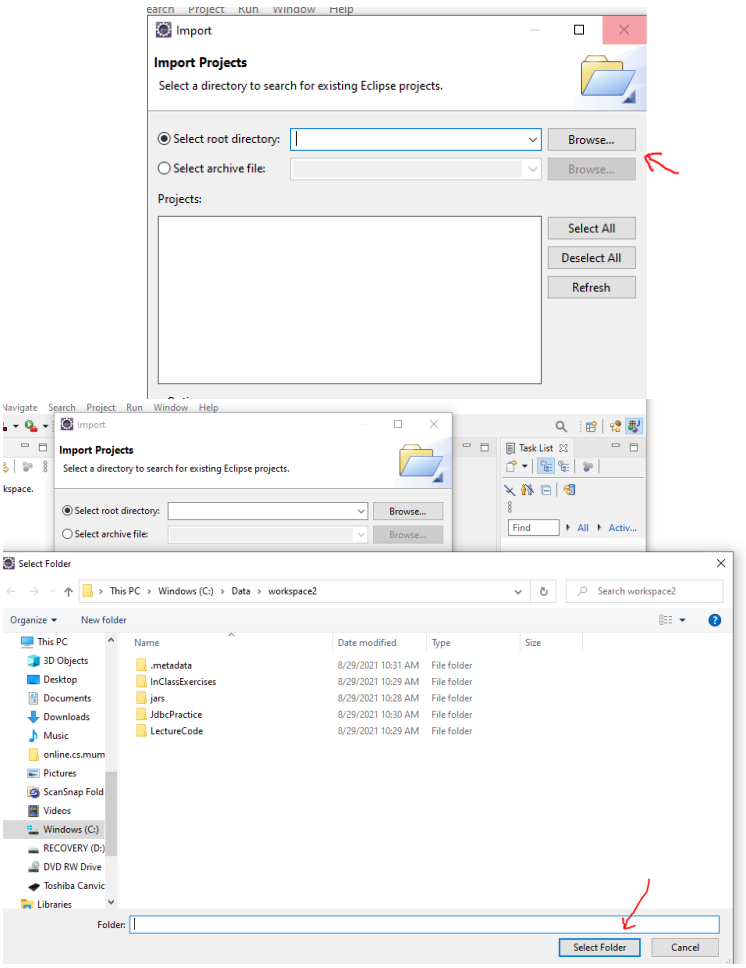
A. Either click the "import" option displayed on the left panel or else right click the left panel and select "Import".



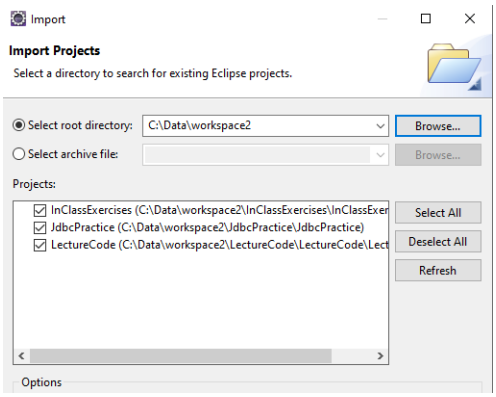
B. Select General > Existing Projects into Workspace



C. Click Browse, do not navigate – just click "Select Folder"

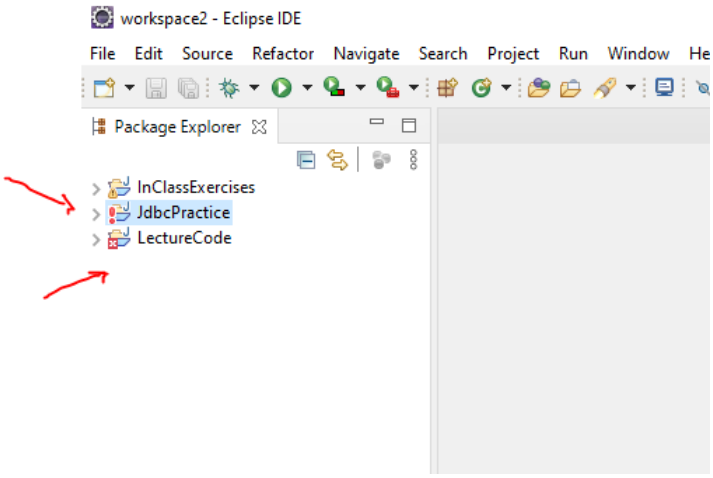


You will see:

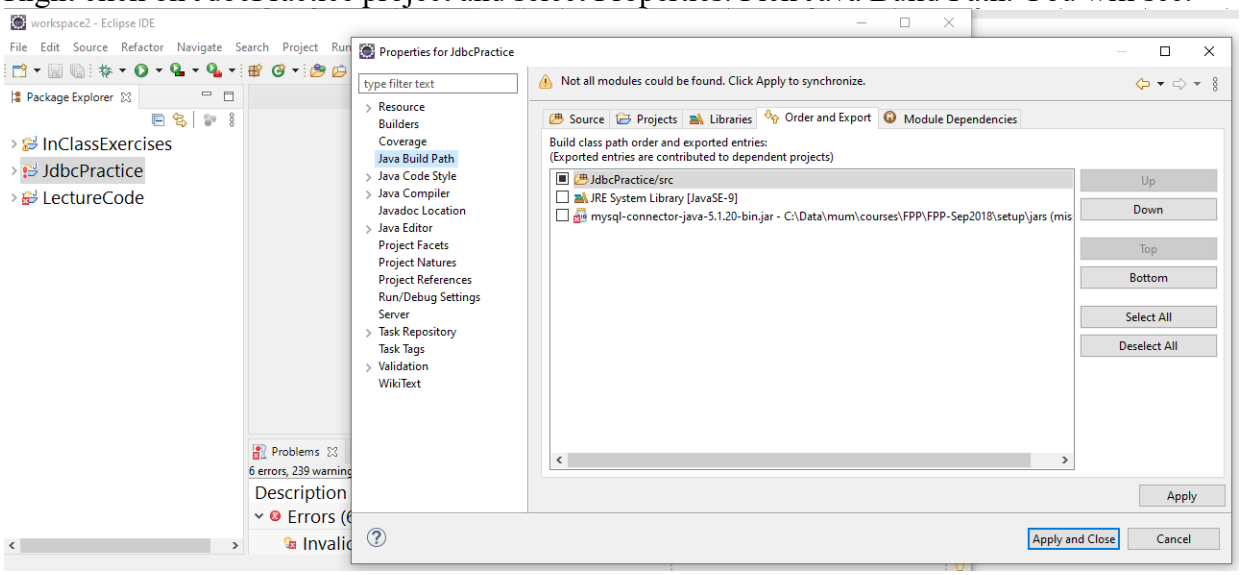


Click Finish.

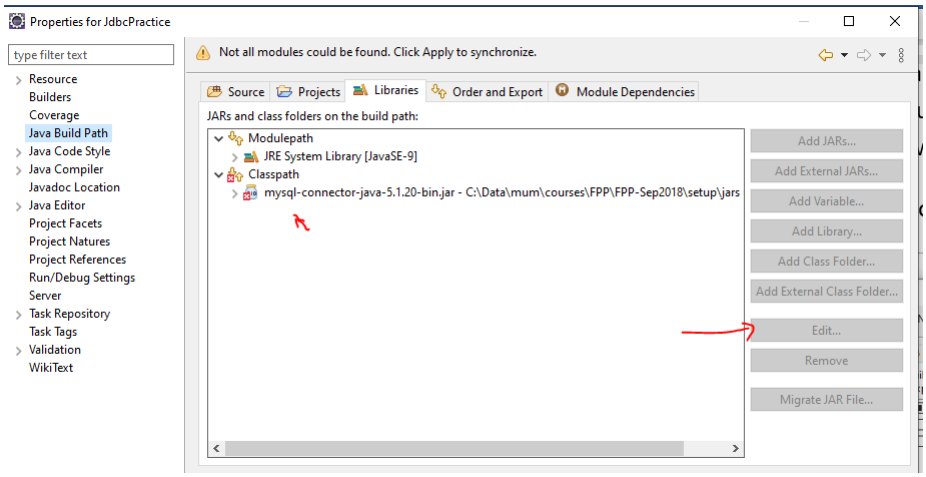
You will see something like this:



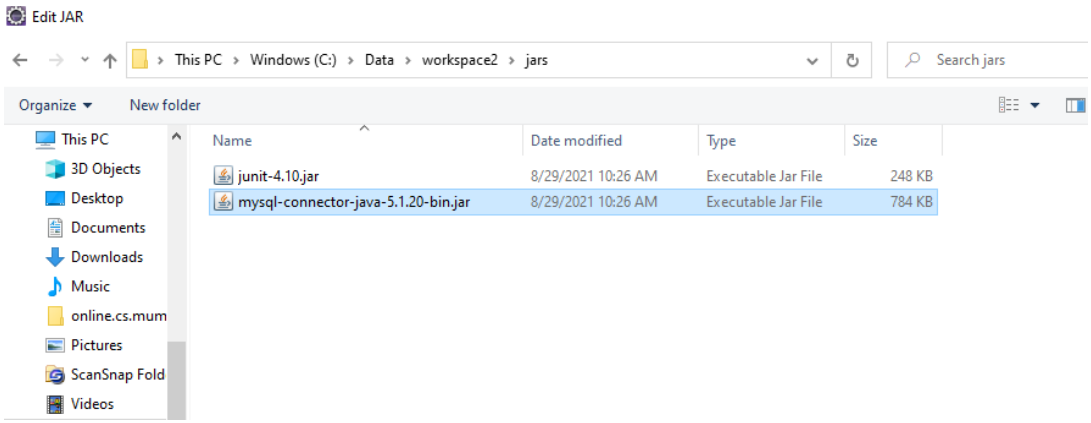
The red marking on LectureCode (if you have this) is present because the character encoding has not been corrected. See instructions for fixing this at the beginning of this document (where "troubleshooting" for Eclipse installation is discussed). The red marking on JdbcPractice indicates that you need to add a jar file to that project – we do that next.
Right click on JdbcPractice project and select Properties. Pick Java Build Path. You will see:



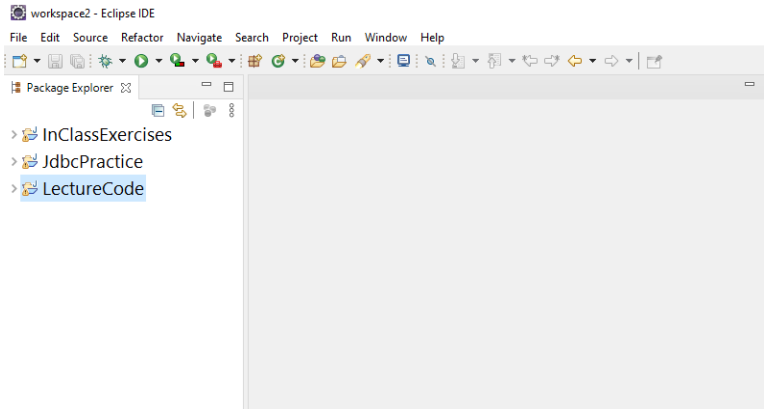
Select Libraries. You will probably see my-sql-connector marked in red. Click the Edit button at the right



Then navigate to your jars folder in the workspace and select the mysql-connector-jar file that you see there.



The red marking should disappear. Click Apply and Close. JdbcPractice project should be free of red marks.



OPTIONAL REFERENCES

Core Java 10th edition (or even better, the 11th edition) by Cay Horstmann
available through Amazon Books and Barnes and Noble (used copies are available at reasonable prices). One topic we will cover that is not in Volume 1 is Java I/O – this is covered in Chapter 2 of Volume 2; this chapter will be provided to you free of charge.

The Elements of Java™ Style by Al Vermeulen et al.
Publisher: CAMBRIDGE UNIVERSITY PRESS

ISBN 0 521 77768 2 paperback

e-ISBN 0 511 00339 0 virtual (netLibrary Edition)

Beginning Java Objects From Concepts to Code, Second Edition by JACQUIE BARKER
Publisher: Springer

ISBN: 978-1-59059-457-5

Free available at: beginning-java-objects-from-concepts-to-code.pdf (wordpress.com)

END-OF-COURSE FEEDBACK

Please give us your feedback about the course. Near the end of the course, you should be receiving an email from Mike Farrer of the Evaluations Office that gives you a one-step login link. If you do not receive this email, you can request access by emailing Mike at mfarrer@mum.edu or go to Smartevals.com/mum and log in there.

- Your Username: your student ID in 000-00-0000 format.
- Your Password: your birth date in MM/DD/YY format.

MAJOR ASSIGNMENTS

Project • 20 points • due on last day of course

The project is meant to allow students to incorporate the knowledge of software architecture design patterns into designing a software web application.

REVIEW ACTIVITY

The following activity will be used frequently at the end of a class to review the new lesson:

1. At the end of the lesson, please write down in your own words what you consider to be the most important point of the lesson. (one sentence)

2. Relate this main idea to the growth of your own creative potential—or to the knowledge of full development of consciousness that you have gained. (one sentence)

3. Draw a diagram or illustration that integrates the two points.

4. One participant: Draw your picture on the board and present your review to the large group. Others: Share your review with a neighbor.

COURSE POLICIES

The following list of policies is meant to remind you of the policies in effect for this course. Most of these are University-wide policies explained in more detail in the University catalog, available online at www.mum.edu/catalog. If you are unsure how the policy works, feel free to discuss it with me after class.

Late homework (department policy) — Unless you are ill or prevented from turning in work by a family emergency, all assignments should be handed in on the day they are due. You may turn in homework one day late for a slightly reduced grade, but not after that. Please do not turn in assignments after the end of the course without prior arrangement (see "Incompletes" below).

Punctuality and attendance — Much of the value of a university class lies in the experience you have in class. For this reason, punctuality and attendance are highly valued at M.U.M. A *class grade will be reduced at the rate of one percentage point for every 20 cumulative minutes late (up to two points per session), and three percentage points for an unexcused absence for a whole session (morning or afternoon).* This policy also applies to leaving class early.

An excused absence is defined as absence due to bona fide illness or family emergency. You are responsible for all readings and all written assignments whether you are able to attend class or not, and, in the interest of efficiency, please arrange to find out adjustments in assignments and other announcements from other classmates rather than from me if possible. I will be happy to give you any handouts you missed while absent.

Repeated unexcused absences are a violation of the M.U.M. Code of Student Behavior. In addition to academic consequences, students with repeated unexcused absences are subject to disciplinary actions.

Contact me — In the rare event you must miss class or are sick, please contact me as soon as possible using the contact information above (email ~~or phone~~) or send a message or note to class with a friend. If you keep me informed, I will know how you are doing and how to plan for each class.

Incomplete work at the end of the course. — At the end of your course your professor will evaluate the work you have turned in according to the grading or evaluation plan announced in the first few days of the course. If there was work assigned that you were not able to complete by the end of the course due to illness, family emergency, or other circumstances beyond your control, you may petition the professor to turn in that work late for credit using a "Late Work Contract". (Late Work Contract forms can be picked up at the Enrollment Center, or downloaded from the MUM website. Search "Late Work Contract.") The petition must be submitted and approved before the end of the course, and then, if granted, the missing work should be submitted by the first Monday following the last day of the block, unless you are ill over the break between blocks. If you are ill over the break, your professor may allow you up to 32 days to complete the missing work. (Note: This option for submitting late work does not apply in the case of students who miss more than six sessions of a four-week course, nor is it to be used in the case of students who do not organize to get assigned work done according to the plan outlined in the syllabus. Students attending a course that ends at the end of a semester, students on Warning or Probation status, and Distance Education students may have different late work submission deadline requirements. Review the back of the Contract Form or contact the Registrar for more details.)