

CIS 2430 A2 README/Checklist

Name:	Javaughn Burke
UoG Email Address:	javaughn@uoguelph.ca
Partner's name (if applicable)	
Percentage of your submission that is taken from the starter code provided (approximately)	50%
How to run your application: provide the exact string to type to compile it, and run it. Must be runnable on the command line.	make
Notes for TAs (anything special we should know when grading your assignment)	Compiled with mac and and the program ran fine

Learning Outcomes	3 examples from your code. File name, line number
refactor and restructure class design for improved encapsulation, modularity, cohesion and coupling	
demonstrate use of inheritance through super/sub classes as well as through the use of interfaces	All frame panels like AdminMode, Planner and Student calls super class JFrame. BCG.java – meets requirements – line 103 uses rqrdrNumberOfCredits from general degree class SelectDegree.java – selectdegree – line 27
demonstrate clear understanding event driven programming through well designed listeners and gui components	MenuItem listeners from Planner.java AdminMode.java - listeners for buttons for change courses 113 SelectDegree.java – listeners for buttons
demonstrate service-based error handling through a rich set of exception classes that communicate specific errors to client classes	Planner.java – login button listener throws a IOException if login info not found. Planner.java – signup button throws an exception if student didn't save student.
create a repeatable testing suite and justify the choice of test cases	
design and create a graphical user interface that is learnable and usable	Planner.java – Gui AdminMode.java – Gui SelectDegree.java – Gui StudentPlanner.java – Gui
use inner classes, anonymous classes, and/or lambdas effectively	SelectDegree – SwitchStudPlan – line 65 AdminMode – ActionListener – line 79 Planner.java – CustomActionListener – line 139

Required elements	Examples from your code (File name, line number) – more than one example preferred
Exceptions and try/catch loops	Planner.java – CustomActionListener – line 155 Planner.java -CustomActionListener – line 173
Error prevention/handling (might also be try/catch or might be input checking)	Planner.java – LoginActionListener – line 123 BCG.java – isDigit – line 22 AdminMode.java – AdminActionActionListener – line 103
Two different layout managers	StudentPlanner.java – StudentPlanner – line 76 AdminMode.java – AdminMode - line 37
Separate window/panel for administration	Run Planner.java click on switch to admin menu item the click on switch to admin mode AdminMode frame should pop up.
Listeners	Planner.java – LoggingActionListener – line 120 Planner.java – SwitchAdd – line 110 AdminMode.java - AdminActionListener - line 79
Course class refactored and immutable	Course.java – setCourseCode – line 43 Course.java – setsemesterOffered – line 49 Course.java – toString – line 146
Attempt class created	
Classes in package	Course and Course Catalog in univ package and all the database files in DBaseConnector package.
Refactor Plan of Study (include how/where you provided the functionality if you eliminated POS)	BCG.java check number of credits remaining for degree – line 111 BCG.java remainingRequired courses returns a list of the required courses remaining – line 155
Database usage	AdminMode.java – Create connection listener connects to the database – line 93 Planner.java – Login button listener loads the student from the database - line 154
Javadoc comments (the most complete examples)	
Evidence of testing	

INSTRUCTIONS FOR PARTNERS

Partners must each complete this sheet and both partners must submit. If you do not submit you will not get a grade for A2. Complete this sheet in such a way as to illustrated YOUR contribution to this assignment.