MODULE 5 – ECLIPSE

ITSE 1003

Introduction to Programming Languages

Goals

- Introduce Integrated Development Environments
- Focus on Eclipse
- Play!

Integrated Development Environments

- IDE
  - Application targeted at software developers
- Typical features
  - Source code editor (often multiple languages)
    - Intelligent code completion features
  - Build automation tools
  - Compiler, interpreter, or both
  - Debugger
  - Version control system
  - Class browser, an object browser, and a class hierarchy diagram
  - Integration with other tools

Integrated Development Environments

- Most popular Java IDEs
  - NetBeans
    - Oracle
    - Free & open source
    - Simple, yet powerful
  - Eclipse
    - Eclipse Foundation
    - Free & open source (Java)
    - Powerful, somewhat harder to use than NetBeans
  - IntelliJ IDEA
    - JetBrains
    - Proprietary and not free
    - Probably the most powerful Java IDE

Eclipse

- Appears to be dominant in the industry
  - We will use Eclipse in the Java series at ACC
- Installed on lab machines already
  - Shortcut on desktop
- Available for download on personal laptops
  - https://www.eclipse.org/downloads/
- Instructor has EE versions for Mac and Windows

- User guide
  - http://help.eclipse.org/kepler/index.jsp

Eclipse concepts

- Workbench
  - Desktop development environment in a window
  - Multiple can exist on the desktop at any given time
  - Provides a common paradigm for the creation, management, and navigation of **workspace resources**
  - Contains one or more **perspectives**
- Workspace
  - Holds all resources for activities targeting a specific task
    - Equivalent to a folder or directory on a file system
  - Workbench in effect a tool to navigate and manipulate a workspace
  - Can have many, but only one active in Workbench
- When you start Eclipse
  - Dialog allows you to select where the workspace location
  - On lab machines, ***put workspace on USB drive***
  - On laptops, can use default or any other location

Eclipse concepts

- Perspective
  - Contains **views** and **editors** targeting
    - A specific type of task
    - Specific types of resources
  - Controls what appears in certain menus and tool bars
  - Examples:
    - Java perspective combines views for editing java source files
    - Debug perspective contains views for debugging java programs

Go live

- Start Eclipse, choose workspace
- Quick look at Workbench
  - Perspectives
  - Views
- Create
  - Project (investigate project explorer view)
  - Class with main()
    - Code completion
    - Error detection
- Run; show console
- Debug
  - Set breakpoint
  - Step
  - Inspection

- Miscellaneous
  - Formatting
  - Errors/warnings
  - “Not saved” indicator
  - Navigate / open type
  - Open / close project
  - Switch perspectives
  - Open perspective/view
  - Properties

Eclipse concepts

- Resources
  - Projects, folders, and files in the workbench
- Files
  - Equivalent to files in a file system
  - Source, executable, configuration, …
- Folders
  - Contain files and other folders
- Projects
  - Resources for a particular “project”
  - Equivalent to directories or folders in a file system
  - Contain files and folders
  - Support builds, version management, sharing, and resource organization
  - Can have many active in a workbench

Go live

- Create
  - Project (investigate project explorer view)
  - Class with main()
    - Code completion
    - Error detection
- Run; show console
- Debug
  - Set breakpoint
  - Step
  - Inspection

- Miscellaneous
  - Formatting
  - Errors/warnings
  - “Not saved” indicator
  - Navigate / open type
  - Open / close project
  - Switch perspectives
  - Open perspective/view
  - Properties

Activity: Play!

- Move your programs to Eclipse
  - Create project
  - Copy (by file or content)
  - Run
  - Force errors
  - Set breakpoints and debug