

RobotBuilder / Command Based Programming

Presenter: Corey Applegate (3244) Lead Programming Mentor

- **Additional setup in Eclipse**
 - Set Refresh using native hooks
 - Windows > Preferences > General > Workspace
- **RobotBuilder**
 - Setup Eclipse workspace
 - Divide Robot into subsystems (DriveTrain, Wrist, Claw,...)
 - Add actuators and sensors (Analog Gyros, DIO, Analog IO,..)
 - Add Operator Interfaces (Joysticks)
 - Add Commands
 - Save All work in Eclipse before refreshing changes in Robotbuilder. You could lose all work since last save. **It will happen*
- **Eclipse**
 - Import project into workspace
 - **Do not** make edits or add your code to areas commented with

```
// BEGIN AUTOGENERATED CODE .....  
...  
// END AUTOGENERATED CODE .....
```

These areas are updated by RobotBuilder. You can delete these comment headers and footers to regain ownership to the data in eclipse but any changes will no longer apply from RobotBuilder
- **RobotBuilder Can Not**
 - Program you robot. It only creates the skeleton
 - Add some third party devices.
 - Follow the layout RobotBuilder created by manually adding IO to RobotMap *Outside of special //BEGIN //END sections.
 - Nothing replaces good Naming and Code Comments. We all get lazy, **Don't Forget** to make comments in eclipse.
 - Use consistent naming so the files sort automatically. I like to add to the start of methods we create "my_" so intellisense jumps to My methods when I start typing.
 - I also like to add the subsystem name to the start of all commands so they group together in eclipse. * **Drive_With_Joy**, **Drive_Auto_Distance**, **Drive_Track_Target**, **Drive_Turn_To_Heading...**
- **Online Resources**
 - RobotBuilder <https://wpilib.screenstepslive.com/s/4485>
 - Tools / RobotBuilder
 - Command Base Programming
 - <http://wpilib.screenstepslive.com/s/currentCS/m/java/c/88893>
 - You tube Videos
 - [FRC 2013 RobotBuilder Brad Miller](#)