Jets Rubric

Criteria	1 – Below	2 – Approaching	3 - Meeting	4 - Exceeding
Single Responsibility Principle	• Classes are extremely entangled • Multiple classes with duplicative behavior are present • Code is highly repetitive	• Most objects are disentangled, however game logic still presents some entanglement • No repetitive classes are present	• Classes are disentangled and reusable	• Game logic is contained within multiple specific worker methods and initiated by a coordinator method
Encapsulation	• Class field access modifiers are left public or default • Constructors are not declared public • Classes are not declared public • Methods are left at default access	• Some fields are made private, others are left default or public • Private methods are accessed via 'getters and setters'	• All class fields are private and accessed only through 'getters and setters' • Constructors, classes and methods are declared public	• Helper methods are kept private as they have no need to be accessed outside of the class
Object Oriented Programming	• Code is largely procedural • Existing classes do not follow Object Oriented principles • Local variables and collections are assigned to objects	• Classes exist which adhere to the Single Responsibility principle • Some procedural code exists, but is contained to bloated class methods	• Classes all adhere to Single Responsibility • Objects are instantiated and passed to other objects for modification/use • Objects are used to store and manipulate data, rather than local collections • Procedural programming is absent	• Polymorphism is utilized to represent like objects
Java Language	• Methods or variables are	• ArrayList is used for collections of objects	• Methods are well named	• Code is well commented • Method and variable

	confusingly named or contain inappropriate language ● Iteration over arrays excludes the use of the foreach loop, always defaulting to traditional for loops ● Method parameters are confusingly named ● Overwritten default constructors are not replaced ● Class names are plural	within classes ● Method parameters are well named ● Class names are singular	● Foreach loop is correctly utilized when iterating over arrays and ArrayLists ● Default constructors are replaced when overridden ● Replaced default constructors use this() to call overloaded constructors	names create near human readable code
Jets	● Program doesn't function properly	● Some of the menu functionality has been implemented	● All menu options function properly	● Pilots are assigned to a Jet