**NAME**:Ben Trane  
**DATE:** February 14rd, 2017  
**Week: 7**

**Code Topic:** MVC  
**Description of Understanding:**.   
MVC Pattern stands for Model-View-Controller Pattern. This pattern is used to separate application's concerns.

* **Model** - Model represents an object or JAVA POJO carrying data. It can also have logic to update controller if its data changes. Creates the class of sport and the three options VolleyBall,FootBall,Basketball, Creates the add and drop sports, and makes the map
* **View** - View represents the visualization of the data that model contains. Shows the outputs, and calls the functions and the outputs for each function.
* **Controller** - Controller acts on both model and view. It controls the data flow into model object and updates the view whenever data changes. It keeps view and model separate. Creates the functions, and takes the inputs and does all the input and outputs. You input what sports you want to add or drop and it kicks it to the view to present the data.

|  |  |  |  |
| --- | --- | --- | --- |
| FILE | GIT HUB LINK | What should I be looking for | CODE: sandbox or mine |
| view | https://github.com/traneb/cit360/compare/master...MVC-1?expand=1 | Shows the view | Mine |
| controller | https://github.com/traneb/cit360/compare/master...MVC-1?expand=1 | Shows the controller | Mine |
| model | https://github.com/traneb/cit360/compare/master...MVC-1?expand=1 | Shows the model | mine |
| Use case | <https://github.com/traneb/cit360/compare/master...MVC-1?expand=1> | Use case diagram | Mine |
| State diagram | <https://github.com/traneb/cit360/compare/master...MVC-1?expand=1> | State diagram | mine |