Date: 28 March

Create a game Class which captures all the info for each game

* Event type, event ID, new Event()
* Competitors, new Competitors()
* Official , new Official()
* Winner prediciton (new Prediction()? print competitors, get selection, assign prediction to competitor.
* Run game and summarise Outcome of games (i.e result of compete()) (Official does this)(define run game method(s) – calls compete())
* Assigns points back to athletes. (Offical does this)(define points assignment method)

Create an empty array of game()[50?]

When new game is created (option 1 of menu), use for loop to find null game - assign new game to that index.

e.g. (i=0; i<game.length(),i++)

if (game[i]==null){

game[i]= new Game ();

break;

}

Athletes need to have a “points” variable. – with a setter to add points cumulatively.

Therefore Participant Data can’t be final. (need to update points) – but must be static.

Winner prediction is option 2 on menu. – if new Game() not yet created, need to send back to menu with message to select a game.

If new Game does exists (calls the predict winner() method in the current game)

To get results of all games, I need to loop through the games array in the SAME CLASS that is was created. (Is this the main class? Or the menu class?)

for( int i = 0, i<game.length(), i++)

if game[i]!=null

game[i].outcome of games()

if game is not yet created, need to send back to menu with message to select a game.

To get points of all atheletes

Run through loop of athletes with a getPoints() method.

runGame() method {

int j = 0 (outside method)

for (int i=0; i<participantData.length(); i++)

if (paricipant[i].getID ()== competitor[j].getAthleteID)

participant[i].compete(); (how do I do this for SuperAthletes?)

participant[i].getResult();

comptetitor[j].setResult();

j++;

}

Need a setResult method in game class – where participant[i].getResult() = competitor[j].setResult

e.g. public int setResult(){

int j = 0 (outside method)

for (int i=0; i<participantData.length(); i++)

if (paricipant[i].getID ()== competitor[j].getAthleteID)

return participant[i].getResult();

j++;

}