CIT 383 – Administrative Scripting
Project 2

Problem
You are to create a program that will record and display the time results of a tri-athalon for any number of given athletes. For this tri-athalon, there will be 3 events: Swim, Run, and Bike. For each athlete, you will record their completion time for each event. At the end, you will then calculate the total completion time across all 3 events for each athlete. Then you will calculate the average time for each event as well as the total completion time across all athletes. At a minimum, store the athlete's time for each event into an array.

Step 1: (5 Points) Acquire data from user
Prompt the user for the number of athletes that will be competing in the tri-athalon. This number will be the size of the arrays.

Step 2: (60 points) Populate event completion times
Populate the arrays by using the random number generator.
You will populate each athlete's completion time in whole **minutes**.
Use the following chart. Note: the minimum and maximum times are inclusive.

Event	**Minimum Time in Minutes**	**Maximum Time in Minutes**
Swim	30	89
Run	90	119
Bike	120	360

Step 3: (5 Points) Calculate Total Completion Time
For each athlete, you are to calculate their total completion time across all three events.

Step 4: (10 Points) Calculate Averages
For each event and for the total completion time, you are to calculate the average in minutes. However, you are to **round to the nearest minute**.

Step 5: (20 Points) Display Results
You will display the results in a tabular format. For the completion times, you are to display the time in the following format: **<hour>:<minutes> (<total minutes>)**
For example: If an athlete completed his time in 72 minutes, the displayed result will be 1:12 (72). It is okay if trailing zeros do not show up.
See the sample run for example

Step 6: Save and Execute file
1. Save your Ruby file, using the file name listed below:
   Project2.rb
2. Upload to linux server (if needed)
3. Execute your ruby script either explicitly or implicitly.

See sample run below

Bonus: (10 points) Mult-Dimensional Array
You will receive 10 additional bonus points if you successfully implement the solution in a 2 dimensional array.

Helpful Hints
1. To hold the completion times of each athlete, create a separate array variable for each event and total time.

2. When using the rand() function, you can use ranges as parameters. For example rand(1..10) returns numbers between 1 and 10. Be careful to use two dots (..) not three dots (…). Two dots are inclusive, three dots the first number is inclusive while the second number is exclusive.
3. For a variable that is a float data type, you can use the .round(<number of decimal points>) function.
4. Use the / and % operators when converting total minutes to <hour>:<minute> format.
5. Instead of trying to code everything at once, pick a single event and code for that single event all the way through to the end. Once you have it working, then code for the other events.

Sample Run