Date: 14-3-2017

Author: Kien Nguyen T.

I. Teaching

- Arrays

- Hashes

- Strings

II. Exercise

Ex1. Create schedule:

Use *football* database from Day\_04 exercise to do:

* Add number of team (Multiples of 4).
* Add team's name
* Choose seed teams
* Create groups (4 teams in 1 group): at least 1 seed team and 1 normal team.
* Create schedule
* Input score
* Display teams OF Semi final round.
* Create schedule for Semi final round.

Ex2.

**Description:**

We want to generate a function that computes the series starting from 0 and ending until the given number following the sequence:

0 1 3 6 10 15 21 28 36 45 55 ....

which is created by

0, 0+1, 0+1+2, 0+1+2+3, 0+1+2+3+4, 0+1+2+3+4+5, 0+1+2+3+4+5+6, 0+1+2+3+4+5+6+7 etc..

**Input:**

LastNumber

**Output:**

series and result

## **Example:**

**Input:**

> 6

**Output:**

0+1+2+3+4+5+6 = 21

**Input:**

> -15

**Output:**

-15<0

**Input:**

> 0

**Output:**

0=0

Ex3.

### Description:

Given an array of numbers return an array of numbers from the array that qualify as perfect squares. A perfect square is defined as a whole number that, when square rooted, is a whole number. (Such as 1, 4, 9, 16, etc, etc.)

Note: Return only one copy of each perfect square in ascending order

get\_squares(1..16) # => [1, 4, 9, 16]  
get\_squares(1..100) # => [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

**Ex4. Write daily Report today.**