Ruby Exercise

**1) Print the sum of elements in array - [21,3,43,5454,23,1233]**

**2) Print the sum of values in hash - {'a'=>13223, 'b'=>23232}**

**3) Sort and print the elements in the array - [21,3,43,5454,23,1233]**

**4) Write a method to generate sum of the two largest number on a given array.**

[1,5,2,7,3,9]

#=>16

**5) Write a method to print out the square of each elements on an array**

Example:

square([1, 2, 3, 4, 5, 6])

#=> [1, 4, 9, 16, 25, 36]

**6) FizzBuzz Problem Write a method take an input (integer) and print out Fizz Buzz.**

For example

fizz\_buzz(5)

#=> 1, 2, Fizz, 4, Buzz

fizz\_buzz(16)

# => 1, 2, Fizz, 4, Buzz, Fizz, 7, 8, Fizz, Buzz, 11, Fizz, 13, 14, Fizz Buzz, 16

fizz\_buzz(36)

# => 1, 2, Fizz, 4, Buzz, Fizz, 7, 8, Fizz, Buzz, 11, Fizz, 13, 14, Fizz Buzz, 16, 17, Fizz, 19, Buzz, Fizz, 22, 23, Fizz, Buzz, 26, Fizz, 28, 29, Fizz Buzz, 31, 32, Fizz, 34, Buzz, Fizz

**7) Write a function to generate unique elements count in an array.**

Example:

array\_ids = ["54af642897722e4ccd000002","54af68ab97722e6a57000003", "54c0f02997722e5cc5000003", "54c0f02a97722e5cc5000013", "56cb1f6697722e219a000002", "54af68ab97722e6a57000003", "54af642897722e4ccd000002", "56cb1f6897722e219a000013"]

#=> uniq\_ids(array\_ids)

{

"54af642897722e4ccd000002" => 2,

"54af68ab97722e6a57000003" => 2,

"54c0f02997722e5cc5000003" => 1,

"54c0f02a97722e5cc5000013" => 1,

"56cb1f6697722e219a000002" => 1,

"56cb1f6897722e219a000013" => 1

}

**8) Write a method that takes collection of strings on an array and convert them into hash with string as key and length of string as value.**

Example

[“aeroplane”, “bird”, “superman”, “desktop”, “documentation”, “board”]

#=>

{

“aeroplane” => 9,

“bird” => 4,

“superman” => 8,

“desktop” => 7,

“documentation” => 13,

“board” => 5

}

**9) Create a class 'Student' with attributes name and grade. Make the grade getter public. Create a better\_grade\_than? method, that you can call like so...**

**puts "Well done!" if joe.better\_grade\_than?(bob)**

**10) Add a class variable to your superclass that can keep track of the number of objects created that inherit from the superclass. Create a method to print out the value of this class variable as well.**

**11) Write a method called age that calls a private method to calculate the age of the vehicle.**

**12) Write a method to write following data into CSV format (person\_list.csv)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SN | First Name | Middle Name | Last Name | DOB |
| 1 | John |  | Doe | 1996-03-29 |
| 2 | Ram | Bd | Magar | 1990-03-29 |
| 3 | Daenerys |  | Targaryen | 1991-03-29 |
| 4 | Tyrion |  | Lannister | 1988-03-29 |

**13) Read the above the CSV and generate a new CSV ( person\_bio.csv ) as below**

|  |  |  |
| --- | --- | --- |
| SN | Full Name | Age |
| 1 | John Doe | 20 |
| 2 | Ram Bd Magar | 26 |
| 3 | Daenerys Targaryen | 25 |
| 4 | Tyrion Lannister | 28 |

**14) Write a class for a Jukebox with attribute songs. Embed methods to play current song, next song, shuffle song, add songs.**

songs = [“song1.mp3”, “song2.mp3”,”song3.mp3”]

jukebox = Jukebox.new(songs)

jukebox.play

#=> “song1.mp3”

jukebox.next

#=> “song2.mp3”

jukebox.previous

#=> “song1.mp3”

jukebox.shuffle

#=> “song2.mp3”

jukebox.add\_song(“songs4.mp3”)

#=> [“song1.mp3”, “song2.mp3”,”song3.mp3”, “song4.mp3”]

**15) Get information through an API call and store it into the CSV.**

Objective:

1. Make an HTTP API call to <http://www.omdbapi.com/> for the following TV Series(type must be tv series)
   1. Fargo
   2. 24
   3. The Knick
   4. Breaking bad
   5. Game of Thrones
   6. Milap
2. Store Title, Year, Genre, Poster, imDB Rating, Type on the csv (i\_love\_series.csv)
3. If the response is false then store “N/A” on each of the column of csv

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Title | Year | Genre | Poster | Imdb Rating | Type |
| Fargo | 2014– | Crime, Drama, Thriller | http://ia.media-imdb.com/images/M/MV5BNDEzOTYzMDkzN15BMl5BanBnXkFtZTgwODkzNTAyNjE@.\_V1\_SX300.jpg | 9.1 | series |
| milap | N/A | N/A | N/A | N/A | N/A |

Hint: Use Gem https://github.com/jnunemaker/httparty