

Ruby - Enumerable - group_by

Another function often used with data collections is one which groups the elements according to some evaluation result.

Consider the following example. Let's say you have a list of 100 integers and you want to group them according to their even and odd value.

In Ruby, you can easily do this by using `group_by` method provide by Enumerable module.

```
> (1..5).group_by {|x| x%2}
{1=>[1,3,5], 0=>[2, 4]}
```

Of course this is a very simple example and it's use can be of much complicated nature.

In this challenge, your task is to group the students into two categories corresponding to their marks obtained in a test. The list of students will be provided in a `marks` hash with student name as key and marks obtained (out of 100) as value pair, along with the `pass_marks` as argument.

The method `group_by_marks` must return a `Hash` containing an array of students who passed as value to `Passed` key, and those who failed as value to `Failed` key. If a particular key is empty, don't return that key.

For example,

```
> marks = {"Ramesh":23, "Vivek":40, "Harsh":88, "Mohammad":60}
> group_by_marks(marks, 30)
=> {"Failed"=>[["Ramesh", 23]], "Passed"=>[["Vivek", 40], ["Harsh", 88], ["Mohammad", 60]]}
```