Ruby Array - Selection

HackerRank

The array class also allows to select and return a subset of an array based on some criteria defined in a block (a block is a group of code within {} that accepts a variable and returns a value).

- Selecting elements that satisfy a given criteria
- Rejecting elements that satisfy a given criteria

```
> arr = [3, 4, 2, 1, 2, 3, 4, 5, 6]
> arr.select {|a| a > 2}
=> [3, 4, 3, 4, 5, 6]
> arr.reject {|a| a > 2}
=> [2, 1, 2]
> arr
=> [3, 4, 2, 1, 2, 3, 4, 5, 6]
> arr.drop_while {|a| a > 1} # removes elements till the block returns false for the first time
=> [1, 2, 3, 4, 5, 6]
```

As you can see, the original array remains unchanged. This is called Non-Destructive Selection.
For destructive behavior (change to the original array), Ruby provides the following methods:

```
> arr = [3, 4, 2, 1, 2, 3, 4, 5, 6]
> arr.delete_if {|a| a < 2}
 => [3, 4, 2, 2, 3, 4, 5, 6]
> arr.keep_if {|a| a < 4}
=> [3, 2, 2, 3]
```

Note

- An element in a block is selected, rejected, deleted, or kept based on the `True` or `False` value generated by that block on that element.
- For a destructive behavior for `select` and `reject` or any method that one wants to enforce a change in the original array, a `!` can be used at the end of the method i.e., `select!` and `reject!`

In this challenge, you have to complete the functions below using syntax as explained above.