Java:

Implement following Menu:

1. Pick array size [1-100]. Arraysize = x create array with all values = 0
2. Fill Array automatically with x random numbers. Number is integer less than 100
3. Find a specific number. user now enters a number. checks if the number is in the array. If so prints "Your number is in the array. Hurray!!!!!". else prints "Your number is not in the array. Boooooooo"
4. Double array values. So, if Arr[i]=3, then after it will be Arr[i]=6. All values of array are doubled by 2
5. Add 1 to array values - So if Arr[i]=3, then after it will be Arr[i]=4. All values of array are added 1
6. Replace specific number. User enters one number to search, and one number for what to replace it with. All instances of the number should be replaced
7. Swap 2 numbers. User enters 2 numbers. If both appear in the array – they will perform a swap
8. Print array.
9. Exit Print "Thanks for using our software. Hate to see you leaving. !@#$%^&*()"

Menu should be printed once, at the beginning, and program should run until user exits.

Every section besides 9 need to have at least 1 function in it.

If an array already exists – create a new one with the same name.

Junit:

Write Junit Test for the following section.

Shahaf to find a solution for everyone regarding how to write to console using Java \ Junit

Make sure you understand what you are testing

Code written by	Tester	Sections to test	
Alex	David	1, 5, 9	
Idan	Shahaf	2,6,7	
Igor	Sergey	7, 3,4	
Sergey	Alex	2,3,5	
Shahaf	Igor	3,4,6	
David	Idan	1,5,9	

Java:

We are going to create a bug tracking system.

WIT = work item type

Bug

Task

User Story

Use case

They all have: unique id – integer. Description – String. Priority – 1-10. Date created – String or date. State = "To Do", "In progress" or "Done"

Bug: Severity – 1 – 10.

Task: Due date – string or date

User story: Sprint name - String

Use case: Array of up to 10 user stories

Implement following menu:

1. Create new item:
    a. Print: "Please choose work item type B for Bug. T for task. US for User story. UC for Use case"
    b. Input from user all data required with proper questions on screen [not including ID]]
2. Set State: This section will end only once state is set

From	To			
To Do	In progress	Valid		
To Do	Done	Not valid	Ask for valid state	
To Do	To Do	Valid		
In progress	To Do	Not valid		
In progress	Done	Valid		
In progress	In progress	Valid		
Done	In progress	Valid		
Done	To Do	Not Valid	Ask for valid state	
Done	Done	Valid		

3. Close all bugs. Set state of all BUGS to Done
4. Close all tasks
5. link user story to Use case by ID's
6. Set new due date for task [print what format is expected]
7. Print use case by id – Prints all linked user stories with all data
8. Set bug severity
9. Change Sprint
10. Exit

every time a not valid input is inserted – print error message and go back to main menu

Selenium:

test following sites using Selenium and Junit

Name	URL	
EVERYONE	Full E2E flow with yourself and everybody else in the course. Send an email to everyone with the top and bottom results https://www.calculator.net/love-calculator.html	
Igor	https://www.calculator.net/age-calculator.html https://www.calculator.net/height-calculator.html	Metric Units
Shahaf	https://www.calculator.net/	
Sergey	https://www.calculator.net/bra-size-calculator.html https://www.calculator.net/time-calculator.html	
David	https://www.calculator.net/fuel-cost-calculator.html	
Idan	https://www.calculator.net/ideal-weight-calculator.html	Metric units
Alex	https://www.calculator.net/date-calculator.html	

Selenium IDE:

1. Record a working flow of you going through 3-5 of your favorite internet sites. Not including porn. Pick things that don't move.
2. Record full tests of http://thedemosite.co.uk/ login and add user