1. Given an age, figure out whether someone is a baby(1 month - 12 months), toddler (1 year - 2 years), child(3 years - 12 years ), teenager(13 years - 17 years), or adult(18 years and more ). Also check that age in months is between 1 and 12.

| Input | Output |
|---|---|
| 45, "years" | "adult" |
| 3, "years" | "child" |
| 8, "months" | "baby" |

2. Percentage marks obtained by a student in three exams are to be entered into a computer. An indication of Pass or Fail is given out after the three marks are entered. The criteria for passing are as follows:
    a. A  student passes if all three examinations are passed.
     b. Additionally a student may pass if only one subject is failed but the overall is average greater than or equal to 50.
The pass mark for an individual subject is 40.

| Input | Output |
|---|---|
| 65, 70, 60 | "Passed" |
| 10, 85, 75 | "Passed" |
| 35, 25, 40 | "Not passed" |
| 20, 40, 40 | "Not passed" |

3. Find the sign of the product of three numbers **without a** multiplication operator. Display the specified sign.

| Input | Output |
|---|---|
| -14, 5, 0 | "unsigned" |
| -8, 9, -6 | "+" |
| 4, 19, -2 | "-" |

4. Input three numbers *a*, *b*, *c* respectively, where a is a non zero number and write a program to solve quadratic equations: $ax^2 + bx + c = 0$. *(Hint: use Math.pow or Math.sqrt)*.

| Input | Output |
|-------|--------|
| 1, 2, 1 | "Solution is -1" |
| 0, 4, -5 | "Enter valid constans" |
| 3, -8, 12 | "Solution does not exists" |
| 5, -13, 6 | "Solutions are 0.6 and 2" |

5. Write a program that reads two strings for playing the game of Rock-Paper-Scissors. If the strings entered by the user are not 'Paper', 'Rock' or 'Scissors', the program keeps on prompting the user to enter new values. If valid strings are inserted, repeat the loop, until one of the sides wins. (*You can use alert instead of console.log*).

| Input | Output |
|-------|--------|
| "Paper", "Pen" | "Invalid input, enter the correct value." |
| "Paper", "Paper" | "Draw! Enter new values." |
| "Paper", "Scissors" | "2nd player wins!" |

6. Write a program, to calculate the value of the following sequence:
   *1 - 1/3 + 1/5 - 1/7 + 1/9 + ….. + $(- 1^k)$* 1/n .

7. Given an array of a size smaller than 100. It consists of numbers from 0 to 99 in any order. Create a new array where each element from that array is placed under the index of its value. Start from the smallest value and end with the biggest one. If there are missing values, put them in their places undefined.

| Input | Output |
|-------|--------|
| [4, 3, 0, 9,] | [0, undefined, undefined, 3, 4, undefined, undefined, undefined, undefined, 9] |
| [26, 30, 19, 5] | [5, *undefined x 13*, 19, *undefined x 4*, 26, *undefined x 3* ] |

8. Given an array consisting of the arrays of numbers (like a two-dimensional array). Find the sum of each row and print them as an array.

| Input | Output |
|---|---|
| [[3, 4, 5], [1, 0, 0], [4, 5, 4], [8, 8, -1]] | [12, 1, 13, 15] |
| [[ 8, 35, 2], [8], [5, 6, -5 , -6], [1, 3, -9, 0, -1]] | [45, 8, 0, -6] |
| [[1], [2], [3], [4]] | [1, 2, 3, 4] |

9. Print the following pattern:

```
*   *   *   *   *   *
*                   *
*                   *
*                   *
*                   *
*                   *
*   *   *   *   *   *
```

10. Write a JavaScript function to get all possible subsets of length 3 of the given array. Assume that all elements in the array are unique.

| Input | Output |
|---|---|
| [4] | [4] |
| [19, 6] | [19, 6] |
| [5, 9, 23, 0, -2, -1] | [[5, 9, 23], [5, 9, 0], [5, 9, -2], [5, 9, -1], [5, 23, 0], [5, 23, -2], [5, 23, -1], [5, 0, -2], [5, 0, -1], [5, -2, -1], [9, 23, 0], [9, 23, -2], [9, 23, -1], [9, 0, -2], [9, 0, -1], [9, -2, -1], [23, 0, -2], [23, 0, -1], [23, -2, -1], [0, -2, -1]] |