

1. **Pagination** nomli class yasang. U o'zining constructoriga array va **options** objectini qabul qiladi. **options** objecti ichiga **limit** parametri bilan array ichidagi elementlar har bir page uchun nechtdan chiqarilishi kiritiladi. Keyinchalik limitni **setLimit** setteri orqali o'zgartirib olish mumkin. **pages** getteri limitga asoslanib arraydagi elementlar jami nechta page bolishini qaytaradi. **getPage()** metodi o'zi biror son qabul qiladi va osha songa mos pageda chiqishi kerak bo'lgan elementlarni arrayda qaytaradi. **filter()** metodi oziga callback funksiya qabul qiladi va callbackning argumentiga keladigan **element**, **index** va **array** ga asoslanib elementlarni filter qiluvchi biror condition qaytaradi. Filter qilingandan so'ng, pagination filter qilingan elementlar ustidagina ishlaydi.

Diqqat!: pagination ichiga kiritiladigan array faqat stringdan yoki objectdan yoki arraydan yoki numberdan tashkil topadi. Barcha class behaviorlari ushbu data tiplar bilan ishlay olsin.

```
let array = ['A', 'B', 'C', 'D', 'E', 'C', 'B', 'A', 'B', 'E']

let pagination = new Pagination(array, { limit: 3 })

pagination.pages           // 4
pagination.getPage(2)      // ['D', 'E', 'C']
pagination.getPage(4)      // ['E']

pagination.setLimit = 2
pagination.pages           // 5
pagination.getPage(1)      // ['A', 'B']

pagination.filter( (el, i, arr) => el !== 'B' )
pagination.pages           // 4
pagination.getPage(1)      // ['A', 'C']
pagination.getPage(2)      // ['D', 'E']
```

2. **divide** nomli funksiya yasang. Funksiya birinchi argumentiga **manfiy** va **musbat** raqamlardan tashkil topgan **array** kiritiladi. Ikkinchi argumentiga biror **raqam** kiritiladi. Funksiya array ichidagi raqamlarni yig'indisi ikkinchi argumentda kiritilgan **raqam**ga teng yoki kichik bo'lmaguncha chapdan o'ngga qarab qo'shib boradi. Funksiya bitta yoki bir nechta raqamlar yig'indisi kiritilgan raqamga teng yoki undan kichik bo'lganda, ularni bitta to'plam qilib oladi va kelgan joyidan yana raqamlarni qo'shib ketaveradi. Funksiya mana shunday to'plamlardan iborat **array** qaytarishi keak.

Eslatma: Funksiyaga kiritilgan array ichidagi raqamlarning hech qaysisi ikkinchi argumentdagi raqamdan katta bo'lmaydi.

```

divide([1, 2, 3, 4, 1, 0, 2, 2], 5)    // [[1, 2], [3], [4, 1, 0], [2, 2]]
divide([1, 0, 1, 1, -1, 0, 0], 1)     // [[1, 0], [1], [1, -1, 0, 0]]
divide([2, 1, 0, -1, 0, 0, 2, 1, 3], 3) // [[2, 1, 0, -1, 0, 0], [2, 1], [3]]

```

3. **join** nomli funksiya yasang. Funksiya argumentiga **string**lardan tashkil topgan **array** kiritiladi. Funksiya array ichidagi har bir stringning **oxirgi** belgilarini, o'zidan keyingi stringning **boshidagi** belgilari bilan taqqoslaydi. Ikkalasida ham umumiy bir xil belgilar bor bolsa, bir xil joyini olib tashlab stringlarni bir biriga qoshadi va o'zidan keyingi stringlar bilan ham xuddi shu ishni takrorlaydi. Funksiya array va uning ichida oxirida hosil bo'lgan stringni va stringlar orasida bo'lgan eng kam umumiylikni raqam qilib qaytaradi.

```

join(["oven", "envier", "erase", "serious"]) // ["ovenvieraserious", 2]
join(["move", "over", "very"])              // ["moververy", 3]
join(["to", "ops", "psy", "syllable"])       // ["topsyllable", 1]
/*
    "to" va "ops" da umumiy belgilar faqat "o" (1)
    "ops" va "psy" da umumiy belgilar faqat "ps" (2)
    "psy" va "syllable" da umumiy belgilar faqat "sy" (2)
    uchala umumiylik ichida eng kichigi 1
*/
join(["aaa", "bbb", "ccc", "ddd"])           // ["aaabbbcccddd", 0]

```

4. **describe** nomli funksiya va **Person** nomli constructor funksiya yasang. Person funksiyasidan instance olinib uning constructori birinchi argumentiga stringda ism, ikkinchiga raqamda yosh kiritiladi. describe funksiyasini olingan instance scopida ishga tushirilganda quyidagicha natija chiqishi kerak.

```
function describe () {}
function Person (name, age) {}

let patrick = new Person('Patrick', 20)
let john = new Person('John', 15)

describe.call(patrick)    // 'Patrick, 20 years old.'
describe.call(john)       // 'John, 15 years old.'
```

5. **group** nomli funksiya yasang. Funksiya birinchi argumentiga raqamlardan tashkil topgan array kiritiladi. Ikkinchi argumentiga biror raqam kiritiladi. Funksiya array ichidagi toq raqamlardan juftlariga qarab uzunligi ikkinchi argumentdagi raqamdan oshmaydigan to'plam arraylarga joylab chiqishi kerak. Funksiya oxirida to'plam arraylardan iborat array qaytaradi.

Eslatma: Har bir to'plamning uzunligi, kiritilgan raqamdan oshib ketmasin.

```
group([1, 2, 3, 4], 2)    // [[1, 3], [2, 4]]
group([1, 2, 3, 4, 5, 6, 7], 4) // [[1, 3, 5, 7], [2, 4, 6]]
group([1, 2, 3, 4, 5], 1)  // [[1], [2], [3], [4], [5]]
group([1, 2, 3, 4, 5, 6], 4) // [[1, 3, 5], [2, 4, 6]]
```

