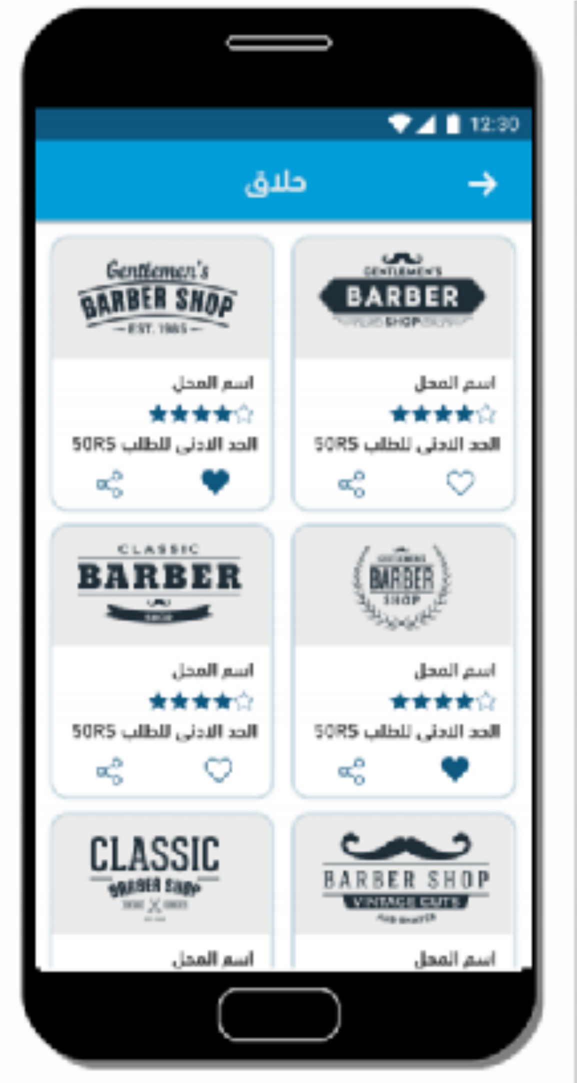
Grand Software Engineers Task

A- PHP task



اسم المحل
★★★★☆
الحد الادنى للطلب 50RS

اسم المحل
★★★★☆
الحد الدنى للطلب 50RS

اسم المحل
★★★★☆
الحد الادنى للطلب 50RS

اسم المحل
★★★★☆
الحد الدنى للطلب 50RS

اسم المحل

اسم المحل

We have a mobile application which list barbers shops for customers.

Your task :

Create the suitable APIS and control panel for this page.

1- Please Just create the suitable APIS for this page like (Get barbers shops list , add barbers shops to favorites also on).

response should be like that

```
{
      "status": "success",
      "msg": “any message if needed",
      "data": [
          {"id": 1 , “shop_name”: “shop1" , "price": 50.5 ,
“image”: “image1.jpg” , “rate”: 4.3 , “is_ favorite”: false},
          {"id": 1 , “shop_name”: “shop1" , "price": 77 ,
“image”: “image1.jpg” , “rate”: 4.3 , “is_ favorite”: false}
]
}
```

2- Also create a control panel that helps us to add ,delete, update ,and active barbers shops.

B- Problem solving task

In our application when two customers need to book at the same time , we create a nice game if customer won he will book first,

The player has a positive integer n and the player is able to do the following operation as many times as the player want (possibly zero): choose a positive integer x that $x<n$ and x is not a divisor of n then subtract x from n
The goal of the player is to minimize the value of n in the end. one of our Customers found the game trivial. Can you also beat the game?

Just create a simple api that take a single integer and return the answer of the game

Input

Single integer n , 1<=n<=1000000000

Output

Output a single integer, the minimum value of n the player can get.

Examples :

Input
8
Output
1

Input
1
Output
1

Note

In the first example, the player can choose
$x=3$ *in the first turn, then n becomes 5.*
He can then choose $x=4$ *in the second turn to get* $n=1$
as the result. There are other ways to get this minimum. However, for example, he cannot choose $x=2$ in the first turn because 2 is a divisor of 8.
In the second example, since $n=1$
initially, the player can do nothing.

Evaluation criteria:

- Running Code.
- Handle all error scenarios like no internet connection…..etc.
- Define and run the test case after you finish each component(write all available test case , each test case will be defined in one sentence).
- Structured code.
- Your code must be documented.
- Your code must not have any duplications codes; all your components must be reusable.
- Please push your code to your git .