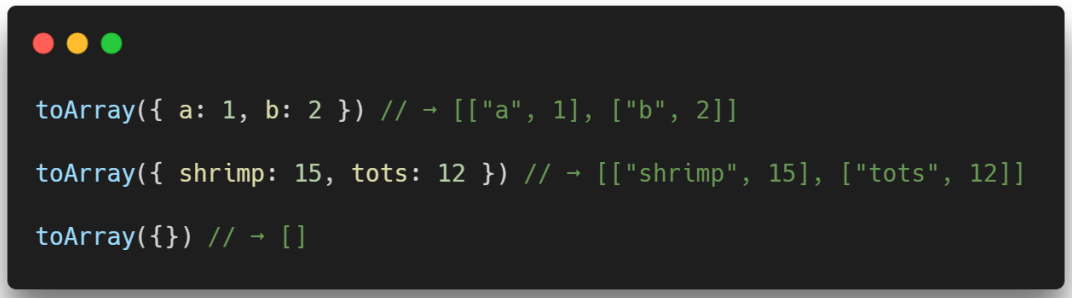
Question-1:

Write a function that converts an object into an array, where each element represents a key-value pair in the form of an array.

Examples



```
toArray({ a: 1, b: 2 }) // → [["a", 1], ["b", 2]]

toArray({ shrimp: 15, tots: 12 }) // → [["shrimp", 15], ["tots", 12]]

toArray({}) // → []
```

Notes

- Return an empty array if the object is empty.
- Don’t use Object.entries().

Question-2:

Given any number, we can create a new number by adding the sums of squares of digits of that number. For example, given 203, our new number is 4 + 0 + 9 = 13. If we repeat this process, we get a sequence of numbers:

```
203 -> 13 -> 10 -> 1 -> 1
```

Sometimes, like with 203, the sequence reaches (and stays at) 1. Numbers like this are called happy.

Not all numbers are happy. If we started with 11, the sequence would be:
`11 -> 2 -> 4 -> 16 -> ...`

This sequence will never reach 1, and so the number 11 is called unhappy.

Given a positive whole number, you have to determine whether that number is happy or unhappy.

Examples

Notes

- You can assume (and it is actually true!) that all positive whole numbers are either happy or unhappy. Any happy number will have a 1 in its sequence, and every unhappy number will have a 4 in its sequence.
- The only numbers passed to your function will be positive whole numbers.