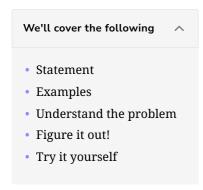
## Time-Based Key-Value Store

Try to solve the Time-Based Key-Value Store problem.



#### **Statement**

Implement a data structure that can store multiple values of the same key at different timestamps and retrieve the key's value at a certain timestamp.

You'll need to implement the  ${\bf TimeStamp}$  class. This class has the following functions:

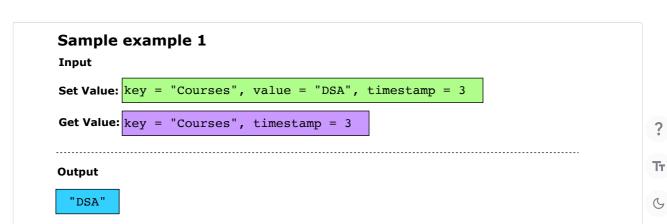
- Init(): This function initializes the values dictionary and timestamp dictionary.
- Set Value(key, value, timestamp): This function stores the key and value at any given timestamp.
- **Get Value(key, timestamp):** This function returns the value set for this key at the specified timestamp.

**Note**: When a query requests the value of a key at a timestamp that is more recent than the most recent entry for that key, our data structure should return the value corresponding to the most recent timestamp.

#### **Constraints:**

- $1 \leq \text{key.length}$ , value.length  $\leq 100$
- key and value consist of lowercase English letters and digits.
- $1 < \texttt{timestamp} < 10^3$
- At most  $2 \times 10^3$  calls will be made to **Set Value** and **Get Value**.
- All the timestamps, timestamp, of **Set Value** are strictly increasing.

#### **Examples**



```
Sample example 2
Input

Set Value: key = "Courses", value = "OOP", timestamp = 5

Get Value: key = "Courses", timestamp = 7

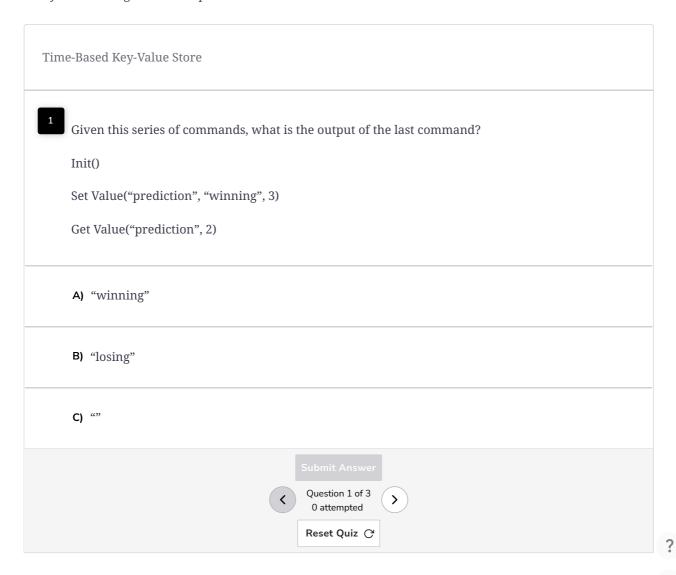
Output

"OOP"
```

# Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps us to check that you're solving the correct problem:

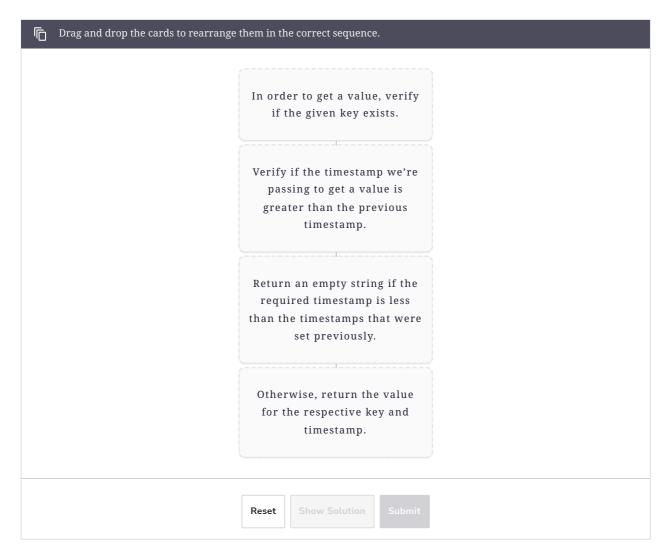
- []



# Figure it out!

TT

We have a game for you to play. Rearrange the logical building blocks required to implement Get Value().



### Try it yourself

Implement your solution in TimeStamp.java in the following coding playground. We have provided a useful code template in the other file that you may build on to solve this problem.

