

Java

```
class Logger {  
  
    public Logger() {  
  
    }  
  
    public boolean shouldPrintMessage(int timestamp, String message) {  
  
    }  
}  
  
/**  
 * Your Logger object will be instantiated and called as such:  
 * Logger obj = new Logger();  
 * boolean param_1 = obj.shouldPrintMessage(timestamp,message);  
 */
```

JavaScript

```
var Logger = function() {  
  
};  
  
/**  
 * @param {number} timestamp  
 * @param {string} message  
 * @return {boolean}  
 */  
Logger.prototype.shouldPrintMessage = function(timestamp, message) {
```

```
};

/**
 * Your Logger object will be instantiated and called as such:
 * var obj = new Logger()
 * var param_1 = obj.shouldPrintMessage(timestamp,message)
 */
```

TypeScript

```
class Logger {
    constructor() {

    }

    shouldPrintMessage(timestamp: number, message: string): boolean {

    }
}

/**
 * Your Logger object will be instantiated and called as such:
 * var obj = new Logger()
 * var param_1 = obj.shouldPrintMessage(timestamp,message)
 */
```

C++

```
class Logger {
public:
    Logger() {

    }

    bool shouldPrintMessage(int timestamp, string message) {

    }
};

/**
 * Your Logger object will be instantiated and called as such:
 * Logger* obj = new Logger();
 * bool param_1 = obj->shouldPrintMessage(timestamp,message);
 */
```

C#

```
public class Logger {

    public Logger() {

    }

    public bool ShouldPrintMessage(int timestamp, string message) {

    }

}

/**
```

```
* Your Logger object will be instantiated and called as such:  
* Logger obj = new Logger();  
* bool param_1 = obj.ShouldPrintMessage(timestamp,message);  
*/
```

Kotlin

```
class Logger() {  
  
    fun shouldPrintMessage(timestamp: Int, message: String): Boolean {  
  
    }  
  
}  
  
/**  
 * Your Logger object will be instantiated and called as such:  
 * var obj = Logger()  
 * var param_1 = obj.shouldPrintMessage(timestamp,message)  
 */
```

Go

```
type Logger struct {  
  
}  
  
func Constructor() Logger {
```

```
}
```

```
func (this *Logger) ShouldPrintMessage(timestamp int, message string) bool {
```

```
}
```

```
/**
```

```
 * Your Logger object will be instantiated and called as such:
```

```
 * obj := Constructor();
```

```
 * param_1 := obj.ShouldPrintMessage(timestamp,message);
```

```
 */
```

```
-----
```