# JS Advanced: Exam 13 November 2016

Problems for exam preparation for the [“JavaScript Advanced” course @ SoftUni](https://softuni.bg/courses/javascript-advanced). Submit your solutions in the SoftUni judge system at <https://judge.softuni.bg/Contests/355/>.

# Problem 3. Mail Box (Simple Class)

Write a **JavaScript class** MailBox to hold a list of mail messages (subject + text).

|  |
| --- |
| **class** MailBox {  *//* ***TODO: implement this class*** } |

Each **mail message** holds **subject** and **text**. Implement the following features:

* **Constructor** – creates an empty mail box.
* Method addMessage(subject, text) – adds a mail message to the mail box. Both **subject** and **text** are strings. Returns the mailbox itself to allow chaining.
* Accessor property messageCount – returns thetotal number of messages in the mail box.
* Method deleteAllMessages() – clears the mail box (deletes all messages).
* Method findBySubject(substr) – returns all mail messages from the mail box that hold the specified substr string in their subject. Order the results by their order of adding to the mail box. Return the results as array of objects {subject, text}. If no messages are matched, return an empty array.
* Method toString() – returns the text representation of the mail box in the following format:
  + Empty mail box:

|  |
| --- |
| \* (empty mailbox) |

* + Non-empty mail box:

|  |
| --- |
| \* [subject1] Text1  \* [subject2] Text2  \* [subject3] Text3  … |

### Examples

This is an example how the MailBox class is **intended to be used**:

|  |
| --- |
| Sample code usage |
| **let *mb*** = **new *MailBox***(); **console**.log(**"Msg count: "** + ***mb***.messageCount); **console**.log(**'Messages:\n'** + ***mb***); ***mb***.addMessage(**"meeting"**, **"Let's meet at 17/11"**); ***mb***.addMessage(**"beer"**, **"Wanna drink beer tomorrow?"**); ***mb***.addMessage(**"question"**, **"How to solve this problem?"**); ***mb***.addMessage(**"Sofia next week"**, **"I am in Sofia next week."**); **console**.log(**"Msg count: "** + ***mb***.messageCount); **console**.log(**'Messages:\n'** + ***mb***); **console**.log(**"Messages holding 'rakiya': "** +  **JSON**.stringify(***mb***.findBySubject(**'rakiya'**))); **console**.log(**"Messages holding 'ee': "** +  **JSON**.stringify(***mb***.findBySubject(**'ee'**)));  ***mb***.deleteAllMessages(); **console**.log(**"Msg count: "** + ***mb***.messageCount); **console**.log(**'Messages:\n'** + ***mb***);  **console**.log(**"New mailbox:\n"** +  **new *MailBox***()  .addMessage(**"Subj 1"**, **"Msg 1"**)  .addMessage(**"Subj 2"**, **"Msg 2"**)  .addMessage(**"Subj 3"**, **"Msg 3"**)  .toString()); |

|  |
| --- |
| Corresponding output |
| Msg count: 0  Messages:  \* (empty mailbox)  Msg count: 4  Messages:  \* [meeting] Let's meet at 17/11  \* [beer] Wanna drink beer tomorrow?  \* [question] How to solve this problem?  \* [Sofia next week] I am in Sofia next week.  Messages holding 'rakiya': []  Messages holding 'ee': [{"subject":"meeting","text":"Let's meet at 17/11"},{"subject":"beer","text":"Wanna drink beer tomorrow?"},{"subject":"Sofia next week","text":"I am in Sofia next week."}]  Msg count: 0  Messages:  \* (empty mailbox)  New mailbox:  \* [Subj 1] Msg 1  \* [Subj 2] Msg 2  \* [Subj 3] Msg 3 |

### Submission

Submit your class MailBox as “**JavaScript code (Mocha unit tests)** ”.