**Forum - Requirement 1**

Jack was fond of Q&A forums which are available. He always wanted to create his own Q&A forum application. You being an aspiring programmer, Jack approaches you for help. Help him to build the application.  
  
**Requirement 1:**

The first and foremost requirement of any application is asking the user to register. And there should not be two User accounts with same E-mail ID.  
  
Let’s start by creating two **User** objects and check whether they are equal.

1. Create a **User** Class with the following attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_name | string |
| \_emailId | string |
| \_dateOfBirth | DateTime |

1. Mark all the attributes as private
2. Include appropriate properties.
3. Add a default constructor and a parameterized constructor to take in all attributes in the given order:   
   **User(long \_id, string \_name, string \_emailId, DateTime \_dateOfBirth)**
4. When the “**User**” object is printed, it should display the following details: **[Override the ToString method]**  
   Print format:  
   Id:"\_id"  
   Name:"\_name"  
   Email Id:"\_emailId"  
   Date of birth:"\_dateOfBirth"
5. Two users are considered same if they have the same **\_emailId** and **\_dateOfBirth**. Implement the logic in the appropriate function.  
   **[Override the Equals method]**  
     
   The input format consists of user details separated by comma in the below order,  
   **(\_id,\_name,\_emailId,\_dateOfBirth)**

The Input to your program would be details of two users, you need to display their details as given in "5th point(refer above)" and compare the two users and display whether they are same or different.  
  
Create a class named as **Program**, which contains Main method. All the operations are done in this method.  
It is used to access the above classes and its method to do this requirement.  
  
**Note:** The user details should be given as a comma-separated value in the below order,  
              \_**id, \_name,\_emailId,\_dateOfBirth**  
There is an empty line between display statements. Print the empty lines in main function.  
  
**Sample INPUT & OUTPUT 1:**

Enter user 1 detail:  
**1004,Morsh,morsh@yahoo.com,26-01-1997**  
Enter user 2 detail:  
**1004,Morsh,morsh@yahoo.com,26-01-1997**  
  
User 1:  
Id:1004  
Name:Morsh  
Email Id:morsh@yahoo.com  
Date of birth:26-01-1997  
  
User 2:  
Id:1004  
Name:Morsh  
Email Id:morsh@yahoo.com  
Date of birth:26-01-1997  
  
User 1 is same as User 2  
  
**Sample INPUT & OUTPUT 2:**  
Enter user 1 detail:  
**1010,Saminathan,saminathan@rediff.com,12-05-1997**  
Enter user 2 detail:  
**1007,Meyyappan,meyyappan@gmail.com,30-03-1997**  
  
User 1:  
Id:1010  
Name:Saminathan  
Email Id:saminathan@rediff.com  
Date of birth:12-05-1997  
  
User 2:  
Id:1007  
Name:Meyyappan  
Email Id:meyyappan@gmail.com  
Date of birth:30-03-1997  
  
User 1 and User 2 are different

**Forum - Requirement 2**

**Requirement 2:**  
Now we are going to start creating a forum and add posts to it. Start with creating a forum and use menu-driven approach to add, remove, display details of the posts in the forum.  
  
a)Create a Class **User** with the following attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_name | string |
| \_emailId | string |
| \_dateOfBirth | DateTime |

b) Create a Class **Post** with the following attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_content | string |
| \_type | string |
| \_postedTime | DateTime |
| \_postedBy | User |
| \_numberOfUpVotes | int |
| \_numberOfDownVotes | int |

Mark all the attributes as private.  
Include appropriate properties.  
Override **ToString()** method to display details in tabular form.  
Add a default constructor and a parameterized constructor to take in all attributes in the given order:  
**User(long \_id, string \_name, string \_emailId, DateTime \_dateOfBirth)** and  
**Post(long \_id, string \_content, string \_type, DateTime \_postedTime,User \_postedBy, int \_numberOfUpVotes, int \_numberOfDownVotes)**  
  
c)Create a Class **Forum** with the following attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_name | string |
| \_postList | List<Post> |

Mark all the attributes as private.  
Include appropriate properties.  
Add a default constructor and a parameterized constructor to take in all attributes in the given order:  
**Forum(Long id,String name, List<Post> postList).**  
In constructor pass the postList value as an empty list. Only one forum will be present at a time.  
  
d) Create the following static method in **User**class,

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public static User CreateUser(string detail) | This method accepts a string which contains user details separated by commas. Split the details and create a user object from the details and return it. |

The user details are given as a comma-separated value in the below order,  
**id,name,emailId,dateofBirth**  
  
e) Create the following static method in **Post**class,

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public static Post CreatePost(string detail,User user) | This method accepts a string which contains post details separated by commas and the User object. Split the details and create a post object from the details and return it. |

The post details are given as a comma-separated value in the below order,  
**id,content,type,postedTime,postedBy,numberOfUpVotes,numberOfDownVotes.**  
  
f) Create the following methods in **Forum**class,

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public void AddPostToForum(Post post) | This method accepts a post object and adds the post to the post list of the current forum. |
| public bool RemovePostFromForum(long id) | This method will get the name of the post and delete the post with the specified name from the current forum.  If a post with the given name found, delete the post and return **true**, then print "**Post successfully deleted**".  If a post with the name is not found return **false**, then print "**Post not found in the Forum**". |
| public void DisplayPosts() | This method will display the post list in the current forum. If the post list is empty display "**No posts to show"**, else display "**Posts in: [Forum name]**" and display all the post details in the specified format. Where [Forum name] specifies the name of the forum. |

The forum details are given as a comma-separated value in the below order,  
**\_id,\_name**  
  
After deletion, if true is returned print "**Post successfully deleted**", else print "**Post not found in the Forum**". After adding a post to the forum, print "**Post successfully added**".  
  
Create a class named as **Program**, which contains **Main** method. All the Input operations are done in this method.  
It is used to access above class and its method to done this requirement.  
  
**Note:** The above print statements should be present in the Main method.  
             While getting the input for a Post, the first line of input corresponds to User details and the next line corresponds to Post details.  
  
Print format:  
**Console.WriteLine("{0,-5} {1,-46} {2,-10} {3,-20} {4,-10} {5,-10} {6}","Id","Content","Type","Posted Time","Posted By","Up Votes","Down Votes");**  
  
**[All text in bold corresponds to the input and rest corresponds to the output]**  
**Sample Input and Output:**

Enter the details of the forum:

**2001,E-Box Forum**

1.Add Post

2.Delete Post

3.Display Posts

4.Exit

Enter your choice:

**1**

Enter the detail

**1004,morsh,morsh@yahoo.com,26-01-1997**

**5126,Protecting database data,MySql,13-02-2015 01:21:30,2015,501**

Post successfully added

1.Add Post

2.Delete Post

3.Display Posts

4.Exit

Enter your choice:

**1**

Enter the detail

**1001,sathish,sathish@hotmail.com,13-03-1996**

**6012,How to search images directly from the folder,MySql,30-01-2016 11:12:45,3012,412**

Post successfully added

1.Add Post

2.Delete Post

3.Display Posts

4.Exit

Enter your choice:

**3**

Posts in: E-Box Forum

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id | Content | Type | Posted Time | Posted By | Up Votes | Down Votes |
| 5126 | Protecting database data | MySql | 13-02-2015 01:21:30 | morsh | 2015 | 501 |
| 6012 | How to search images directly from the folder | MySql | 30-01-2016 11:12:45 | sathish | 3012 | 412 |

1.Add Post

2.Delete Post

3.Display Posts

4.Exit

Enter your choice:

**2**

Enter the id of the post to be deleted:

**5126**

Post successfully deleted

1.Add Post

2.Delete Post

3.Display Posts

4.Exit

Enter your choice:

**3**

Posts in: E-Box Forum

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id | Content | Type | Posted Time | Posted By | Up Votes | Down Votes |
| 6012 | How to search images directly from the folder | MySql | 30-01-2016 11:12:45 | sathish | 3012 | 412 |

1.Add Post

2.Delete Post

3.Display Posts

4.Exit

Enter your choice:

**4**

**Forum - Requirement 3**

**Requirement 3:**  
   In this requirement, you need to validate the emailid of the user and review whether the content of the post is spam or not.  
  
a)Create a Class **Program**with the following static methods:

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| static bool ValidateEmailId(string email) | Validate the emailId based on the rules given below. Returns **true**ifemailId is valid else return **false**. |
| static bool ValidateContent(string content) | Review the content based on the rules given below. Returns **true**ifcontent is valid else return **false**. |

b) While validating email id follow the below rules. The format of the email id is given below  
   
            username@domain.TLD  
  
where, TLD - Top Level Domain  
  
1. The email id should start only with alphabets(either uppercase or lowercase).  
2. The email username can contain alphabets(either uppercase or lowercase), numbers and the special characters (  .   and   \_    ).  
3. The email username should not contain any special characters other than**" . "**and**" \_ "**.  
4. After the username special character @ should present.  
5. The email domain should contain only alphabets(either uppercase or lowercase).  
6. After email domain, a value dot ( . ) should present.  
7. The email Top Level Domain should contain only alphabets(both uppercase and lowercase) and it should have only 2 to 6 characters.  
  
Example: **alpha\_Beta.01@google.com** is a valid email id.  
Since the username contain only alphabets, numbers and a special character ( \_ and . ), then the @ symbol is present. The domain name should contain only alphabets and the symbol dot( . ) and the Top Level Domain have only 3 characters.  
  
c) While reviewing content follow the below rules,  
  
1.Same character should not appear more than twice consecutively in the content.  Eg (aaa is spam)

2.The content should contains more than 15 characters excluding spaces.  
  
**Note:** Print "**Email is valid**" if email is valid else print "**Email is invalid**".  
            Print "**Content is a spam**" if content is spam else print "**Content is not a spam**".  
            All the above print statements are present in the **Main** method.  
  
**Problem Overview:**  
This requirement contains menu driven,  
Menu:  
1.Validate Email  
2.Review for Spam  
  
If the user select option-1, then get the email from the user, and validate the user entered email(Use ValidateEmailId method to validate email), If that method returns true, then print "**Email is valid**", else print "**Email is invalid**".  
  
If the user select option-2, then get the content from the user, and validate the user entered content(Use ValidateContent method to validate content), If that method returns true, then print "**Content is not a spam**", else print "**Content is a spam**".

**Sample Input and Output 1:**  
  
1.Validate Email  
2.Review for Spam  
Enter your choice:  
**1**  
Enter the email to be validated:  
**sathish@hotmail.com**  
Email is valid  
  
**Sample Input and Output 2:**  
  
1.Validate Email  
2.Review for Spam  
Enter your choice:  
**1**  
Enter the email to be validated:  
**meyyappan@gmail.c**  
Email is invalid  
  
**Sample Input and Output 3:**  
  
1.Validate Email  
2.Review for Spam  
Enter your choice:  
**2**  
Enter the content to be reviewed:  
**Using camera calibration to segment the region of interest**  
Content is not a spam  
  
**Sample Input and Output 4:**  
1.Validate Email  
2.Review for Spam  
Enter your choice:  
**2**  
Enter the content to be reviewed:  
**SQL complex queries aaabbbaaa**  
Content is a spam

**Forum - Requirement 4**

**Requirement 4:**  
In this requirement develop a feature in which you can search a List of Posts by user's name, numberOfUpVotes.  
  
a) Create a Class **User** with the following attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_name | string |
| \_emailId | string |
| \_dateOfBirth | DateTime |

b)Create a class **Post** with the follwing attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_content | string |
| \_type | string |
| \_postedTime | DateTime |
| \_postedBy | User |
| \_numberOfUpVotes | int |
| \_numberOfDownVotes | int |

c) Create a class **PostBO**with the following methods,

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public List<Post> FindPost(List<Post> postList,string name) | This method accepts a list of posts and user's name as arguments and returns a list of posts that match with given user's name. |
| public List<Post> FindPost(List<Post> postList,int numberOfUpVotes) | This method accepts a list of posts and numberOfUpVotes as arguments, then returns the list of posts having numberOfUpVotes greater than or equal to specified numberOfUpVotes. |

Mark all the attributes as private.  
Include appropriate properties.  
Add a default constructor and a parameterized constructor to take in all attributes in the given order:  
**Post(long \_id, string \_content, string \_type, DateTime \_postedTime,User \_postedBy, int \_numberOfUpVotes, int \_numberOfDownVotes)** and   
**User(long \_id, string \_name, string \_emailId, DateTime \_dateOfBirth)**  
  
d) Create the following method in the User Class

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public static User CreateUser(string detail) | This method accepts a string which contains user details separated by commas. Split the details and create a user object from the details and return it. |

e) Create the following method in the Post Class

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public static Post CreatePost(string detail,User user) | This method accepts a string which contains post details separated by commas and the User object. Split the details and create a post object from the details and return it. |

The user details should be given as a comma-separated value in the below order,  
**\_id,\_name,\_emailId,\_dateOfBirth**  
  
The post details should be given as a comma-separated value in the below order,  
**\_id,\_content, \_type,\_postedTime,\_postedBy,\_numberOfUpVotes,\_numberOfDownVotes.**

Create a class named as **Program**, which contains **Main**method, All the input operations are done in this method.  
It is used to access the above class and its method to done this requirement.  
  
Print format:  
**Console.WriteLine("{0,-5} {1,-46} {2,-10} {3,-20} {4,-10} {5,-10} {6}","Id","Content","Type","Posted Time","Posted By","Up Votes","Down Votes");**  
  
**Note:**The post lists are displayed in the main method.  
              If any other choice is selected, display "**Invalid choice**"  
              If search detail is not found, display "**No such post is present**"  
             While giving inputs, the first line corresponds to the User details and next line corresponds to the Post details.  
  
Example:  
i)Below is a user detail (\_id,\_name,\_emailId,\_dateOfBirth)  
**1004,morsh,morsh@yahoo.com,26-01-1997**  
  
ii)Below is a post detail (\_id,\_content, \_type,\_postedTime,\_numberOfUpVotes,\_numberOfDownVotes)  
**8022,Java stuck in a simple for loop,Java,13-01-2018 12:24:02,100,10**  
  
**[All text in bold corresponds to the input and rest corresponds to the output]**  
**Sample Input and Output 1:**

Enter the number of Posts:

**3**

Enter detail 1:

**1004,morsh,morsh@yahoo.com,26-01-1997**

**8022,Java stuck in a simple for loop,Java,13-01-2018 12:24:02,100,10**

Enter detail 2:

**1010,saminathan,saminathan@rediff.com,12-05-1997**

**6099,Code for limiting password characters,Java,09-06-2016 08:45:22,2000,50**

Enter detail 3:

**1007,meyyappan,meyyappan@gmail.com,30-03-1997**

**7015,Java 8 Streams IO,Java,22-12-2017 23:06:55,200,60**

Enter a search type:

1.By User Name

2.By Number of UpVotes

**1**

Enter the User Name:

**saminathan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id | Content | Type | Posted Time | Posted By | Up Votes | Down Votes |
| 6099 | Code for limiting password characters | Java | 09-06-2016 08:45:22 | saminathan | 2000 | 50 |

**Sample Input and Output 2:**

Enter the number of Posts:

**3**

Enter detail 1:

**1004,morsh,morsh@yahoo.com,26-01-1997**

**8022,Java stuck in a simple for loop,Java,13-01-2018 12:24:02,100,10**

Enter detail 2:

**1010,saminathan,saminathan@rediff.com,12-05-1997**

**6099,Code for limiting password characters,Java,09-06-2016 08:45:22,2000,50**

Enter detail 3:

**1007,meyyappan,meyyappan@gmail.com,30-03-1997**

**7015,Java 8 Streams IO,Java,22-12-2017 23:06:55,200,60**

Enter a search type:

1.By User Name

2.By Number of UpVotes

**2**

Enter the Number of UpVotes:

**200**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id | Content | Type | Posted Time | Posted By | Up Votes | Down Votes |
| 6099 | Code for limiting password characters | Java | 09-06-2016 08:45:22 | saminathan | 2000 | 50 |
| 7015 | Java 8 Streams IO | Java | 22-12-2017 23:06:55 | meyyappan | 200 | 60 |

**Sample Input and Output 3:**

Enter the number of Posts:

**3**

Enter detail 1:

**1004,morsh,morsh@yahoo.com,26-01-1997**

**8022,Java stuck in a simple for loop,Java,13-01-2018 12:24:02,100,10**

Enter detail 2:

**1010,saminathan,saminathan@rediff.com,12-05-1997**

**6099,Code for limiting password characters,Java,09-06-2016 08:45:22,2000,50**

Enter detail 3:

**1007,meyyappan,meyyappan@gmail.com,30-03-1997**

**7015,Java 8 Streams IO,Java,22-12-2017 23:06:55,200,60**

Enter a search type:

1.By User Name

2.By Number of UpVotes

**4**

Invalid choice

**Forum - Requirement 5**

**Requirement 5:**  
  
In this requirement, given a list of post you need to find the number of posts created each day using SortedDictionary.  
  
a) Create a Class **User** with the following attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_name | string |
| \_emailId | string |
| \_dateOfBirth | DateTime |

Mark all the attributes as private,  
Include appripriate properties.  
 Add a default constructor and a parameterized constructor to take in all attributes in the given order:   
**User(long \_id, string \_name, string \_emailId, DateTime \_dateOfBirth)**  
  
b) Create a Class **Post** with the following attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_content | string |
| \_type | string |
| \_postedTime | DateTime |
| \_postedBy | User |
| \_numberOfUpVotes | int |
| \_numberOfDownVotes | int |

Mark all the attributes as private,  
Include appropriate properties.  
 Add a default constructor and a parameterized constructor to take in all attributes in the given order:   
**Post(long \_id, string \_content, string \_type, DateTime \_postedTime,User \_postedBy, int \_numberOfUpVotes, int \_numberOfDownVotes)**  
  
c) Create the following static methods in the Post class,

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| static SortedDictionary<Date,Integer> CalculateDateCount(List<Post> list) | This method accepts a list of Post as arguments and returns a**SortedDictionary** with the **postedDate** as key and number of postscreated on that date as value and returns the SortedDictionary. |

In the SortedDictionary have the postedDate as key and Count the number of posts created on the date and keep the number of posts as value. Print the value sorted by Date.  
  
**Note**:The user details should be given as a comma separated value in the below order,  
**\_id, \_name,\_emailId,\_dateOfBirth**  
           The Post details should be given as a comma separated value in the below order,  
            **\_id, \_content,\_type,\_postedTime,\_numberOfUpVotes,\_numberOfDownVotes**  
  
Print format:  
**Console.WriteLine("{0,-15} {1}","Date","Count");**  
  
**Sample Input and Output 1:**  
Enter the number of posts:  
**3**  
Enter the detail 1  
**1001,sathish,sathish@hotmail.com,13-03-1996  
7015,Java 8 Streams IO,Java,22-12-2017 23:06:55,200,60**  
Enter the detail 2  
**1004,morsh,morsh@yahoo.com,26-01-1997  
6562,Thresholding a signal using hysteresis,Matlab,03-03-2016 11:12:45,3506,200**  
Enter the detail 3  
**1007,meyyappan,meyyappan@gmail.com,30-03-1997  
9865,How change values from Server System Variables,MySql,31-12-2017 02:12:13,568,45**  
Date            Count  
03-03-2016      1  
22-12-2017      1  
31-12-2017      1

**Forum -Requirement 6**

**Requirement 6:**  
  
In this requirement, you need to find the current trending topic based on the type of posts.  
A trending topic is the one which has maximum number of posts.  
  
a) Create a Class **User** with the following attributes:

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_name | string |
| \_emailId | string |
| \_dateOfBirth | DateTime |

b) Create the following static methods in the **User** class,

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| static User CreateUser(string detail) | This method accepts a String. The user detail separated by commas is passed as the argument. Split the details and create a user object and returns it. |

The note details should be given as a comma-separated value in the below order,  
**\_id,\_name,\_emailId,\_dateOfBirth**  
  
c)Create a class **Post** with the following attributes

|  |  |
| --- | --- |
| **Member Field Name** | **Type** |
| \_id | long |
| \_content | string |
| \_type | string |
| \_postedTime | DateTime |
| \_postedBy | User |
| \_numberOfUpVotes | int |
| \_numberOfDownVotes | int |

d) Create the following methods in the **Post** Class

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public static Post CreatePost(string detail,User user) | This method accepts a string which contains post details separated by commas and the User object. Split the details and create a post object from the details and return it. |
| public static SortedDictionary<string,int> CalculatePostCount(List<Post> list) | This method accepts post list as argument. It iterates the list to store the type as key and count of post type as value in the SortedDictionary and returns it. |
| public static String PredictTrending(SortedDictionary<string,int> postCount) | This method accepts the postCount SortedDictionary as argument and returns the current trending topic based on the count of post types. |

The note details should be given as a comma-separated value in the below order,  
**\_id,\_content,\_type,\_postedTime,\_postedBy,\_numberOfUpVotes,\_numberOfDownVotes**  
  
Mark all the attributes as private.  
Include appropriate properties.  
Add a default constructor and a parameterized constructor to take in all attributes in the given order:   
**Post(long \_id, string \_content, string \_type, DateTime \_postedTime,User \_postedBy, int \_numberOfUpVotes, int \_numberOfDownVotes)** and   
**User(long \_id, string \_name, string \_emailId, DateTime \_dateOfBirth)**  
  
**Input Format:**  
While getting the input for a Post, the first line of input corresponds to User details and the next line corresponds to Post details.  
  
**Example:**  
i)Below is a user detail (\_id,\_name,\_emailId,\_dateOfBirth)  
**1004,morsh,morsh@yahoo.com,26-01-1997**  
  
ii)Below is a post detail (\_id,\_content, \_type,\_postedTime,\_numberOfUpVotes,\_numberOfDownVotes)  
**8022,Java stuck in a simple for loop,Java,13-01-2018 12:24:02,100,10**  
  
**Note:** There will be only one type which will be trending   
  
Create a driver class **Main** to display the Current trending topic.  
**[Strictly adhere to the Object-Oriented Specifications given in the problem statement.  
All class names, attribute names and method names should be the same as specified in the problem statement.]**  
  
**[All text in bold corresponds to the input and rest corresponds to the output]**  
**Sample Input and Output 1:**

Enter the number of posts

**4**

Enter detail 1:

**1004,morsh,morsh@yahoo.com,26-01-1997**

**6099,Code for limiting password characters,Java,09-06-2016 08:45:22,2000,50**

Enter detail 2:

**1010,saminathan,saminathan@rediff.com,12-05-1997**

**7523,How do I create matrix in MATLAB,Matlab,31-12-2017 10:12:13,200,10**

Enter detail 3:

**1007,meyyappan,meyyappan@gmail.com,30-03-1997**

**6562,Thresholding a signal using hysteresis,Matlab,03-03-2016 11:12:45,3506,200**

Enter detail 4:

**1010,saminathan,saminathan@rediff.com,12-05-1997**

**8345,Using camera calibration to segment the region,Matlab,05-01-2018 23:55:56,245,30**

Current trending topic is Matlab