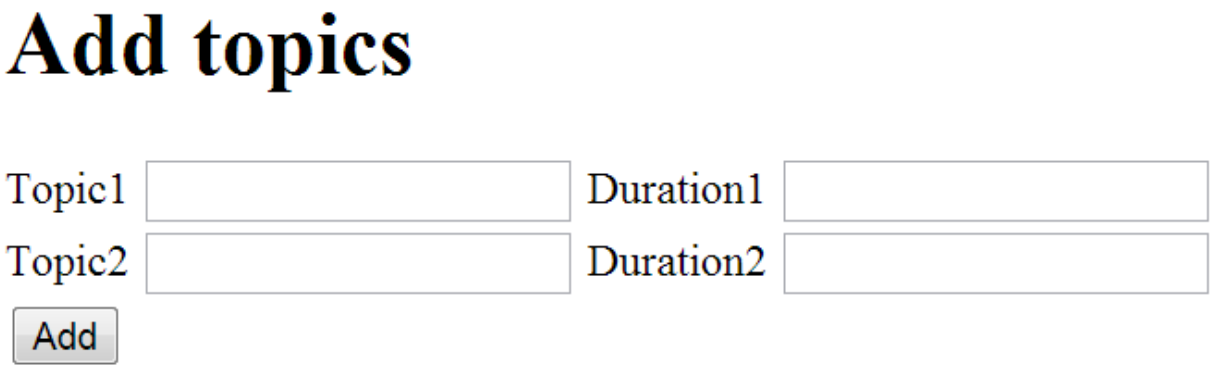
Create a controller **'SessionsController'**. This controller will be used to maintain a list of topics along with their duration and an auto-generated unique id. You should be able to add, list and show topics.

The application can be split into a **RESTful** approach and a **form-based** approach.

1) For the form-based approach, you can just create a simple page to add the topics. You can have a form built using Spring Form tags to add one or two topics at a time as shown here.



When you add the topics just display a message in the same page say, "Topics added successfully"

2) You can implement the add, show, list operations in a RESTful style as shown below.

- If you enter the URL **http://<server>/topic/add/Ruby/60** Ruby course gets added to the list with a unique id and you get a message "Ruby successfully added to db: Id is 101".

  If you enter the URL **http://<server>/topic/add/Groovy/45** Groovy course gets added to the list with a unique id and you get a message "Groovy successfully added to db; Id is 102"

- **http://<server>/topic/show/101.** The output is "Ruby(60 mins)"
- If you enter the URL **http://<server>/topic/list** the output is a JSON formatted list {"topics":[{title:"Ruby",duration:60},{"title":"Groovy",duration:45}]}

You will use the table 'topics' in any database in mysql.

```
create table topics(
        id int(5) not null primary key auto_increment,
        title varchar(50),
        duration int(4)
);
```

Create ConferenceService and TopicDao classes. Create additional classes and interfaces appropriately. Configure these beans in applicationContext.xml file. Provide a declarative transaction support for adding one or two courses at the same time.