Learning Questions of Week 4:

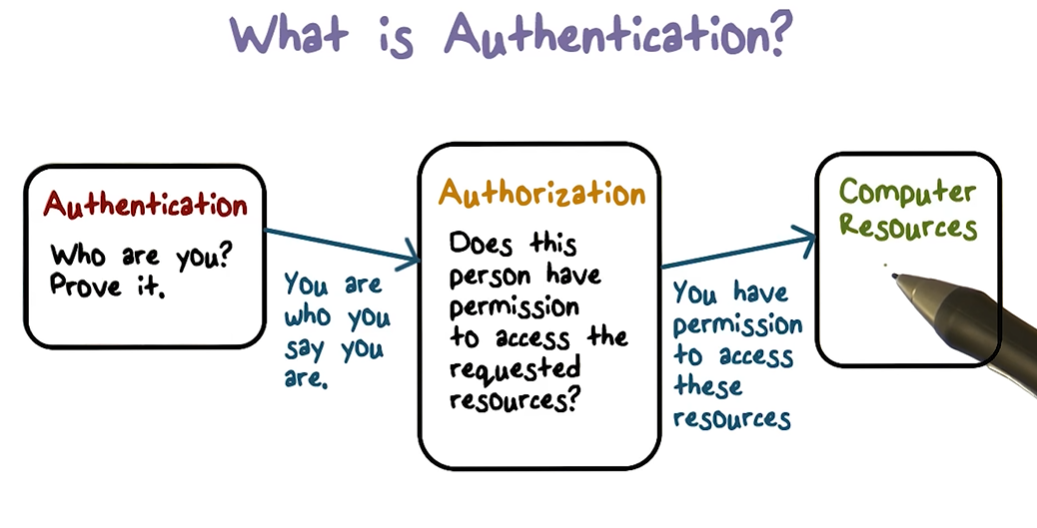
1. What is the [**Authentication**](https://en.wikipedia.org/wiki/Authentication) and why we would need it? (Who is user)

Authentication refers to the process of asserting and verifying the identity of a computer system user in order to make assure that a valid user is accessing to the correct and relevant data, information and resources where usually be asked for in the start-up of our systems.

Since for the sake of keeping system secure, it is necessary to know who makes requests, authentication be used to prevent non-valid parties to access different kind of data and resources.

In cyber security research, it is advised that a user authenticate his identity using following possible factors which preferably should more than one to become Multi-factor authentication rather than Single-factor one. Possible factors are:

1. **Knowledge Factor:** Information that only user knows like password, partial password, pass phrase, personal identification number (PIN), security questions of pattern recognition.
2. **Ownership Factor:** Something that user has like Id Card, security token, built in implemented devices or software and hardware token on specific device.
3. **Inheritance Factor:** Something that only belongs to user and is the part of its body or physical characteristics like fingerprint, face or voice recognition or DNA sequence
4. **Location Factor:** Things that belongs to specific location like place that we are calling or the IP address of a system that is requesting for access or modification.

[](https://www.youtube.com/watch?v=927KdwZZoU0)

1. What is [**Authorization**](https://en.wikipedia.org/wiki/Authorization) and why we would need it? (What user has access to)

Authorization is the process of permission checking based on defined access control policies for a set of resources to determine if an authenticated user has the enough permission to access to requested data or perform specific operations or not. Authorization usually being done after the successful authentication process

1. How to implement authentication? ([Good Guide](https://stackabuse.com/implementing-user-authentication-the-right-way/))

[](https://www.youtube.com/watch?v=ORus3-By4lk)To implement knowledge one we can save username and password in database and then compare submitted credential with the stored one to check the validity of user.

We should also take care that the user cannot choose insecure passwords by posing requirement of the password having some characteristics like minimum length, using different kind of characters. Also, the password should be saved in database in hashed format rather than plain text for the sake of preventing leaks.

Also the data should be send using POST method of an HTTP request and transmission should be done using HTTPS protocol to be safely and secured.

1. How to implement authorization? ([Gates and Policies](https://laravel.com/docs/7.x/authorization))

Laravel provide the opportunity to implement authorization using two primary ways: **Gates or Policies** where gates provide a simple closure-based approach to authorization while policies like controllers group their logic around a model or resource.

[](https://www.youtube.com/watch?v=bw3gKdIN1R0)

[](https://www.youtube.com/watch?v=XkOghdp84Pk)

In most of applications, a mixture of both would be used exclusively. Gates are most applicable to actions which are not related to any model or resource such viewing administrative dashboard while policies should be use when we want to authorize for specific action for particular model.

1. How to bind a model to a form?

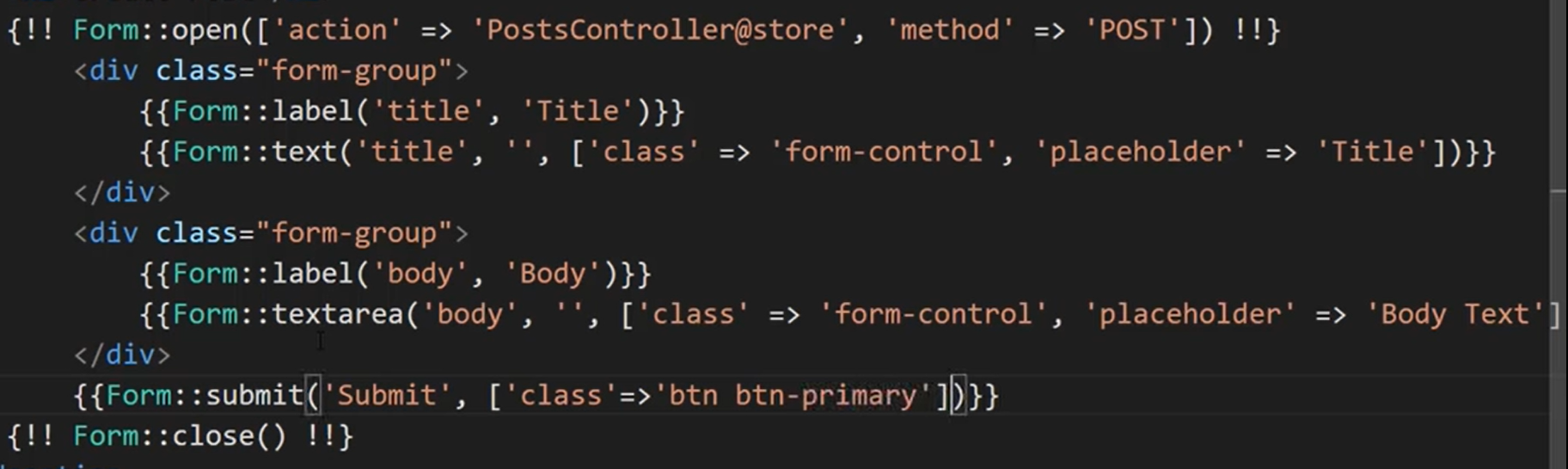
We can use form model binding to associate a form with one of our application’s model and automatically perform following things:

1. matches inputs named after model fields
2. populates the form’s fields with an existing model object’s data (if you’re editing and existing object)
3. repopulates the form with session data (say, if you’ve redirected to show validation errors after a failed object creation)

We can do this by installing Form and HTML library in order to create and interact with forms. The tutorial who to use it and add forms to Laravel application can be find in following [link](https://selftaughtcoders.com/from-idea-to-launch/lesson-23/laravel-5-application-form-model-binding-laravelcollective-forms-html-library-bootstrap-framework/) and [link](https://www.youtube.com/watch?v=-QapNzUE4V0&list=PLillGF-RfqbYhQsN5WMXy6VsDMKGadrJ-&index=7).

1. How we can to retrieve user input?

The user data would be sent through post method of using form which can be passed to a method of one of the controllers which we have specified as a request in order to be processed and save into database.



1. How can we ‘secure/monkey proof the form post & input?

Laravel has a validate method which can check for the inputs of a post of a request in order to prevent any insecurity like SQL injection or the penetration methods.

Also we can also use [CSRF](https://laravel.com/docs/5.5/csrf) token field in the form where stores it the values of a form like a session variable and then validate the request using it middleware when we submit it. CSRF token is used to verify that the authenticated user is the one actually making the requests to the application.

1. How we can CRUD the form input to the table?

After passing the information of through post method to one of the methods of specified controllers we can use create new object and then save it using tinker ORM or based on the data retrieve a row from the table and do modifications like deleting or updating and then save those operation to table again using tinker

1. What kind of `responses / replies` can a form post have? ([Reference](https://laravel.com/docs/4.2/responses#basic-responses))

A response can be either a plain text, http response, JSON response or redirection to another page using returning a view or returing response macros.