#### Questions for this assignment

What is Controller?

What is an Action Method?

Explain different types of Action Results in asp.net core?

What’s the HttpContext object? How can you access it within a Controller?

What is Controller?

Controller is a class that is used to group-up a set of action methods.

Action methods do perform certain operation when a request is received & returns the IActionResult that can be sent as response to browser.

It performs the following tasks:

* Reading requests such as receiving query string parameters, request body, request cookies, request headers etc.
* Validation of the request details.
* Invoking models (business logic)
* Creating objects of ViewModel or DTO
* Sending DTO objects to view (in case of view result)

The controller class should be either or both:

* The class name should be suffixed with “Controller”. Eg: HomeController
* The [Controller] attribute is applied to the same class or to its base class.

What is an Action Method?

An action method is a method in a controller class with the following restrictions:

* It must be public. Private or protected methods are not allowed.
* It cannot be overloaded (unless you use different Http methods or different action names).
* It cannot be a static method.
* An action method executes an action in response to an HTTP request.

Explain different types of Action Results in asp.net core?

Controller is a class that is used to group-up a set of action methods.

**IActionResult**

Defines a contract that represents the result of an action method.

**ActionResult**

A default implementation of IActionResult.

**ContentResult**

Represents a text result.

**EmptyResult**

Represents an ActionResult that when executed will do nothing.

**JsonResult**

An action result which formats the given object as JSON.

**PartialViewResult**

Represents an ActionResult that renders a partial view to the response.

**ViewResult**

Represents an ActionResult that renders a view to the response.

**ViewComponentResult**

An IActionResult which renders a view component to the response.

**StatusCodeResult**

An IActionResult which sends a specific HTTP status code in response, without any response body.

**UnauthorizedResult**

An IActionResult which sends HTTP 401 status code in response, with / without any response body.

**BadRequestResult**

An IActionResult which sends HTTP 400 status code in response, with / without any response body.

**NotFoundResult**

An IActionResult which sends HTTP 404 status code in response, with / without any response body.

**ObjectResult**

An IActionResult which sends the data of the specified object in response body.

**FileResult**

An IActionResult which sends content of the specified file in response body.

**RedirectToActionResult**

An IActionResult which sends HTTP 301 or 302 status code in response to redirect the request to the specific action method.

**LocalRedirectResult**

An IActionResult which sends HTTP 301 or 302 status code in response to redirect the request to the specified local URL (within the same domain).

**RedirectResult**

An IActionResult which sends HTTP 301 or 302 status code in response to redirect the request to the specified local URL (within the same domain) or an external URL.

What’s the HttpContext object? How can you access it within a Controller?

HttpContext encapsulates all HTTP-specific information about an individual HTTP request. You can access this object in controllers by using the ControllerBase.HttpContext property.

The HttpContext object has properties such as Items, Request, Response, Session, User etc.