统一返回格式需要用到IResultFilter过滤器

public class AddHeaderFilterWithDi : IResultFilter

{

    private ILogger \_logger;

    public AddHeaderFilterWithDi (ILoggerFactory loggerFactory) {

        \_logger = loggerFactory.CreateLogger<AddHeaderFilterWithDi> ();

}

    // 这里我们使用 AjaxResponseBase （我们自己定义的对象）对象作为统一返回格式

    // AjaxResponse 是 AjaxResponseBase 的默认实现

    // 如果返回结果不是一个 AjaxResponseBase 对象，那么我们就使用 AjaxResponse 对该结果进行封装

    public void OnResultExecuting (ResultExecutingContext context) {

        if (!context.ActionDescriptor.IsControllerAction ()) {

            return;

        }

        // 对象结果

        if (context.Result is ObjectResult) {

            var objectResult = context.Result as ObjectResult;

            if (objectResult == null) {

                throw new ArgumentException ($"{nameof(context)} should be ObjectResult!");

            }

            // 如果返回的值不是一个 AjaxResponseBase 对象

            if (!(objectResult.Value is AjaxResponseBase)) {

                // 用 AjaxResponse 封装原先的结果，然后用 AjaxResponse 作为返回值

                objectResult.Value = new AjaxResponse (objectResult.Value);

                if (!objectResult.Formatters.Any (f => f is JsonOutputFormatter)) {

                    objectResult.Formatters.Add (

                        new JsonOutputFormatter (

                            \_serviceProvider.GetRequiredService<IOptions<MvcJsonOptions>> ().Value.SerializerSettings,

                            \_serviceProvider.GetRequiredService<ArrayPool<char>> ()

                        )

                    );

                }

            }

        }

        //  json结果

        if (context.Result is JsonResult) {

            var jsonResult = context.Result as JsonResult;

            if (jsonResult == null) {

                throw new ArgumentException ($"{nameof(context)} should be JsonResult!");

            }

            // 如果返回的值不是一个 AjaxResponseBase 对象

            if (!(jsonResult.Value is AjaxResponseBase)) {

                jsonResult.Value = new AjaxResponse (jsonResult.Value);

            }

        }

        // 空结果

        if (context.Result is EmptyResult) {

            context.Result = new ObjectResult (new AjaxResponse ());

        }

}

    public void OnResultExecuted (ResultExecutedContext context) {

        // Can't add to headers here because response has already begun.

    }

}