

# Namespace API.ApiDoc.Tags.Requests

## Classes

[AddTagCommandRequestExample](#)

# Class AddTagCommandRequestExample

Namespace: [API.ApiDoc.Tags.Requests](#)

Assembly: API.dll

```
public class AddTagCommandRequestExample : IExamplesProvider<AddTagCommand>
```

## Inheritance

object ← AddTagCommandRequestExample

## Implements

IExamplesProvider<[AddTagCommand](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### GetExamples()

```
public AddTagCommand GetExamples()
```

## Returns

[AddTagCommand](#)

# Namespace API.Controllers

## Classes

[LookupController](#)

# Class LookupController

Namespace: [API.Controllers](#)

Assembly: API.dll

```
[Route("api/[controller]")]
public class LookupController : ControllerBase
```

## Inheritance

object ← ControllerBase ← LookupController

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### LookupController(IMediator)

```
public LookupController(IMediator mediator)
```

## Parameters

mediator IMediator

## Methods

### AddLangugue(AddLanguageCommand)

```
[HttpPost("language", Name = "AddLangugue")]
public Task<IActionResult> AddLangugue(AddLanguageCommand command)
```

## Parameters

command [AddLanguageCommand](#)

Returns

[Task](#) <IActionResult>

## GetLangugues()

```
[HttpGet("language", Name = "AddLangugue")]
public Task<IActionResult> GetLanguges()
```

Returns

[Task](#) <IActionResult>

# Namespace API.Controllers.V1

## Classes

[CategoryController](#)

[QuestionsController](#)

[SourceController](#)

[SystemController](#)

[TagController](#)

# Class CategoryController

Namespace: [API.Controllers.V1](#)

Assembly: API.dll

```
[Route("api/category")]
public class CategoryController : ControllerBase
```

## Inheritance

object ← ControllerBase ← CategoryController

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### CategoryController(IMediator, IHttpResultResponder)

```
public CategoryController(IMediator mediator, IHttpResultResponder httpResultResponder)
```

## Parameters

mediator IMediator

httpResultResponder [IHttpResultResponder](#)

## Methods

### AddCateogry(AddCategoryCommand)

```
[HttpPost(Name = "AddCateogry")]
public Task<IActionResult> AddCateogry(AddCategoryCommand command)
```

## Parameters

command [AddCategoryCommand](#)

Returns

[Task](#) <[IActionResult](#)>

## GetAllCategory(GetAllCategoryQuery)

```
[HttpGet(Name = "GetAllCategory")]
public Task<IActionResult> GetAllCategory(GetAllCategoryQuery query)
```

Parameters

query [GetAllCategoryQuery](#)

Returns

[Task](#) <[IActionResult](#)>

## GetCategoryById(GetCategoryByIdQuery)

```
[HttpGet("{id:guid}", Name = "GetCategoryById")]
public Task<IActionResult> GetCategoryById(GetCategoryByIdQuery query)
```

Parameters

query [GetCategoryByIdQuery](#)

Returns

[Task](#) <[IActionResult](#)>

## GetCategoryTimeline(Guid, GetEntityTimelineQuery)

```
[HttpGet("{categoryId:guid}/timeline", Name = "GetCategoryTimeline")]
[EndpointName("GetCategoryTimeline")]
[EndpointSummary("Tag Timeline")]
[EndpointDescription("get the timeline for tag")]
```

```
public Task<IActionResult> GetCategoryTimeline(Guid categoryId,  
GetEntityTimelineQuery query)
```

Parameters

categoryId [Guid](#)

query [GetEntityTimelineQuery](#)

Returns

[Task](#) <[IActionResult](#)>

## GetCategoryTimelineDetails(Guid, Guid)

```
[HttpGet("{categoryId:guid}/timeline/{timelineId:guid}", Name  
= "GetCategoryTimelineDetails")]  
[EndpointName("GetCategoryTimelineDetails")]  
[EndpointSummary("Tag Timeline details")]  
[EndpointDescription("get the timeline for tag")]  
public Task<IActionResult> GetCategoryTimelineDetails(Guid categoryId, Guid timelineId)
```

Parameters

categoryId [Guid](#)

timelineId [Guid](#)

Returns

[Task](#) <[IActionResult](#)>

## UpdateCategory(Guid, UpdateCategoryCommand)

```
[HttpPut("{categoryId:guid}", Name = "UpdateCategory")]  
public Task<IActionResult> UpdateCategory(Guid categoryId, UpdateCategoryCommand command)
```

Parameters

categoryId [Guid](#)

command [UpdateCategoryCommand](#)

Returns

[Task](#) <[IActionResult](#)>

## UpdateParrentRefrence(Guid, UpdateCategoryCommand)

```
[HttpPut("{categoryId:guid}/parent-reference", Name = "UpdateParrentRefrence")]
public Task<IActionResult> UpdateParrentRefrence(Guid categoryId,
UpdateCategoryCommand command)
```

Parameters

categoryId [Guid](#)

command [UpdateCategoryCommand](#)

Returns

[Task](#) <[IActionResult](#)>

# Class QuestionsController

Namespace: [API.Controllers.V1](#)

Assembly: API.dll

```
[Route("api/[controller]")]
[ApiController]
public class QuestionsController : ControllerBase
```

## Inheritance

object ← ControllerBase ← QuestionsController

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### QuestionsController(IMediator)

```
public QuestionsController(IMediator mediator)
```

## Parameters

mediator IMediator

## Methods

### AddQuestion(AddQuestionCommand)

```
[HttpPost]
public Task<IActionResult> AddQuestion(AddQuestionCommand command)
```

## Parameters

command [AddQuestionCommand](#)

Returns

[Task](#) <IActionResult>

## DeleteAllQuestion()

```
[HttpDelete(Name = "DeleteAllQuestion")]
public Task<IActionResult> DeleteAllQuestion()
```

Returns

[Task](#) <IActionResult>

## DeleteQuestion(Guid)

```
[HttpDelete("{id}", Name = "DeleteQuestion")]
public Task<IActionResult> DeleteQuestion(Guid id)
```

Parameters

**id** [Guid](#)

Returns

[Task](#) <IActionResult>

## GetAllQuestions(GetAllQuestionsQuery)

```
[HttpGet]
public Task<IActionResult> GetAllQuestions(GetAllQuestionsQuery query)
```

Parameters

**query** [GetAllQuestionsQuery](#)

Returns

[Task](#) <IActionResult>

## GetQuestionById(Guid)

```
[HttpGet("{id}")]
public Task<IActionResult> GetQuestionById(Guid id)
```

Parameters

**id** [Guid](#)

Returns

[Task](#) <IActionResult>

## UpdateQuestion(UpdateQuestionCommand)

```
[HttpPut(Name = "UpdateQuestion")]
public Task<IActionResult> UpdateQuestion(UpdateQuestionCommand command)
```

Parameters

**command** [UpdateQuestionCommand](#)

Returns

[Task](#) <IActionResult>

# Class SourceController

Namespace: [API.Controllers.V1](#)

Assembly: API.dll

```
[Route("api/source")]
public class SourceController : ControllerBase
```

## Inheritance

object ← ControllerBase ← SourceController

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### SourceController(IMediator, IHttpResultResponder)

```
public SourceController(IMediator mediator, IHttpResultResponder resultResponder)
```

## Parameters

mediator [IMediator](#)

resultResponder [IHttpResultResponder](#)

## Methods

### AddBulkNewTags(Guid, AddBulkTagsToSourceCommand)

```
[HttpPost("{sourceId:guid}/tag/bulk")]
public Task<IActionResult> AddBulkNewTags(Guid sourceId, AddBulkTagsToSourceCommand command)
```

## Parameters

sourceId [Guid](#)

command [AddBulkTagsToSourceCommand](#)

Returns

[Task](#) <ActionResult>

## AddNewTag(Guid, Guid)

```
[HttpPost("{sourceId:guid}/tag/{tagId:guid}")]
public Task<ActionResult> AddNewTag(Guid sourceId, Guid tagId)
```

Parameters

sourceId [Guid](#)

tagId [Guid](#)

Returns

[Task](#) <ActionResult>

## AddReference(AddSourceCommand)

```
[HttpPost("{id:guid}/reference", Name = "AddReference")]
public Task<ActionResult> AddReference(AddSourceCommand command)
```

Parameters

command [AddSourceCommand](#)

Returns

[Task](#) <ActionResult>

## AddSource(AddSourceCommand)

```
[HttpPost(Name = "AddSource")]
public Task<IActionResult> AddSource(AddSourceCommand command)
```

Parameters

command [AddSourceCommand](#)

Returns

[Task](#)<[IActionResult](#)>

## GetAllSources(GetAllSourceQuery)

```
[HttpGet(Name = "GetAllSources")]
[ProducesResponseType(200, Type = typeof(PageResponse< GetAllSourceQueryResult >))]
public Task<IActionResult> GetAllSources(GetAllSourceQuery query)
```

Parameters

query [GetAllSourceQuery](#)

Returns

[Task](#)<[IActionResult](#)>

## GetSourceById(Guid)

```
[HttpGet("{id:guid}", Name = "GetSourceById")]
[ProducesResponseType(200, Type = typeof(PageResponse< GetSourceByIdQueryResult >))]
public Task<IActionResult> GetSourceById(Guid id)
```

Parameters

id [Guid](#)

Returns

[Task](#) <IActionResult>

## GetSourceTimeline(Guid, GetEntityTimelineQuery)

```
[HttpGet("{id:guid}/timeline", Name = "GetSourceTimeline")]
[EndpointName("GetSourceTimeline")]
[EndpointSummary("Tag Timeline")]
[EndpointDescription("get the timeline for tag")]
public Task<IActionResult> GetSourceTimeline(Guid id, GetEntityTimelineQuery query)
```

Parameters

**id** [Guid](#)

**query** [GetEntityTimelineQuery](#)

Returns

[Task](#) <IActionResult>

## GetSourceTimelineDetails(Guid, Guid, EntityTimelineDetailsQuery)

```
[HttpGet("{id:guid}/timeline/{timelineId:guid}", Name = "GetSourceTimelineDetails")]
[EndpointName("GetSourceTimelineDetails")]
[EndpointSummary("Tag Timeline details")]
[EndpointDescription("get the timeline for tag")]
public Task<IActionResult> GetSourceTimelineDetails(Guid id, Guid timelineId,
EntityTimelineDetailsQuery query)
```

Parameters

**id** [Guid](#)

**timelineId** [Guid](#)

**query** [EntityTimelineDetailsQuery](#)

Returns

[Task](#) <IActionResult>

## RemoveTag(RemoveTagFromSourceCommand)

```
[HttpDelete("{sourceId:guid}/tag/{tagId:guid}")]
public Task<IActionResult> RemoveTag(RemoveTagFromSourceCommand command)
```

### Parameters

command [RemoveTagFromSourceCommand](#)

### Returns

[Task](#) <IActionResult>

## RemoveTag(Guid, RemoveBulkTagFromSourceCommand)

```
[HttpDelete("{sourceId:guid}/tag/bulk")]
public Task<IActionResult> RemoveTag(Guid sourceId, RemoveBulkTagFromSourceCommand command)
```

### Parameters

sourceId [Guid](#)

command [RemoveBulkTagFromSourceCommand](#)

### Returns

[Task](#) <IActionResult>

# Class SystemController

Namespace: [API.Controllers.V1](#)

Assembly: API.dll

```
[Route("api/system")]
public class SystemController : ControllerBase
```

## Inheritance

object ← ControllerBase ← SystemController

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

SystemController(ISystemManager, IMediator,  
IHttpResponseResponder)

```
public SystemController(ISystemManager systemManager, IMediator mediator,
IHttpResponseResponder resultResponder)
```

## Parameters

systemManager [ISystemManager](#)

mediator IMediator

resultResponder [IHttpResponseResponder](#)

## Methods

Debug()

```
[HttpPost("logs/debug")]
public IActionResult Debug()
```

Returns

[IActionResult](#)

## Disable()

```
[HttpPost("logs/disable")]
public IActionResult Disable()
```

Returns

[IActionResult](#)

## Enable()

```
[HttpPost("logs/enable")]
public IActionResult Enable()
```

Returns

[IActionResult](#)

## GetDeletedEntities(GetDeletedEntitiesQuery)

```
[HttpGet("deletedEntity")]
public Task<IActionResult> GetDeletedEntities(GetDeletedEntitiesQuery query)
```

Parameters

query [GetDeletedEntitiesQuery](#)

Returns

[Task](#) <[IActionResult](#)>

## GetLogs(LogSearchQuery)

```
[HttpGet("logs")]
public Task<IActionResult> GetLogs(LogSearchQuery query)
```

### Parameters

query [LogSearchQuery](#)

### Returns

[Task](#) <[IActionResult](#)>

## GetSystemStatus()

```
[HttpGet("status")]
public Task<IActionResult> GetSystemStatus()
```

### Returns

[Task](#) <[IActionResult](#)>

## Status()

```
[HttpGet("logs/status")]
public IActionResult Status()
```

### Returns

[IActionResult](#)

# Class TagController

Namespace: [API.Controllers.V1](#)

Assembly: API.dll

```
[Route("api/tag")]
public class TagController : ControllerBase
```

## Inheritance

object ← ControllerBase ← TagController

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

TagController(IMediator, IHttpResultResponder, IRepositoryManager)

```
public TagController(IMediator mediator, IHttpResultResponder resultResponder,
IRepositoryManager repositoryManager)
```

## Parameters

mediator IMediator

resultResponder [IHttpResultResponder](#)

repositoryManager [IRepositoryManager](#)

## Methods

AddTag(AddTagCommand)

```
[HttpPost(Name = "AddTag")]
[SwaggerRequestExample(typeof(AddTagCommand), typeof(AddTagCommandRequestExample))]
```

```
public Task<IActionResult> AddTag(AddTagCommand command)
```

Parameters

command [AddTagCommand](#)

Returns

[Task](#) <IActionResult>

## ArchiveTag(ArchiveTagCommand)

```
[HttpPost("{id:guid}/archive", Name = "ArchiveTag")]
[EndpointName("ArchiveTag")]
[EndpointSummary("Tag Archive")]
[EndpointDescription("Description")]
public Task ArchiveTag(ArchiveTagCommand command)
```

Parameters

command [ArchiveTagCommand](#)

Returns

[Task](#)

## DeleteTag(DeleteTagCommand)

```
[HttpDelete("{id:guid}", Name = "DeleteTag")]
public Task<IActionResult> DeleteTag(DeleteTagCommand command)
```

Parameters

command [DeleteTagCommand](#)

Returns

[Task](#) <IActionResult>

## ExportTags(ExportTagsQuery)

```
[HttpGet("export")]
public Task<IActionResult> ExportTags(ExportTagsQuery query)
```

Parameters

query [ExportTagsQuery](#)

Returns

[Task](#) <[IActionResult](#)>

## GetAllTags(GetAllTagsQuery)

```
[HttpGet(Name = "GetAllTags")]
public Task<IActionResult> GetAllTags(GetAllTagsQuery query)
```

Parameters

query [GetAllTagsQuery](#)

Returns

[Task](#) <[IActionResult](#)>

## GetTagTimeline(Guid, GetEntityTimelineQuery)

```
[HttpGet("{id:guid}/timeline", Name = "GetTagTimeline")]
[EndpointName("GetTagTimeline")]
[EndpointSummary("Tag Timeline")]
[EndpointDescription("get the timeline for tag")]
public Task<IActionResult> GetTagTimeline(Guid id, GetEntityTimelineQuery query)
```

Parameters

[id](#) [Guid](#)

[query](#) [GetEntityTimelineQuery](#)

Returns

[Task](#) <[IActionResult](#)>

## GetTagTimelineDetails(Guid, Guid)

```
[HttpGet("{tagId:guid}/timeline/{timelineId:guid}", Name = "GetTagTimelineDetails")]
[EndpointName("GetTagTimelineDetails")]
[EndpointSummary("Tag Timeline details")]
[EndpointDescription("get the timeline for tag")]
public Task<IActionResult> GetTagTimelineDetails(Guid tagId, Guid timelineId)
```

Parameters

[tagId](#) [Guid](#)

[timelineId](#) [Guid](#)

Returns

[Task](#) <[IActionResult](#)>

## ImportTags(IFormFile)

```
[HttpPost("import")]
public Task<IActionResult> ImportTags(IFormFile file)
```

Parameters

[file](#) [IFormFile](#)

Returns

[Task](#) <[IActionResult](#)>

## TagAutoComplete(AutoCompleteTagsQuery)

```
[HttpGet("autocomplete")]
public Task<IActionResult> TagAutoComplete(AutoCompleteTagsQuery query)
```

### Parameters

query [AutoCompleteTagsQuery](#)

### Returns

[Task](#)<[IActionResult](#)>

## TagDetails(Guid)

```
[HttpGet("{id:guid}", Name = "TagDetails")]
[EndpointName("TagDetails")]
[EndpointSummary("Tag Details")]
[EndpointDescription("Tag Details")]
public Task<IActionResult> TagDetails(Guid id)
```

### Parameters

id [Guid](#)

### Returns

[Task](#)<[IActionResult](#)>

## TagRelatedSources(Guid, GetRelatedSourcesQuery)

```
[HttpGet("{tagId:guid}/related-sources", Name = "TagRelatedSources")]
[EndpointName("TagRelatedSources")]
[EndpointSummary("Tag related sources")]
[EndpointDescription("Tag related sources")]
public Task<IActionResult> TagRelatedSources(Guid tagId, GetRelatedSourcesQuery query)
```

Parameters

tagId [Guid](#)

query [GetRelatedSourcesQuery](#)

Returns

[Task](#) <[IActionResult](#)>

## UnarchiveTag(UnarchiveTagCommand)

```
[HttpPost("{id:guid}/unarchive", Name = "UnarchiveTag")]
[EndpointName("UnarchiveTag")]
[EndpointSummary("Tag Unarchive")]
[EndpointDescription("Description")]
public Task<IActionResult> UnarchiveTag(UnarchiveTagCommand command)
```

Parameters

command [UnarchiveTagCommand](#)

Returns

[Task](#) <[IActionResult](#)>

## UpdateTag(UpdateTagCommand)

```
[HttpPut(Name = "UpdateTag")]
public Task<IActionResult> UpdateTag(UpdateTagCommand command)
```

Parameters

command [UpdateTagCommand](#)

Returns

[Task](#) <[IActionResult](#)>

# Namespace API.Filters

## Classes

[ExampleSchemaFilter](#)

# Class ExampleSchemaFilter

Namespace: [API.Filters](#)

Assembly: API.dll

```
public class ExampleSchemaFilter : ISchemaFilter
```

## Inheritance

object ← ExampleSchemaFilter

## Implements

[ISchemaFilter](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### Apply(OpenApiSchema, SchemaFilterContext)

```
public void Apply(OpenApiSchema schema, SchemaFilterContext context)
```

## Parameters

schema [OpenApiSchema](#)

context [SchemaFilterContext](#)

# Namespace API.Interfaces

## Interfaces

[IHttpResultResponder](#)

# Interface IHttpResultResponder

Namespace: [API.Interfaces](#)

Assembly: API.dll

```
public interface IHttpResultResponder
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Methods

### FromResult(HttpContext, Result)

```
IActionResult FromResult(HttpContext context, Result result)
```

#### Parameters

context [HttpContext](#)

result [Result](#)

#### Returns

[IActionResult](#)

### FromResult<T>(HttpContext, Result<T>)

```
IActionResult FromResult<T>(HttpContext context, Result<T> result)
```

#### Parameters

context [HttpContext](#)

result [Result](#)<T>

Returns

[IActionResult](#) ↗

Type Parameters

T

# Namespace API.Middleware

## Classes

[CorrelationIdMiddleware](#)

[LocalizationMiddleware](#)

[ResultMiddleware](#)

# Class CorrelationIdMiddleware

Namespace: [API.Middleware](#)

Assembly: API.dll

```
public class CorrelationIdMiddleware
```

## Inheritance

object ← CorrelationIdMiddleware

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### Invoke(HttpContext, RequestDelegate)

```
public Task Invoke(HttpContext context, RequestDelegate next)
```

#### Parameters

context [HttpContext](#)

next [RequestDelegate](#)

#### Returns

[Task](#)

# Class LocalizationMiddleware

Namespace: [API.Middleware](#)

Assembly: API.dll

```
public class LocalizationMiddleware : IMiddleware
```

## Inheritance

object ← LocalizationMiddleware

## Implements

[IMiddleware](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### InvokeAsync(HttpContext, RequestDelegate)

Request handling method.

```
public Task InvokeAsync(HttpContext context, RequestDelegate next)
```

#### Parameters

**context** [HttpContext](#)

The [HttpContext](#) for the current request.

**next** [RequestDelegate](#)

The delegate representing the remaining middleware in the request pipeline.

#### Returns

[Task](#)

A [Task](#) that represents the execution of this middleware.



# Class ResultMiddleware

Namespace: [API.Middleware](#)

Assembly: API.dll

```
public class ResultMiddleware
```

## Inheritance

object ← ResultMiddleware

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### ResultMiddleware(RequestDelegate)

```
public ResultMiddleware(RequestDelegate next)
```

## Parameters

next [RequestDelegate](#)

## Methods

### Invoke(HttpContext)

```
public Task Invoke(HttpContext context)
```

## Parameters

context [HttpContext](#)

## Returns

## Task ↗

# Namespace API.Services

## Classes

[HttpResultResponder](#)

# Class HttpResponseMessage

Namespace: [API.Services](#)

Assembly: API.dll

```
public class HttpResponseMessage : IHttpActionResult
```

## Inheritance

object ← HttpResponseMessage

## Implements

[IHttpActionResult](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### FromResult(HttpContext, Result)

```
public IActionResult FromResult(HttpContext context, Result result)
```

#### Parameters

context [HttpContext](#)

result [Result](#)

#### Returns

[IActionResult](#)

### FromResult<T>(HttpContext, Result<T>)

```
public IActionResult FromResult<T>(HttpContext context, Result<T> result)
```

Parameters

`context HttpContext`

`result Result<T>`

Returns

[IActionResult](#)

Type Parameters

`T`

# Namespace Application

## Classes

[ApplicationRegisterServices](#)

# Class ApplicationRegisterServices

Namespace: [Application](#)

Assembly: Application.dll

```
public static class ApplicationRegisterServices
```

## Inheritance

object ← ApplicationRegisterServices

## Methods

### RegisterApplication(IServiceCollection)

```
public static IServiceCollection RegisterApplication(this IServiceCollection services)
```

#### Parameters

services [IServiceCollection](#) ↗

#### Returns

[IServiceCollection](#) ↗

# Namespace Application.Auditing.Application Logs.Questions.LogSearchQuery

## Classes

[LogSearchQuery](#)

[LogSearchQueryHandler](#)

# Class LogSearchQuery

Namespace: [Application.Auditing.ApplicationLogs.Queries.LogSearchQuery](#)

Assembly: Application.dll

```
public class LogSearchQuery : PageRequest, IRequest<PageResponse<ApplicationLog>>,  
IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← LogSearchQuery

## Implements

IRequest<[PageResponse](#)<[ApplicationLog](#)>>, IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## EndDate

```
public DateTimeOffset? EndDate { get; set; }
```

## Property Value

[DateTimeOffset](#)?

## Level

```
public LogEventLevel? Level { get; set; }
```

## Property Value

LogEventLevel?

## Message

```
public string? Message { get; set; }
```

Property Value

[string](#) ↗

## StartDate

```
public DateTimeOffset? StartDate { get; set; }
```

Property Value

[DateTimeOffset](#) ↗?

# Class LogSearchQueryHandler

Namespace: [Application.Auditing.ApplicationLogs.Queries.LogSearchQuery](#)

Assembly: Application.dll

```
public class LogSearchQueryHandler : IRequestHandler<LogSearchQuery,  
PageResponse<ApplicationLog>>
```

## Inheritance

object ← LogSearchQueryHandler

## Implements

IRequestHandler<[LogSearchQuery](#), [PageResponse](#)<[ApplicationLog](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### LogSearchQueryHandler(IAuditManager)

```
public LogSearchQueryHandler(IAuditManager auditManager)
```

## Parameters

auditManager [IAuditManager](#)

## Methods

### Handle(LogSearchQuery, CancellationToken)

Handles a request

```
public Task<PageResponse<ApplicationLog>> Handle(LogSearchQuery request,  
CancellationToken cancellationToken)
```

## Parameters

request [LogSearchQuery](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[PageResponse](#)<[ApplicationLog](#)>>

Response from the request

# Namespace Application.Auditing.EntitiesLog. Queries.GetDeletedEntities

## Classes

[GetDeletedEntitiesQuery](#)

[GetDeletedEntitiesQueryHandler](#)

[GetDeletedEntitiesQueryResult](#)

# Class GetDeletedEntitiesQuery

Namespace: [Application.Auditing.EntitiesLog.Questions.GetDeletedEntities](#)

Assembly: Application.dll

```
public class GetDeletedEntitiesQuery : PageRequest,  
 IRequest<Result<PageResponse<GetDeletedEntitiesQueryResult>>>, IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← GetDeletedEntitiesQuery

## Implements

IRequest<[Result<PageResponse<GetDeletedEntitiesQueryResult>>](#)>, IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## ActionBy

```
public ActionBy? ActionBy { get; set; }
```

## Property Value

[ActionBy?](#)

## Comment

```
public string? Comment { get; set; }
```

## Property Value

[string](#) ↴

## EndDate

```
public DateTimeOffset? EndDate { get; set; }
```

Property Value

[DateTimeOffset](#) ↴?

## EntityName

```
public EntityName? EntityName { get; set; }
```

Property Value

[EntityName](#)?

## StartDate

```
public DateTimeOffset? StartDate { get; set; }
```

Property Value

[DateTimeOffset](#) ↴?

# Class GetDeletedEntitiesQueryHandler

Namespace: [Application.Auditing.EntitiesLog.Questions.GetDeletedEntities](#)

Assembly: Application.dll

```
public class GetDeletedEntitiesQueryHandler : IRequestHandler<GetDeletedEntitiesQuery, Result<PageResponse<GetDeletedEntitiesQueryResult>>>
```

## Inheritance

object ← GetDeletedEntitiesQueryHandler

## Implements

IRequestHandler<[GetDeletedEntitiesQuery](#), [Result<PageResponse<GetDeletedEntitiesQueryResult>>](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### GetDeletedEntitiesQueryHandler(IAuditManager, IMapper)

```
public GetDeletedEntitiesQueryHandler(IAuditManager auditManager, IMapper mapper)
```

## Parameters

auditManager [IAuditManager](#)

mapper IMapper

## Methods

### Handle(GetDeletedEntitiesQuery, CancellationToken)

Handles a request

```
public Task<Result<PageResponse<GetDeletedEntitiesQueryResult>>>
```

```
Handle(GetDeletedEntitiesQuery request, CancellationToken cancellationToken)
```

## Parameters

request [GetDeletedEntitiesQuery](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) < [Result](#) < [PageResponse](#) < [GetDeletedEntitiesQueryResult](#) > >

Response from the request

# Class GetDeletedEntitiesQueryResult

Namespace: [Application.Auditing.EntitiesLog.Questions.GetDeletedEntities](#)

Assembly: Application.dll

```
public class GetDeletedEntitiesQueryResult
```

## Inheritance

object ← GetDeletedEntitiesQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### ActionBy

```
public ActionBy ActionBy { get; set; }
```

Property Value

[ActionBy](#)

### ActionDate

```
public DateTimeOffset ActionDate { get; set; }
```

Property Value

[DateTimeOffset](#)

### EntityId

```
public Guid EntityId { get; set; }
```

Property Value

[Guid](#) ↗

## EntityName

```
public EntityName EntityName { get; set; }
```

Property Value

[EntityName](#)

## VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int](#) ↗

# Namespace Application.Categories.Commands. AddCategory

## Classes

[AddCategoryCommand](#)

[AddCategoryCommandHandler](#)

[AddCategoryCommandResult](#)

# Class AddCategoryCommand

Namespace: [Application.Categories.Commands.AddCategory](#)

Assembly: Application.dll

```
public class AddCategoryCommand : IRequest<Result<AddCategoryCommandResult>>, IBaseRequest
```

## Inheritance

object ← AddCategoryCommand

## Implements

IRequest<[Result<AddCategoryCommandResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Description

```
public string? Description { get; set; }
```

#### Property Value

[string](#) ↗

### Name

```
public string Name { get; set; }
```

#### Property Value

[string](#) ↗

## ParentId

```
public Guid? ParentId { get; set; }
```

### Property Value

[Guid](#)?

# Class AddCategoryCommandHandler

Namespace: [Application.Categories.Commands.AddCategory](#).

Assembly: Application.dll

```
public class AddCategoryCommandHandler : IRequestHandler<AddCategoryCommand,  
Result<AddCategoryCommandResult>>
```

## Inheritance

object ← AddCategoryCommandHandler

## Implements

IRequestHandler<[AddCategoryCommand](#), [Result<AddCategoryCommandResult>](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### AddCategoryCommandHandler(IServiceManager, IMapper)

```
public AddCategoryCommandHandler(IServiceManager serviceManager, IMapper mapper)
```

## Parameters

serviceManager [IServiceManager](#)

mapper IMapper

## Methods

### Handle(AddCategoryCommand, CancellationToken)

Handles a request

```
public Task<Result<AddCategoryCommandResult>> Handle(AddCategoryCommand request,
```

```
CancellationToken cancellationToken)
```

## Parameters

request [AddCategoryCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[AddCategoryCommandResult](#)>>

Response from the request

# Class AddCategoryCommandResult

Namespace: [Application.Categories.Commands.AddCategory](#).

Assembly: Application.dll

```
public class AddCategoryCommandResult
```

## Inheritance

object ← AddCategoryCommandResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Description

```
public string? Description { get; set; }
```

#### Property Value

[string](#) ↗

### HasChildren

```
public bool HasChildren { get; set; }
```

#### Property Value

[bool](#) ↗

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

IsRoot

```
public bool IsRoot { get; }
```

Property Value

[bool](#) ↗

Level

```
public int Level { get; set; }
```

Property Value

[int](#) ↗

Name

```
public string Name { get; set; }
```

Property Value

[string](#) ↗

ParentId

```
public Guid? ParentId { get; set; }
```

Property Value

[Guid](#)?

# Namespace Application.Categories.Commands. UpdateCategory

## Classes

[UpdateCategoryCommand](#)

[UpdateCategoryCommandHandler](#)

# Class UpdateCategoryCommand

Namespace: [Application.Categories.Commands.UpdateCategory](#)

Assembly: Application.dll

```
public class UpdateCategoryCommand : IRequest<Result>, IBaseRequest
```

## Inheritance

object ← UpdateCategoryCommand

## Implements

IRequest<[Result](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## CategoryId

```
public Guid CategoryId { get; set; }
```

## Property Value

[Guid](#) ↗

## Description

```
public string? Description { get; set; }
```

## Property Value

[string](#) ↗

## Name

```
public string Name { get; set; }
```

## Property Value

[string](#) ↗

# Class UpdateCategoryCommandHandler

Namespace: [Application.Categories.Commands.UpdateCategory](#)

Assembly: Application.dll

```
public class UpdateCategoryCommandHandler : IRequestHandler<UpdateCategoryCommand, Result>
```

## Inheritance

object ← UpdateCategoryCommandHandler

## Implements

IRequestHandler<[UpdateCategoryCommand](#), [Result](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### UpdateCategoryCommandHandler(IServiceManager)

```
public UpdateCategoryCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(UpdateCategoryCommand, CancellationToken)

Handles a request

```
public Task<Result> Handle(UpdateCategoryCommand request, CancellationToken cancellationToken)
```

## Parameters

request [UpdateCategoryCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

Returns

[Task](#) <[Result](#)>

Response from the request

# Namespace Application.Categories.Commands. UpdateCategoryParentReference

## Classes

[UpdateCategoryParentRefrenceCommand](#)

# Class UpdateCategoryParentRefrenceCommand

Namespace: [Application.Categories.Commands.UpdateCategoryParentReference](#)

Assembly: Application.dll

```
public class UpdateCategoryParentRefrenceCommand
```

## Inheritance

object ← UpdateCategoryParentRefrenceCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### MyProperty

```
public int MyProperty { get; set; }
```

#### Property Value

[int](#)

# Namespace Application.Categories.Queries.GetAllCategory

## Classes

[GetAllCategoryQuery](#)

[GetAllCategoryQueryHandler](#)

[GetAllCategoryQueryResult](#)

# Class GetAllCategoryQuery

Namespace: [Application.Categories.Queries.GetAllCategory](#)

Assembly: Application.dll

```
public class GetAllCategoryQuery : PageRequest,  
 IRequest<PageResponse<GetAllCategoryQueryResult>>, IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← GetAllCategoryQuery

## Implements

IRequest<[PageResponse<GetAllCategoryQueryResult>](#)>, IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## HasChildren

```
public bool? HasChildren { get; set; }
```

## Property Value

[bool](#)?

## IsRoot

```
public bool? IsRoot { get; set; }
```

## Property Value

bool?

## Level

`public int? Level { get; set; }`

Property Value

int?

## Search

`public string? Search { get; set; }`

Property Value

string?

# Class GetAllCategoryQueryHandler

Namespace: [Application.Categories.Queries.GetAllCategory](#)

Assembly: Application.dll

```
public class GetAllCategoryQueryHandler : IRequestHandler<GetAllCategoryQuery,
PageResponse<GetAllCategoryQueryResult>>
```

## Inheritance

object ← GetAllCategoryQueryHandler

## Implements

IRequestHandler<[GetAllCategoryQuery](#), [PageResponse](#)<[GetAllCategoryQueryResult](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### GetAllCategoryQueryHandler(IRepositoryManager, IMapper)

```
public GetAllCategoryQueryHandler(IRepositoryManager repositoryManager, IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

mapper IMapper

## Methods

### Handle(GetAllCategoryQuery, CancellationToken)

Handles a request

```
public Task<PageResponse<GetAllCategoryQueryResult>> Handle(GetAllCategoryQuery request,
```

```
CancellationToken cancellationToken)
```

## Parameters

request [GetAllCategoryQuery](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[PageResponse](#)<[GetAllCategoryQueryResult](#)>>

Response from the request

# Class GetAllCategoryQueryResult

Namespace: [Application.Categories.Queries.GetAllCategory](#)

Assembly: Application.dll

```
public class GetAllCategoryQueryResult
```

## Inheritance

object ← GetAllCategoryQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### HasChildren

```
public bool HasChildren { get; set; }
```

#### Property Value

[bool](#) ↗

### Id

```
public Guid Id { get; set; }
```

#### Property Value

[Guid](#) ↗

### IsRoot

```
public bool IsRoot { get; }
```

Property Value

[bool ↗](#)

Level

```
public int Level { get; set; }
```

Property Value

[int ↗](#)

Name

```
public string Name { get; set; }
```

Property Value

[string ↗](#)

ParentId

```
public Guid? ParentId { get; set; }
```

Property Value

[Guid ↗?](#)

# Namespace Application.Categories.Queries.GetCategoryById

## Classes

[GetCategoryByIdQuery](#)

[GetCategoryByIdQueryHandler](#)

[GetCategoryByIdQueryResult](#)

# Class GetCategoryByIdQuery

Namespace: [Application.Categories.Queries.GetCategoryById](#)

Assembly: Application.dll

```
public class GetCategoryByIdQuery : IRequest<Result<GetCategoryByIdQueryResult>>,  
IBaseRequest
```

## Inheritance

object ← GetCategoryByIdQuery

## Implements

IRequest<[Result<GetCategoryByIdQueryResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Properties

## Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

# Class GetCategoryByIdQueryHandler

Namespace: [Application.Categories.Queries.GetCategoryById](#)

Assembly: Application.dll

```
public class GetCategoryByIdQueryHandler : IRequestHandler<GetCategoryByIdQuery,
Result<GetCategoryByIdQueryResult>>
```

## Inheritance

object ← GetCategoryByIdQueryHandler

## Implements

IRequestHandler<[GetCategoryByIdQuery](#), [Result<GetCategoryByIdQueryResult>](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### GetCategoryByIdQueryHandler(IRepositoryManager, IMapper)

```
public GetCategoryByIdQueryHandler(IRepositoryManager repositoryManager, IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

mapper IMapper

## Methods

### BuildCategoryTree(Guid?, IEnumerable<Category>)

```
public List<GetCategoryByIdQueryResult> BuildCategoryTree(Guid? parentId,
IEnumerable<Category> allCategories)
```

Parameters

parentId [Guid](#)?

allCategories [IEnumerable](#)<[Category](#)>

Returns

[List](#)<[GetCategoryByIdQueryResult](#)>

## Handle(GetCategoryByIdQuery, CancellationToken)

Handles a request

```
public Task<Result<GetCategoryByIdQueryResult>> Handle(GetCategoryByIdQuery request,  
CancellationToken cancellationToken)
```

Parameters

request [GetCategoryByIdQuery](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

Returns

[Task](#)<[Result](#)<[GetCategoryByIdQueryResult](#)>>

Response from the request

# Class GetCategoryByIdQueryResult

Namespace: [Application.Categories.Queries.GetCategoryById](#)

Assembly: Application.dll

```
public class GetCategoryByIdQueryResult
```

## Inheritance

object ← GetCategoryByIdQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### CreationDate

```
public DateTimeOffset CreationDate { get; set; }
```

Property Value

[DateTimeOffset](#)

### Description

```
public string? Description { get; set; }
```

Property Value

[string](#)

### HasChildren

```
public bool HasChildren { get; set; }
```

Property Value

[bool](#) ↗

Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

IsArchived

```
public bool IsArchived { get; set; }
```

Property Value

[bool](#) ↗

IsRoot

```
public bool IsRoot { get; set; }
```

Property Value

[bool](#) ↗

LastArchiveDate

```
public DateTimeOffset? LastArchiveDate { get; set; }
```

Property Value

[DateTimeOffset](#)?

## LastModifiedDate

```
public DateTimeOffset? LastModifiedDate { get; set; }
```

Property Value

[DateTimeOffset](#)?

## Level

```
public int Level { get; set; }
```

Property Value

[int](#)

## Name

```
public string Name { get; set; }
```

Property Value

[string](#)

## ParentId

```
public Guid? ParentId { get; set; }
```

Property Value

[Guid](#)?

VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int](#)

# Namespace Application.Commons.Attributes

## Classes

[MinMaxAttribute](#)

[SwaggerExampleAttribute](#)

# Class MinMaxAttribute

Namespace: [Application.Commons.Attributes](#)

Assembly: Application.dll

```
[AttributeUsage(AttributeTargets.Property, AllowMultiple = false)]
public class MinMaxAttribute : Attribute
```

## Inheritance

object ← [Attribute](#) ← MinMaxAttribute

## Inherited Members

[Attribute.Equals\(object\)](#) , [Attribute.GetCustomAttribute\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttribute\(Assembly, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(Module, Type\)](#) , [Attribute.GetCustomAttribute\(Module, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type, bool\)](#) , [Attribute.GetCustomAttributes\(Assembly\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, bool\)](#) , [Attribute.GetCustomAttributes\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, Type, bool\)](#) , [Attribute.GetCustomAttributes\(MemberInfo\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type, bool\)](#) , [Attribute.GetCustomAttributes\(Module\)](#) ,  
[Attribute.GetCustomAttributes\(Module, bool\)](#) , [Attribute.GetCustomAttributes\(Module, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Module, Type, bool\)](#) , [Attribute.GetCustomAttributes\(ParameterInfo\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type, bool\)](#) , [Attribute.GetHashCode\(\)](#) ,  
[Attribute.IsDefaultAttribute\(\)](#) , [Attribute.IsDefined\(Assembly, Type\)](#) ,  
[Attribute.IsDefined\(Assembly, Type, bool\)](#) , [Attribute.IsDefined\(MemberInfo, Type\)](#) ,  
[Attribute.IsDefined\(MemberInfo, Type, bool\)](#) , [Attribute.IsDefined\(Module, Type\)](#) ,  
[Attribute.IsDefined\(Module, Type, bool\)](#) , [Attribute.IsDefined\(ParameterInfo, Type\)](#) ,  
[Attribute.IsDefined\(ParameterInfo, Type, bool\)](#) , [Attribute.Match\(object\)](#) , [Attribute.TypeId](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### MinMaxAttribute(int, int)

```
public MinMaxAttribute(int min, int max)
```

#### Parameters

min [int ↗](#)

max [int ↗](#)

## Properties

### Max

```
public int Max { get; }
```

#### Property Value

[int ↗](#)

### Min

```
public int Min { get; }
```

#### Property Value

[int ↗](#)

# Class SwaggerExampleAttribute

Namespace: [Application.Commons.Attributes](#)

Assembly: Application.dll

```
[AttributeUsage(AttributeTargets.Property, AllowMultiple = false)]
public class SwaggerExampleAttribute : Attribute
```

## Inheritance

object ← [Attribute](#) ← SwaggerExampleAttribute

## Inherited Members

[Attribute.Equals\(object\)](#) , [Attribute.GetCustomAttribute\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttribute\(Assembly, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(Module, Type\)](#) , [Attribute.GetCustomAttribute\(Module, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type, bool\)](#) , [Attribute.GetCustomAttributes\(Assembly\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, bool\)](#) , [Attribute.GetCustomAttributes\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, Type, bool\)](#) , [Attribute.GetCustomAttributes\(MemberInfo\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type, bool\)](#) , [Attribute.GetCustomAttributes\(Module\)](#) ,  
[Attribute.GetCustomAttributes\(Module, bool\)](#) , [Attribute.GetCustomAttributes\(Module, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Module, Type, bool\)](#) , [Attribute.GetCustomAttributes\(ParameterInfo\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type, bool\)](#) , [Attribute.GetHashCode\(\)](#) ,  
[Attribute.IsDefaultAttribute\(\)](#) , [Attribute.IsDefined\(Assembly, Type\)](#) ,  
[Attribute.IsDefined\(Assembly, Type, bool\)](#) , [Attribute.IsDefined\(MemberInfo, Type\)](#) ,  
[Attribute.IsDefined\(MemberInfo, Type, bool\)](#) , [Attribute.IsDefined\(Module, Type\)](#) ,  
[Attribute.IsDefined\(Module, Type, bool\)](#) , [Attribute.IsDefined\(ParameterInfo, Type\)](#) ,  
[Attribute.IsDefined\(ParameterInfo, Type, bool\)](#) , [Attribute.Match\(object\)](#) , [Attribute.TypeId](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### SwaggerExampleAttribute(object)

```
public SwaggerExampleAttribute(object example)
```

#### Parameters

example object

## Properties

### Example

```
public object Example { get; }
```

#### Property Value

object

# Namespace Application.CommonsBehaviors Classes

[LoggingBehavior<TRequest, TResponse>](#)

[MetricsBehavior<TRequest, TResponse>](#)

[ValidationBehavior<TRequest, TResponse>](#)

# Class LoggingBehavior<TRequest, TResponse>

Namespace: [Application.Commons.Behaviors](#)

Assembly: Application.dll

```
public class LoggingBehavior<TRequest, TResponse> : IPipelineBehavior<TRequest, TResponse>
where TRequest : notnull
```

## Type Parameters

TRequest

TResponse

## Inheritance

object ← LoggingBehavior<TRequest, TResponse>

## Implements

IPipelineBehavior<TRequest, TResponse>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

LoggingBehavior(ILogger<LoggingBehavior<TRequest, TResponse>>)

```
public LoggingBehavior(ILogger<LoggingBehavior<TRequest, TResponse>> logger)
```

## Parameters

logger [ILogger](#)<[LoggingBehavior](#)<TRequest, TResponse>>

## Methods

## Handle(TRequest, RequestHandlerDelegate<TResponse>, CancellationToken)

Pipeline handler. Perform any additional behavior and await the `next` delegate as necessary

```
public Task<TResponse> Handle(TRequest request, RequestHandlerDelegate<TResponse> next,  
CancellationToken cancellationToken)
```

### Parameters

`request` TRequest

Incoming request

`next` RequestHandlerDelegate<TResponse>

Awaitable delegate for the next action in the pipeline. Eventually this delegate represents the handler.

`cancellationToken` [CancellationToken](#)

Cancellation token

### Returns

[Task](#) <TResponse>

Awaitable task returning the `TResponse`

# Class MetricsBehavior<TRequest, TResponse>

Namespace: [Application.Commons.Behaviors](#)

Assembly: Application.dll

```
public class MetricsBehavior<TRequest, TResponse> : IPipelineBehavior<TRequest, TResponse>
where TRequest : notnull
```

## Type Parameters

TRequest

TResponse

## Inheritance

object ← MetricsBehavior<TRequest, TResponse>

## Implements

IPipelineBehavior<TRequest, TResponse>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

MetricsBehavior(ILogger<MetricsBehavior<TRequest, TResponse>>)

```
public MetricsBehavior(ILogger<MetricsBehavior<TRequest, TResponse>> logger)
```

## Parameters

logger [ILogger](#)<MetricsBehavior<TRequest, TResponse>>

## Methods

## Handle(TRequest, RequestHandlerDelegate<TResponse>, CancellationToken)

Pipeline handler. Perform any additional behavior and await the `next` delegate as necessary

```
public Task<TResponse> Handle(TRequest request, RequestHandlerDelegate<TResponse> next,  
CancellationToken cancellationToken)
```

### Parameters

`request` TRequest

Incoming request

`next` RequestHandlerDelegate<TResponse>

Awaitable delegate for the next action in the pipeline. Eventually this delegate represents the handler.

`cancellationToken` [CancellationToken](#)

Cancellation token

### Returns

[Task](#) <TResponse>

Awaitable task returning the `TResponse`

# Class ValidationBehavior<TRequest, TResponse>

Namespace: [Application.Commons.Behaviors](#)

Assembly: Application.dll

```
public class ValidationBehavior<TRequest, TResponse> : IPipelineBehavior<TRequest, TResponse> where TRequest : notnull where TResponse : class
```

## Type Parameters

TRequest

TResponse

## Inheritance

object ← ValidationBehavior<TRequest, TResponse>

## Implements

IPipelineBehavior<TRequest, TResponse>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

ValidationBehavior(IEnumerable<IValidator<TRequest>>)

```
public ValidationBehavior(IEnumerable<IValidator<TRequest>> validators)
```

## Parameters

validators [IEnumerable](#)<IValidator<TRequest>>

## Methods

## Handle(TRequest, RequestHandlerDelegate<TResponse>, CancellationToken)

Pipeline handler. Perform any additional behavior and await the `next` delegate as necessary

```
public Task<TResponse> Handle(TRequest request, RequestHandlerDelegate<TResponse> next,  
CancellationToken cancellationToken)
```

### Parameters

`request` TRequest

Incoming request

`next` RequestHandlerDelegate<TResponse>

Awaitable delegate for the next action in the pipeline. Eventually this delegate represents the handler.

`cancellationToken` [CancellationToken](#)

Cancellation token

### Returns

[Task](#) <TResponse>

Awaitable task returning the `TResponse`

# Namespace Application.Commons.Extensions

## Classes

[CloneExtension](#)

[LoggingLevelController](#)

[StringExtension](#)

# Class CloneExtension

Namespace: [Application.Commons.Extensions](#)

Assembly: Application.dll

```
public static class CloneExtension
```

## Inheritance

object ← CloneExtension

## Methods

### DeepClone<T>(T)

```
public static T DeepClone<T>(T obj)
```

#### Parameters

**obj** T

#### Returns

T

#### Type Parameters

T

# Class LoggingLevelController

Namespace: [Application.Commons.Extensions](#)

Assembly: Application.dll

```
public static class LoggingLevelController
```

## Inheritance

object ← LoggingLevelController

## Fields

### LevelSwitch

```
public static readonly LoggingLevelSwitch LevelSwitch
```

Field Value

LoggingLevelSwitch

# Class StringExtension

Namespace: [Application.Commons.Extensions](#)

Assembly: Application.dll

```
public static class StringExtension
```

## Inheritance

object ← StringExtension

## Methods

### FirstCharToLower(string)

```
public static string FirstCharToLower(this string input)
```

#### Parameters

**input** [string](#) ↗

#### Returns

[string](#) ↗

### ToReadableSize(double)

```
public static string ToReadableSize(double bytes)
```

#### Parameters

**bytes** [double](#) ↗

#### Returns

[string](#) ↗

# Namespace Application.Commons.Extentions Classes

[QuestionDifficultyExtenstion](#)

# Class QuestionDifficultyExtenstion

Namespace: [Application.Commons.Extentions](#)

Assembly: Domain.dll

```
public static class QuestionDifficultyExtenstion
```

## Inheritance

object ← QuestionDifficultyExtenstion

## Methods

### GetDifficultyCategory(short)

```
public static QuestionDifficulty GetDifficultyCategory(this short mattrix)
```

#### Parameters

mattrix [short](#)

#### Returns

[QuestionDifficulty](#)

float

### GetMattrix(QuestionDifficulty)

calcalute the startign mattrix for each given difficulty,

```
public static short GetMattrix(this QuestionDifficulty difficulty)
```

#### Parameters

difficulty [QuestionDifficulty](#)

## Returns

[short ↗](#)

float

# Namespace Application.Commons.Managers

## Interfaces

[IAuditManager](#)

[IServiceManager](#)

[ISystemManager](#)

# Interface IAuditManager

Namespace: [Application.Commons.Managers](#)

Assembly: Application.dll

```
public interface IAuditManager
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### ApplicationLogRepository

```
IApplicationLogRepository ApplicationLogRepository { get; }
```

Property Value

[IApplicationLogRepository](#)

### AuditTrailService

```
IAuditTrailService AuditTrailService { get; }
```

Property Value

[IAuditTrailService](#)

# Interface IServiceManager

Namespace: [Application.Commons.Managers](#)

Assembly: Application.dll

```
public interface IServiceManager : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### CategoryService

```
ICategoryService CategoryService { get; }
```

#### Property Value

[ICategoryService](#)

### LookupService

```
ILookupService LookupService { get; }
```

#### Property Value

[ILookupService](#)

### QuestionService

```
IQuestionService QuestionService { get; }
```

Property Value

[IQuestionService](#)

## SourceService

ISourceService SourceService { **get**; }

Property Value

[ISourceService](#)

## TagService

ITagService TagService { **get**; }

Property Value

[ITagService](#)

# Interface ISystemManager

Namespace: [Application.Commons.Managers](#)

Assembly: Application.dll

```
public interface ISystemManager
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### SystemService

```
ISystemService SystemService { get; }
```

#### Property Value

[ISystemService](#)

# Namespace Application.Commons.Mapping Config

## Classes

[CategoryMapper](#)

[SourceMapper](#)

[TagMapper](#)

# Class CategoryMapper

Namespace: [Application.Commons.MappingConfig](#)

Assembly: Application.dll

```
public class CategoryMapper : IRegister
```

## Inheritance

object ← CategoryMapper

## Implements

IRegister

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### Register(TypeAdapterConfig)

```
public void Register(TypeAdapterConfig config)
```

## Parameters

**config** TypeAdapterConfig

# Class SourceMapper

Namespace: [Application.Commons.MappingConfig](#)

Assembly: Application.dll

```
public class SourceMapper : IRegister
```

## Inheritance

object ← SourceMapper

## Implements

IRegister

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### Register(TypeAdapterConfig)

```
public void Register(TypeAdapterConfig config)
```

## Parameters

**config** TypeAdapterConfig

# Class TagMapper

Namespace: [Application.Commons.MappingConfig](#)

Assembly: Application.dll

```
public class TagMapper : IRegister
```

**Inheritance**

object ← TagMapper

**Implements**

IRegister

**Extension Methods**

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### Register(TypeAdapterConfig)

```
public void Register(TypeAdapterConfig config)
```

Parameters

**config** TypeAdapterConfig

# Namespace Application.Commons.Models.Commands

## Classes

[AddMetadataCommand](#)

# Class AddMetadataCommand

Namespace: [Application.Commons.Models.Commands](#)

Assembly: Application.dll

```
public class AddMetadataCommand
```

## Inheritance

object ← AddMetadataCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### FiledName

```
public string FiledName { get; set; }
```

Property Value

[string](#) ↗

### FiledType

```
public FiledType FiledType { get; set; }
```

Property Value

[FiledType](#)

### Value

```
public string Value { get; set; }
```

Property Value

[string](#) ↗

# Namespace Application.Commons.Models. Pageination

## Classes

[PageMetaData](#)

[PageRequest](#)

[PageResponse<T>](#)

# Class PageMetaData

Namespace: [Application.Commons.Models.Paginatio](#)n

Assembly: Application.dll

```
public class PageMetaData
```

## Inheritance

object ← PageMetaData

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### FirstItemIndex

```
[SwaggerExample(1)]  
public int FirstItemIndex { get; }
```

#### Property Value

[int](#)

### HasNextPage

```
[SwaggerSchema("has the next page?")]  
[SwaggerExample(true)]  
public bool HasNextPage { get; }
```

#### Property Value

[bool](#)

## HasPreviousPage

```
[SwaggerSchema("has the previous page?")]
[SwaggerExample(true)]
public bool HasPreviousPage { get; }
```

Property Value

[bool](#) ↗

## IsLastPage

```
[SwaggerSchema("Is this the last page")]
[SwaggerExample(false)]
public bool IsLastPage { get; }
```

Property Value

[bool](#) ↗

## LastItemIndex

```
[SwaggerExample(99)]
public int LastItemIndex { get; }
```

Property Value

[int](#) ↗

## NextPage

```
[SwaggerSchema("The number of next page")]
[SwaggerExample(4)]
public int? NextPage { get; }
```

Property Value

[int](#)?

## PageNumber

```
[SwaggerSchema("The current page number")]
[SwaggerExample(3)]
public int PageNumber { get; set; }
```

Property Value

[int](#)

## PageSize

```
[SwaggerSchema("The size of current page")]
[SwaggerExample(50)]
public int PageSize { get; set; }
```

Property Value

[int](#)

## PreviousPage

```
[SwaggerSchema("The number of previous page")]
[SwaggerExample(2)]
public int? PreviousPage { get; }
```

Property Value

[int](#)?

## TotalPages

```
[SwaggerSchema("The total number of pages")]
[SwaggerExample(100)]
public int TotalPages { get; set; }
```

Property Value

[int ↗](#)

## TotalRecords

```
[SwaggerSchema("The total number of record")]
[SwaggerExample(500)]
public int TotalRecords { get; set; }
```

Property Value

[int ↗](#)

# Class PageRequest

Namespace: [Application.Commons.Models.Paginatio](#)n

Assembly: Application.dll

```
public class PageRequest
```

## Inheritance

object ← PageRequest

## Derived

[LogSearchQuery](#), [GetDeletedEntitiesQuery](#), [GetAllCategoryQuery](#), [GetEntityTimelineQuery](#),  
[GetQuestionHistoryQuery](#), [GetAllTagsQuery](#), [GetAllQuestionsQuery](#), [GetAllSourceQuery](#),  
[GetRelatedSourcesQuery](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### PageNumber

```
public int PageNumber { get; set; }
```

#### Property Value

[int](#)

### PageSize

```
public int PageSize { get; set; }
```

#### Property Value

[int](#)

# Class PageResponse<T>

Namespace: [Application.Commons.Models.Paginatio](#)n

Assembly: Application.dll

```
[SwaggerSchema("")]  
public class PageResponse<T> where T : class
```

## Type Parameters

T

## Inheritance

object ← PageResponse<T>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### PageResponse(IEnumerable<T>, int, int, int)

```
public PageResponse(IEnumerable<T> data, int pageNumber, int pageSize, int totalRecords)
```

## Parameters

data [IEnumerable](#)<T>

pageNumber [int](#)

pageSize [int](#)

totalRecords [int](#)

### PageResponse(int, int, int)

```
public PageResponse(int pageNumber, int pageSize, int totalRecords)
```

## Parameters

pageNumber [int](#)

pageSize [int](#)

totalRecords [int](#)

## Properties

### Data

[SwaggerSchema(null)]

```
public IEnumerable<T> Data { get; set; }
```

### Property Value

[IEnumerable](#)<T>

### MetaData

[SwaggerSchema("a meta data for pagination object")]

```
public PageMetaData MetaData { get; set; }
```

### Property Value

[PageMetaData](#)

# Namespace Application.Commons.Models. Results

## Classes

[PartialsSuccessResult](#)

[Result](#)

[Result<T>](#)

# Class PartialsSuccessResult

Namespace: [Application.Commons.Models.Results](#)

Assembly: Application.dll

```
public class PartialsSuccessResult
```

## Inheritance

object ← PartialsSuccessResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Errors

```
public Dictionary<string, string[]> Errors { get; set; }
```

### Property Value

[Dictionary](#) <[string](#), [string](#)[]>

### Successed

```
public List<Guid> Successed { get; set; }
```

### Property Value

[List](#) <[Guid](#)>

# Class Result

Namespace: [Application.Commons.Models.Results](#)

Assembly: Application.dll

```
public class Result
```

## Inheritance

object ← Result

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Errors

```
public Dictionary<string, string[]> Errors { get; init; }
```

## Property Value

[Dictionary](#) <[string](#), [string](#)[]>

## IsSuccess

```
public bool IsSuccess { get; init; }
```

## Property Value

[bool](#)

## StatusCode

```
public HttpStatusCode StatusCode { get; set; }
```

Property Value

[HttpStatusCode](#) ↗

## Methods

AddError(string, string)

```
public Result AddError(string key, string error)
```

Parameters

key [string](#) ↗

error [string](#) ↗

Returns

[Result](#)

ConflictFailure(string, string)

```
public static Result ConflictFailure(string key, string error)
```

Parameters

key [string](#) ↗

error [string](#) ↗

Returns

[Result](#)

## Failure(Dictionary<string, string[]>)

```
public static Result Failure(Dictionary<string, string[]> errors)
```

Parameters

errors [Dictionary<string, string\[\]>](#)

Returns

[Result](#)

## Failure(Dictionary<string, string[]>, HttpStatusCode)

```
public static Result Failure(Dictionary<string, string[]> errors, HttpStatusCode statusCode)
```

Parameters

errors [Dictionary<string, string\[\]>](#)

statusCode [HttpStatusCode](#)

Returns

[Result](#)

## Failure(string, IEnumerable<string>)

```
public static Result Failure(string key, IEnumerable<string> errors)
```

Parameters

key [string](#)

errors [IEnumerable<string>](#)

Returns

## [Result](#)

### Failure(string, IEnumerable<string>, HttpStatusCode)

```
public static Result Failure(string key, IEnumerable<string> errors,  
HttpStatusCode statusCode)
```

#### Parameters

key [string](#)

errors [IEnumerable<string>](#)

statusCode [HttpStatusCode](#)

#### Returns

## [Result](#)

### Failure(string, string)

```
public static Result Failure(string key, string error)
```

#### Parameters

key [string](#)

error [string](#)

#### Returns

## [Result](#)

### Failure(string, string, HttpStatusCode)

```
public static Result Failure(string key, string error, HttpStatusCode statusCode)
```

Parameters

`key` [string](#)

`error` [string](#)

`statusCode` [HttpStatusCode](#)

Returns

[Result](#)

## NoContentSuccess()

```
public static Result NoContentSuccess()
```

Returns

[Result](#)

## NotFoundFailure(string, string)

```
public static Result NotFoundFailure(string key, string error)
```

Parameters

`key` [string](#)

`error` [string](#)

Returns

[Result](#)

## Success()

```
public static Result Success()
```

Returns

[Result](#)

## Success(HttpStatusCode)

```
public static Result Success(HttpStatusCode statusCode)
```

Parameters

statusCode [HttpStatusCode](#)

Returns

[Result](#)

## UnprocessableEntityFailure(string, IEnumerable<string>)

```
public static Result UnprocessableEntityFailure(string key, IEnumerable<string> errors)
```

Parameters

key [string](#)

errors [IEnumerable](#)<[string](#)>

Returns

[Result](#)

## UnprocessableEntityFailure(string, string)

```
public static Result UnprocessableEntityFailure(string key, string error)
```

## Parameters

**key** [string](#) ↗

**error** [string](#) ↗

## Returns

[Result](#)

# Class Result<T>

Namespace: [Application.Commons.Models.Results](#)

Assembly: Application.dll

```
public class Result<T>
```

## Type Parameters

T

## Inheritance

object ← Result<T>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Errors

```
public Dictionary<string, string[]> Errors { get; init; }
```

## Property Value

[Dictionary<string\[\], string\[\]\[\]>](#)

## IsSuccess

```
public bool IsSuccess { get; init; }
```

## Property Value

[bool](#)

# StatusCode

```
public HttpStatusCode StatusCode { get; set; }
```

Property Value

[HttpStatusCode](#) ↗

Value

```
public T? Value { get; init; }
```

Property Value

T

## Methods

AddError(string, string)

```
public Result<T> AddError(string key, string error)
```

Parameters

**key** [string](#) ↗

**error** [string](#) ↗

Returns

[Result](#)<T>

ConflictFailure(string, string)

```
public static Result<T> ConflictFailure(string key, string error)
```

Parameters

**key** [string](#)

**error** [string](#)

Returns

[Result](#)<T>

## CreatedSuccess(T)

```
public static Result<T> CreatedSuccess(T value)
```

Parameters

**value** T

Returns

[Result](#)<T>

## Failure(Dictionary<string, string[]>)

```
public static Result<T> Failure(Dictionary<string, string[]> errors)
```

Parameters

**errors** [Dictionary](#)<[string](#), [string](#)[]>

Returns

[Result](#)<T>

## Failure(Dictionary<string, string[]>, HttpStatusCode)

```
public static Result<T> Failure(Dictionary<string, string[]> errors,  
HttpStatusCode statusCode)
```

## Parameters

errors [Dictionary<string, string\[\]>](#)

statusCode [HttpStatusCode](#)

## Returns

[Result<T>](#)

## Failure(string, IEnumerable<string>)

```
public static Result<T> Failure(string key, IEnumerable<string> errors)
```

## Parameters

key [string](#)

errors [IEnumerable<string>](#)

## Returns

[Result<T>](#)

## Failure(string, IEnumerable<string>, HttpStatusCode)

```
public static Result<T> Failure(string key, IEnumerable<string> errors,  
HttpStatusCode statusCode)
```

## Parameters

key [string](#)

errors [IEnumerable<string>](#)

`statusCode` [HttpStatusCodes](#)

Returns

[Result](#)<T>

## Failure(string, string)

```
public static Result<T> Failure(string key, string error)
```

Parameters

`key` [string](#)

`error` [string](#)

Returns

[Result](#)<T>

## Failure(string, string, HttpStatusCode)

```
public static Result<T> Failure(string key, string error, HttpStatusCode statusCode)
```

Parameters

`key` [string](#)

`error` [string](#)

`statusCode` [HttpStatusCodes](#)

Returns

[Result](#)<T>

## NotFoundFailure(string, string)

```
public static Result<T> NotFoundFailure(string key, string error)
```

Parameters

key [string](#)

error [string](#)

Returns

[Result](#)<T>

## Success(T)

```
public static Result<T> Success(T value)
```

Parameters

value T

Returns

[Result](#)<T>

## Success(T, HttpStatusCode)

```
public static Result<T> Success(T value, HttpStatusCode statusCode)
```

Parameters

value T

statusCode [HttpStatusCode](#)

Returns

[Result](#)<T>

## UnprocessableEntityFailure(Dictionary<string, string[]>)

```
public static Result<T> UnprocessableEntityFailure(Dictionary<string, string[]> errors)
```

### Parameters

errors [Dictionary<string, string\[\]>](#)

### Returns

[Result<T>](#)

## UnprocessableEntityFailure(string, IEnumerable<string>)

```
public static Result<T> UnprocessableEntityFailure(string key, IEnumerable<string> errors)
```

### Parameters

key [string](#)

errors [IEnumerable<string>](#)

### Returns

[Result<T>](#)

## UnprocessableEntityFailure(string, string)

```
public static Result<T> UnprocessableEntityFailure(string key, string error)
```

### Parameters

key [string](#)

error [string](#)

Returns

[Result](#)<T>

# Namespace Application.Commons.Models.ServicesModel.Source

## Classes

[AddMetadataSourceModel](#)

[AddSourceReferenceServiceModel](#)

[AddSourceServiceModel](#)

# Class AddMetadataSourceModel

Namespace: [Application.Commons.Models.ServicesModel.Source](#)

Assembly: Application.dll

```
public class AddMetadataSourceModel
```

## Inheritance

object ← AddMetadataSourceModel

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### FiledName

```
public string FiledName { get; set; }
```

#### Property Value

[string](#) ↗

### FiledType

```
public FiledType FiledType { get; set; }
```

#### Property Value

[FiledType](#)

### Value

```
public string Value { get; set; }
```

Property Value

[string](#) ↗

# Class AddSourceReferenceServiceModel

Namespace: [Application.Commons.Models.ServicesModel.Source](#)

Assembly: Application.dll

```
public class AddSourceReferenceServiceModel
```

## Inheritance

object ← AddSourceReferenceServiceModel

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Metadata

```
public List<AddMetadataSourceModel>? Metadata { get; set; }
```

### Property Value

[List](#) <[AddMetadataSourceModel](#)>

### Note

```
public string Note { get; set; }
```

### Property Value

[string](#)

# Class AddSourceServiceModel

Namespace: [Application.Commons.Models.ServicesModel.Source](#)

Assembly: Application.dll

```
public class AddSourceServiceModel
```

## Inheritance

object ← AddSourceServiceModel

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### CategoryId

```
public Guid? CategoryId { get; set; }
```

Property Value

[Guid](#)?

### Description

```
public string? Description { get; set; }
```

Property Value

[string](#)?

### FileExtension

```
public string? FileExtension { get; set; }
```

Property Value

[string](#)

## FilePath

```
public string? FilePath { get; set; }
```

Property Value

[string](#)

## HasAttachment

```
public bool HasAttachment { get; set; }
```

Property Value

[bool](#)

## Metadata

```
public List<AddMetadataSourceModel>? Metadata { get; set; }
```

Property Value

[List](#)<[AddMetadataSourceModel](#)>

## Tags

```
public List<Guid>? Tags { get; set; }
```

Property Value

[List](#) <[Guid](#)>

Title

```
public string Title { get; set; }
```

Property Value

[string](#)

Type

```
public SourceType Type { get; set; }
```

Property Value

[SourceType](#)

# Namespace Application.Commons.Models. SystemModels

## Classes

[AppStatusResponseModel](#)

[CollectionStorageStat](#)

# Class AppStatusResponseModel

Namespace: [Application.Commons.Models.SystemModels](#)

Assembly: Application.dll

```
public class AppStatusResponseModel
```

## Inheritance

object ← AppStatusResponseModel

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### AppVersion

```
public string AppVersion { get; set; }
```

#### Property Value

[string](#)

## Collections

```
public List<CollectionStorageStat> Collections { get; set; }
```

#### Property Value

[List](#) <[CollectionStorageStat](#)>

## CpuArchitecture

```
public string CpuArchitecture { get; set; }
```

Property Value

[string](#)

## DatabaseSizeBytes

```
public string DatabaseSizeBytes { get; set; }
```

Property Value

[string](#)

## MemoryUsageMB

```
public string MemoryUsageMB { get; set; }
```

Property Value

[string](#)

## OSVersion

```
public string OSVersion { get; set; }
```

Property Value

[string](#)

## Uptime

```
public TimeSpan Uptime { get; set; }
```

Property Value

[TimeSpan](#) ↗

# Class CollectionStorageStat

Namespace: [Application.Commons.Models.SystemModels](#)

Assembly: Application.dll

```
public class CollectionStorageStat
```

## Inheritance

object ← CollectionStorageStat

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### AverageDocumentSizeBytes

```
public string AverageDocumentSizeBytes { get; set; }
```

Property Value

[string](#) ↗

### DocumentCount

```
public int DocumentCount { get; set; }
```

Property Value

[int](#) ↗

### EstimatedTotalSizeBytes

```
public string EstimatedTotalSizeBytes { get; set; }
```

Property Value

[string](#)

IndexCount

```
public int IndexCount { get; set; }
```

Property Value

[int](#)

Name

```
public string Name { get; set; }
```

Property Value

[string](#)

# Namespace Application.Commons.Services Interfaces

[IAuditTrailService](#)

[ICategoryService](#)

[ILoggingService](#)

[ILookupService](#)

[IQuestionService](#)

[ISourceService](#)

[ISystemService](#)

[ITagService](#)

# Interface IAuditTrailService

Namespace: [Application.Commons.Services](#)

Assembly: Application.dll

```
public interface IAuditTrailService
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#), [ObjectExtention.IsNull\(object\)](#).

## Methods

AddNewEntity<T>(EntityName, Guid, ActionBy, T, int, string?)

```
void AddNewEntity<T>(EntityName entityName, Guid entityId, ActionBy actionBy, T entity, int  
versionNumber, string? comment = null)
```

### Parameters

entityName [EntityName](#)

entityId [Guid](#)

actionBy [ActionBy](#)

entity T

versionNumber [int](#)

comment [string](#)

### Type Parameters

T

Count(Guid)

```
int Count(Guid entityId)
```

Parameters

entityId [Guid](#)

Returns

[int](#)

## CountDeletedEntity()

```
int CountDeletedEntity()
```

Returns

[int](#)

## DeleteEntity<T>(EntityName, Guid, ActionType, ActionBy, T, int, string?)

```
void DeleteEntity<T>(EntityName entityName, Guid entityId, ActionType actionType, ActionBy  
actionBy, T entity, int versionNumber, string? comment = null)
```

Parameters

entityName [EntityName](#)

entityId [Guid](#)

actionType [ActionType](#)

actionBy [ActionBy](#)

entity T

versionNumber [int](#)

comment [string](#)

## Type Parameters

T

## GetDeletedEntities(Func<AuditTrail, bool>, int, int)

IEnumerable<AuditTrail> **GetDeletedEntities**(Func<AuditTrail, [bool](#)> func, [int](#) pageNumber, [int](#) pageSize)

## Parameters

func [Func](#)<[AuditTrail](#), [bool](#)>

pageNumber [int](#)

pageSize [int](#)

## Returns

[IEnumerable](#)<[AuditTrail](#)>

## GetEntityTrail(Func<AuditTrail, bool>, int, int, Guid)

IEnumerable<AuditTrail> **GetEntityTrail**(Func<AuditTrail, [bool](#)> func, [int](#) pageNumber, [int](#) pageSize, Guid entityId)

## Parameters

func [Func](#)<[AuditTrail](#), [bool](#)>

pageNumber [int](#)

pageSize [int](#)

entityId [Guid](#)

## Returns

[IEnumerable](#) <[AuditTrail](#)>

## GetEntityTrailDetails(Guid, EntityName, Guid)

AuditTrail [GetEntityTrailDetails](#)(Guid trailId, EntityName entityName, Guid entityId)

### Parameters

trailId [Guid](#)

entityName [EntityName](#)

entityId [Guid](#)

### Returns

[AuditTrail](#)

## UpdateEntitiesBulk<T>(EntityName, IEnumerable<(Guid EntityId, ActionType ActionType, ActionBy ActionBy, T OldEntity, T NewEntity, int Version, string? Comment)>)

void [UpdateEntitiesBulk](#)<T>(EntityName entityName, IEnumerable<(Guid EntityId, ActionType ActionType, ActionBy ActionBy, T OldEntity, T NewEntity, int Version, string? Comment)> changes)

### Parameters

entityName [EntityName](#)

changes [IEnumerable](#) <(Guid EntityId, ActionType ActionType, ActionBy ActionBy, T OldEntity, T NewEntity, int Version, string? Comment)>

### Type Parameters

T

# UpdateEntity<T>(EntityName, Guid, ActionType, ActionBy, T, T, int, string?)

```
void UpdateEntity<T>(EntityName entityName, Guid entityId, ActionType actionType, ActionBy  
actionBy, T oldEntity, T newEntity, int versionNumber, string? comment = null)
```

## Parameters

entityName [EntityName](#)

entityId [Guid](#)

actionType [ActionType](#)

actionBy [ActionBy](#)

oldEntity T

newEntity T

versionNumber [int](#)

comment [string](#)

## Type Parameters

T

# Interface ICategoryService

Namespace: [Application.Commons.Services](#)

Assembly: Application.dll

```
public interface ICategoryService
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### AddCategory(string, string, Guid?)

```
Result<Category> AddCategory(string name, string description, Guid? parentId)
```

#### Parameters

name [string](#)

description [string](#)

parentId [Guid](#)?

#### Returns

[Result<Category>](#)

### UpdateCategory(Guid, string, string)

```
Result UpdateCategory(Guid categoryId, string name, string description)
```

#### Parameters

categoryId [Guid](#)

`name` [string](#)

`description` [string](#)

Returns

[Result](#)

# Interface ILoggingService

Namespace: [Application.Commons.Services](#)

Assembly: Application.dll

```
public interface ILoggingService
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Methods

### GetAllLogs()

```
IEnumerable<ApplicationLog> GetAllLogs()
```

Returns

[IEnumerable](#) <[ApplicationLog](#)>

# Interface ILookupService

Namespace: [Application.Commons.Services](#)

Assembly: Application.dll

```
public interface ILookupService : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### AddLanguage(Language)

```
void AddLanguage(Language language)
```

#### Parameters

language [Language](#)

### AddSource(Source)

```
void AddSource(Source source)
```

#### Parameters

source [Source](#)

### GetLanguageReference(string)

```
Guid GetLanguageReference(string code)
```

Parameters

code [string](#)

Returns

[Guid](#)

## GetLanguages()

```
IEnumerable<Language> GetLanguages()
```

Returns

[IEnumerable](#) <[Language](#)>

## GetSource(Guid)

```
Source GetSource(Guid id)
```

Parameters

id [Guid](#)

Returns

[Source](#)

## UpdateSource(Source)

```
void UpdateSource(Source source)
```

## Parameters

source [Source](#)

# Interface IQuestionService

Namespace: [Application.Commons.Services](#)

Assembly: Application.dll

```
public interface IQuestionService : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### Count()

```
int Count()
```

Returns

[int](#)

### DeleteAllQuestions()

```
void DeleteAllQuestions()
```

### DeleteQuestion(Guid)

```
void DeleteQuestion(Guid id)
```

Parameters

[id Guid](#)

## GetQuestion(Guid)

Question **GetQuestion**(Guid id)

Parameters

**id** [Guid](#)

Returns

[Question](#)

## GetQuestions(int, int)

IEnumerable<Question> **GetQuestions**(int pageNumber, int pageSize)

Parameters

**pageNumber** [int](#)

**pageSize** [int](#)

Returns

[IEnumerable](#) <[Question](#)>

## InsertQuestion(Question)

Question **InsertQuestion**(Question question)

Parameters

**question** [Question](#)

Returns

[Question](#)

## UpdateQuestion(Question)

```
void UpdateQuestion(Question question)
```

Parameters

question [Question](#)

# Interface ISourceService

Namespace: [Application.Commons.Services](#)

Assembly: Application.dll

```
public interface ISourceService
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#), [ObjectExtention.IsNull\(object\)](#).

## Methods

### AddBulkTag(Guid, IEnumerable<Guid>)

```
Result<PartialsSuccessResult> AddBulkTag(Guid sourceId, IEnumerable<Guid> tagIds)
```

#### Parameters

sourceId [Guid](#)

tagIds [IEnumerable](#)<[Guid](#)>

#### Returns

[Result](#)<[PartialsSuccessResult](#)>

### AddReference(IEnumerable<AddSourceReferenceServiceModel>, Guid)

```
Result<IEnumerable<SourceReference>>
AddReference(IEnumerable<AddSourceReferenceServiceModel> sourceReferences, Guid sourceId)
```

#### Parameters

sourceReferences [IEnumerable](#)<[AddSourceReferenceServiceModel](#)>

`sourceId` [Guid](#)

Returns

[Result](#)<IEnumerable<[SourceReference](#)>>

## AddSource(AddSourceServiceModel)

`Result<Source> AddSource(AddSourceServiceModel model)`

Parameters

`model` [AddSourceServiceModel](#)

Returns

[Result](#)<[Source](#)>

## AddTag(Guid, Guid)

`Result AddTag(Guid sourceId, Guid tagId)`

Parameters

`sourceId` [Guid](#)

`tagId` [Guid](#)

Returns

[Result](#)

## RemoveBulkTags(Guid, IEnumerable<Guid>)

`Result<PartialsSuccessResult> RemoveBulkTags(Guid sourceId, IEnumerable<Guid> tagIds)`

Parameters

`sourceId` [Guid](#)

`tagIds` [IEnumerable](#)<[Guid](#)>

Returns

[Result](#)<[PartialsSuccessResult](#)>

## RemoveTag(Guid, Guid)

`Result RemoveTag(Guid sourceId, Guid tagId)`

Parameters

`sourceId` [Guid](#)

`tagId` [Guid](#)

Returns

[Result](#)

# Interface ISystemService

Namespace: [Application.Commons.Services](#)

Assembly: Application.dll

```
public interface ISystemService
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Methods

### AnalyzeCollections(int)

```
List<CollectionStorageStat> AnalyzeCollections(int sampleSize = 10)
```

#### Parameters

sampleSize [int](#)

#### Returns

[List](#) <[CollectionStorageStat](#)>

### GetDatabaseFileSizeBytes()

```
long GetDatabaseFileSizeBytes()
```

#### Returns

[long](#)

# Interface ITagService

Namespace: [Application.Commons.Services](#)

Assembly: Application.dll

```
public interface ITagService
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Methods

### AddTag(string, string?, string)

```
Result<Tag> AddTag(string name, string? backgroundColorCode = null, string textColorCode = null)
```

#### Parameters

name [string](#)

backgroundColorCode [string](#)

textColorCode [string](#)

#### Returns

[Result<Tag>](#)

### ArchiveAllTag()

```
Result ArchiveAllTag()
```

#### Returns

[Result](#)

## ArchiveTag(Guid)

Result **ArchiveTag**(Guid id)

### Parameters

**id** [Guid](#)

### Returns

[Result](#)

## DeleteTag(Guid)

Result **DeleteTag**(Guid id)

### Parameters

**id** [Guid](#)

### Returns

[Result](#)

## GetAllTags()

IEnumerable<Tag> **GetAllTags**()

### Returns

[IEnumerable](#)<[Tag](#)>

## UnArchiveTag(Guid)

Result **UnArchiveTag**(Guid id)

Parameters

**id** [Guid](#)

Returns

[Result](#)

## UpdateTag(Guid, string, string, string)

Result **UpdateTag**(Guid id, **string** name, **string** backgroundColorCode, **string** textColorCode)

Parameters

**id** [Guid](#)

**name** [string](#)

**backgroundColorCode** [string](#)

**textColorCode** [string](#)

Returns

[Result](#)

# Namespace Application.Commons.Shared ModelError

## Classes

[LanguageResult](#)

[QuestionSourceResult](#)

[TagResult](#)

# Class LanguageResult

Namespace: [Application.Commons.SharedModelResult](#)

Assembly: Application.dll

```
public class LanguageResult
```

## Inheritance

object ← LanguageResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Code

```
public string Code { get; set; }
```

#### Property Value

[string](#) ↗

### Name

```
public string Name { get; set; }
```

#### Property Value

[string](#) ↗

# Class QuestionSourceResult

Namespace: [Application.Commons.SharedModelResult](#)

Assembly: Application.dll

```
public class QuestionSourceResult
```

## Inheritance

object ← QuestionSourceResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Description

```
public string? Description { get; set; }
```

#### Property Value

[string](#) ↗

### Id

```
public Guid Id { get; set; }
```

#### Property Value

[Guid](#) ↗

### Title

```
public string Title { get; set; }
```

Property Value

[string](#)

Type

```
public SourceTypeLookup Type { get; set; }
```

Property Value

[SourceTypeLookup](#)

URL

```
public string URL { get; set; }
```

Property Value

[string](#)

# Class TagResult

Namespace: [Application.Commons.SharedModelResult](#)

Assembly: Application.dll

```
public class TagResult
```

## Inheritance

object ← TagResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### ColorCode

```
[SwaggerSchema(null, Description = "An collection of tags", Nullable = false, Title =  
"Tag color")]  
[SwaggerExample("#8910A3")]  
[StringLength(7, MinimumLength = 7)]  
public string ColorCode { get; set; }
```

### Property Value

[string](#) ↗

### Id

```
[SwaggerSchema(null, Description = "The Tag id", Nullable = false, Title = "Tag Id")]  
[SwaggerExample("0197c089-8943-736d-b211-b7f4b6bf9728")]  
[StringLength(100, MinimumLength = 1)]  
public Guid Id { get; set; }
```

### Property Value

[Guid](#)

## Name

```
[SwaggerSchema(null, Description = "An collection of tags", Nullable = false, Title =
"Tag name")]
[SwaggerExample("MATH-101")]
[StringLength(100, MinimumLength = 1)]
public string Name { get; set; }
```

## Property Value

[string](#)

# Namespace Application.Commons.Shared ModelError.Source

## Classes

[MetadataResult](#)

[SourceResult](#)

# Class MetadataResult

Namespace: [Application.Commons.SharedModelResult.Source](#)

Assembly: Application.dll

```
public class MetadataResult
```

## Inheritance

object ← MetadataResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### FiledName

```
public string FiledName { get; set; }
```

Property Value

[string](#) ↗

### FileType

```
public FileType FileType { get; set; }
```

Property Value

[FileType](#)

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

Value

```
public string Value { get; set; }
```

Property Value

[string](#) ↗

# Class SourceResult

Namespace: [Application.Commons.SharedModelResult.Source](#)

Assembly: Application.dll

```
public class SourceResult
```

## Inheritance

object ← SourceResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Description

```
[SwaggerSchema(null, Description = "short description for a source", Nullable = true, Title = "Source description")]
[SwaggerExample("Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim. Aliquam lorem ante, dapibus in, viverra quis, feugiat a, tellus. Phasellus viverra nulla ut metus varius laoreet. Quisque rutrum. Aenean imperdiet. Etiam ultricies nisi vel augue. Curabitur ullamcorper ultricies nisi. Nam eget dui. Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet adipiscing sem neque sed ipsum. Nam quam nunc, blandit vel, luctus pulvinar, hendrerit id, lorem. Maecenas nec odio et ante tincidunt tempus. Donec vitae sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt. Duis leo. Sed fringilla mauris sit amet nibh. Donec sodales sagittis magna. Sed consequat, leo eget bibendum sodales, augue velit cursus nunc")]
[StringLength(2500, MinimumLength = 1)]
public string? Description { get; set; }
```

### Property Value

[string](#)

## Id

```
[SwaggerSchema(null, Description = "An UUID unique id for source", ReadOnly = true, Format = "UUID", Nullable = false, Title = "Source Unique Identifier")]
[SwaggerExample("01979c44-b446-7380-ad8b-df6b2a2be0bc")]
public Guid Id { get; set; }
```

## Property Value

[Guid](#)

## Title

```
[SwaggerSchema(null, Description = "A source Title", Nullable = false, Title = "Source title")]
[SwaggerExample("Intro programming")]
[StringLength(100, MinimumLength = 1)]
public string Title { get; set; }
```

## Property Value

[string](#)

## Type

```
[SwaggerSchema(null, Description = "An type of a source", ReadOnly = true, Format = "UUID", Nullable = false, Title = "Source Unique Identifier")]
[SwaggerExample(SourceType.Article)]
[StringLength(50, MinimumLength = 1)]
public SourceType Type { get; set; }
```

## Property Value

[SourceType](#)

# Namespace Application.Commons.Shared ModelResult.Timeline.EntityTimeline

## Classes

[TimelineQueryResult](#)

# Class TimelineQueryResult

Namespace: [Application.Commons.SharedModelResult.Timeline.EntityTimeline](#)

Assembly: Application.dll

```
public class TimelineQueryResult
```

## Inheritance

object ← TimelineQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### ActionBy

```
public ActionBy ActionBy { get; set; }
```

Property Value

[ActionBy](#)

### ActionType

```
public ActionType ActionType { get; set; }
```

Property Value

[ActionType](#)

### Comment

```
public string? Comment { get; set; }
```

Property Value

[string](#) ↗

Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

Timestamp

```
public DateTimeOffset Timestamp { get; set; }
```

Property Value

[DateTimeOffset](#) ↗

VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int](#) ↗

# Namespace Application.Commons.Shared ModelError.Timeline.EntityTimelineDetails Classes

[EntityTimelineDetailsQueryResult](#)

[PropertyChangeResult](#)

# Class EntityTimelineDetailsQueryResult

Namespace: [Application.Commons.SharedModelResult.Timeline.EntityTimelineDetails](#)

Assembly: Application.dll

```
public class EntityTimelineDetailsQueryResult
```

## Inheritance

object ← EntityTimelineDetailsQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### ActionBy

```
public ActionBy ActionBy { get; set; }
```

Property Value

[ActionBy](#)

### ActionType

```
public ActionType ActionType { get; set; }
```

Property Value

[ActionType](#)

### Comment

```
public string? Comment { get; set; }
```

Property Value

[string](#)

Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

ModifiedProperties

```
public IEnumerable<PropertyChangeResult>? ModifiedProperties { get; set; }
```

Property Value

[IEnumerable](#) <[PropertyChangeResult](#)>

Timestamp

```
public DateTimeOffset Timestamp { get; set; }
```

Property Value

[DateTimeOffset](#)

VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int ↗](#)

# Class PropertyChangeResult

Namespace: [Application.Commons.SharedModelResult.Timeline.EntityTimelineDetails](#)

Assembly: Application.dll

```
public class PropertyChangeResult
```

## Inheritance

object ← PropertyChangeResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### NewValue

```
public object? NewValue { get; set; }
```

Property Value

object

### OldValue

```
public object? OldValue { get; set; }
```

Property Value

object

### PropertyName

```
public string PropertyName { get; set; }
```

PropertyValue

[string](#) ↗

PropertyType

```
public string PropertyType { get; set; }
```

PropertyValue

[string](#) ↗

# Namespace Application.Commons.Validation. Pagination

## Classes

[PageRequestValidator](#)

# Class PageRequestValidator

Namespace: [Application.Commons.Validation.Pagination](#)

Assembly: Application.dll

```
public class PageRequestValidator : AbstractValidator<PageRequest>, IValidator<PageRequest>,  
IValidator, IEnumerable<IValidationRule>, IEnumerable
```

## Inheritance

object ← AbstractValidator<[PageRequest](#)> ← PageRequestValidator

## Implements

IValidator<[PageRequest](#)>, IValidator, [IEnumerable](#) ↴ <IValidationRule>, [IEnumerable](#) ↴

## Inherited Members

AbstractValidator<PageRequest>.Validate(PageRequest) ,  
[AbstractValidator<PageRequest>.ValidateAsync\(PageRequest, CancellationToken\)](#) ↴ ,  
AbstractValidator<PageRequest>.Validate(ValidationContext<PageRequest>) ,  
[AbstractValidator<PageRequest>.ValidateAsync\(ValidationContext<PageRequest>, CancellationToken\)](#) ↴ ,  
AbstractValidator<PageRequest>.CreateDescriptor() ,  
[AbstractValidator<PageRequest>.RuleFor<TProperty>\(Expression<Func<PageRequest, TProperty>>\)](#) ↴ ,  
[AbstractValidator<PageRequest>.Transform<TProperty, TTransformed>\(Expression<Func<PageRequest, TProperty>>, Func<TProperty, TTransformed>\)](#) ↴ ,  
[AbstractValidator<PageRequest>.Transform<TProperty, TTransformed>\(Expression<Func<PageRequest, TProperty>>, Func<PageRequest, TProperty, TTransformed>\)](#) ↴ ,  
[AbstractValidator<PageRequest>.RuleForEach<TElement>\(Expression<Func<PageRequest, IEnumerable<TElement>>\)](#) ↴ ,  
[AbstractValidator<PageRequest>.TransformForEach<TElement, TTransformed>\(Expression<Func<PageRequest, IElement>>, Func<TElement, TTransformed>\)](#) ↴ ,  
[AbstractValidator<PageRequest>.TransformForEach<TElement, TTransformed>\(Expression<Func<PageRequest, IElement>>, Func<PageRequest, TElement, TTransformed>\)](#) ↴ ,  
[AbstractValidator<PageRequest>.RuleSet\(string, Action\)](#) ↴ ,  
[AbstractValidator<PageRequest>.When\(Func<PageRequest, bool>, Action\)](#) ↴ ,  
[AbstractValidator<PageRequest>.When\(Func<PageRequest, ValidationContext<PageRequest>, bool>, Action\)](#) ↴ ,  
[AbstractValidator<PageRequest>.Unless\(Func<PageRequest, bool>, Action\)](#) ↴ ,  
[AbstractValidator<PageRequest>.Unless\(Func<PageRequest, ValidationContext<PageRequest>, bool>, Action\)](#) ↴ ,

[AbstractValidator<PageRequest>.WhenAsync\(Func<PageRequest, CancellationToken, Task<bool>>, Action\)](#) ↴ ,  
[AbstractValidator<PageRequest>.WhenAsync\(Func<PageRequest, ValidationContext<PageRequest>, CancellationToken, Task<bool>>, Action\)](#) ↴ ,  
[AbstractValidator<PageRequest>.UnlessAsync\(Func<PageRequest, CancellationToken, Task<bool>>, Action\)](#) ↵ ,  
[AbstractValidator<PageRequest>.UnlessAsync\(Func<PageRequest, ValidationContext<PageRequest>, CancellationToken, Task<bool>>, Action\)](#) ↵ ,  
AbstractValidator<PageRequest>.Include(IValidator<PageRequest>) ,  
[AbstractValidator<PageRequest>.Include<TValidator>\(Func<PageRequest, TValidator>\)](#) ↴ ,  
AbstractValidator<PageRequest>.GetEnumerator() ,  
AbstractValidator<PageRequest>.EnsureInstanceNotNull(object) ,  
AbstractValidator<PageRequest>.PreValidate(ValidationContext<PageRequest>, ValidationResult) ,  
AbstractValidator<PageRequest>.RaiseValidationException(ValidationContext<PageRequest>, ValidationResult) ,  
AbstractValidator<PageRequest>.OnRuleAdded(IValidationRule<PageRequest>) ,  
AbstractValidator<PageRequest>.CascadeMode ,  
AbstractValidator<PageRequest>.ClassLevelCascadeMode ,  
AbstractValidator<PageRequest>.RuleLevelCascadeMode

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Constructors

## PageRequestValidator()

```
public PageRequestValidator()
```

# Namespace Application.EntitiesTimeline. Queries.EntityTimeline

## Classes

[GetEntityTimelineQuery](#)

# Class GetEntityTimelineQuery

Namespace: [Application.EntitiesTimeline.Questions.EntityTimeline](#)

Assembly: Application.dll

```
public class GetEntityTimelineQuery : PageRequest,  
 IRequest<Result<PageResponse<TimelineQueryResult>>>, IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← GetEntityTimelineQuery

## Implements

IRequest<[Result<PageResponse<TimelineQueryResult>>>](#), IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## ActionBy

```
public ActionBy? ActionBy { get; set; }
```

## Property Value

[ActionBy?](#)

## ActionType

```
public ActionType? ActionType { get; set; }
```

## Property Value

[ActionType](#)?

## Comment

```
public string? Comment { get; set; }
```

Property Value

[string](#) ↗

## EndDate

```
public DateTimeOffset? EndDate { get; set; }
```

Property Value

[DateTimeOffset](#) ↗?

## EntityName

```
[JsonIgnore]  
public EntityName EntityName { get; set; }
```

Property Value

[EntityName](#)

## Id

```
[JsonIgnore]  
public Guid Id { get; set; }
```

Property Value

[Guid ↗](#)

## StartDate

```
public DateTimeOffset? StartDate { get; set; }
```

Property Value

[DateTimeOffset ↗](#)?

# Namespace Application.EntitiesTimeline. Queries.EntityTimelineDetails

## Classes

[EntityTimelineDetailsQuery](#)

[EntityTimelineDetailsQueryHandler](#)

# Class EntityTimelineDetailsQuery

Namespace: [Application.EntitiesTimeline.Questions.EntityTimelineDetails](#)

Assembly: Application.dll

```
public class EntityTimelineDetailsQuery :  
IRequest<Result<EntityTimelineDetailsQueryResult>>, IBaseRequest
```

## Inheritance

object ← EntityTimelineDetailsQuery

## Implements

IRequest<[Result<EntityTimelineDetailsQueryResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### EntityId

```
public Guid EntityId { get; set; }
```

#### Property Value

[Guid](#) ↗

### EntityName

```
public EntityName EntityName { get; set; }
```

#### Property Value

[EntityName](#)

## TimelineId

```
public Guid TimelineId { get; set; }
```

Property Value

[Guid](#) ↗

# Class EntityTimelineDetailsQueryHandler

Namespace: [Application.EntitiesTimeline.Questions.EntityTimelineDetails](#)

Assembly: Application.dll

```
public class EntityTimelineDetailsQueryHandler : IRequestHandler<EntityTimelineDetailsQuery, Result<EntityTimelineDetailsQueryResult>>
```

## Inheritance

object ← EntityTimelineDetailsQueryHandler

## Implements

IRequestHandler<[EntityTimelineDetailsQuery](#), [Result<EntityTimelineDetailsQueryResult>](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

EntityTimelineDetailsQueryHandler(IAuditManager, IMapper)

```
public EntityTimelineDetailsQueryHandler(IAuditManager auditManager, IMapper mapper)
```

## Parameters

auditManager [IAuditManager](#)

mapper IMapper

## Methods

Handle(EntityTimelineDetailsQuery, CancellationToken)

Handles a request

```
public Task<Result<EntityTimelineDetailsQueryResult>> Handle(EntityTimelineDetailsQuery
```

```
request, CancellationToken cancellationToken)
```

## Parameters

request [EntityTimelineDetailsQuery](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[EntityTimelineDetailsQueryResult](#)>>

Response from the request

# Namespace Application.Histories.Queries.GetQuestionHistory

## Classes

[GetQuestionHistoryQuery](#)

# Class GetQuestionHistoryQuery

Namespace: [Application.Histories.Queries.GetQuestionHistory](#)

Assembly: Application.dll

```
public class GetQuestionHistoryQuery : PageRequest,  
 IRequest<PageResponse<GetQuestionHistoryQueryResult>>, IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← GetQuestionHistoryQuery

## Implements

IRequest<[PageResponse](#)<GetQuestionHistoryQueryResult>>, IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## QuestionId

```
[JsonIgnore]  
public Guid QuestionId { get; set; }
```

## Property Value

[Guid](#) ↗

# Namespace Application.Histories.Queries.Get QuestionHistory.Models

## Classes

[PropertyChangesQueryResult](#)

# Class PropertyChangesQueryResult

Namespace: [Application.Histories.Queries.GetQuestionHistory.Models](#)

Assembly: Application.dll

```
public class PropertyChangesQueryResult
```

## Inheritance

object ← PropertyChangesQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### DataType

```
public string DataType { get; set; }
```

### Property Value

[string](#) ↗

### IsNullable

```
public bool IsNullable { get; set; }
```

### Property Value

[bool](#) ↗

### Name

```
public string Name { get; set; }
```

Property Value

string ↗

NewValue

```
public object? NewValue { get; set; }
```

Property Value

object

OldValue

```
public object? OldValue { get; set; }
```

Property Value

object

# Namespace Application.Lookups.Commands. Languages

## Classes

[AddLanguageCommand](#)

[AddLanguageCommandHandler](#)

# Class AddLanguageCommand

Namespace: [Application.Lookups.Commands.Languages](#)

Assembly: Application.dll

```
public class AddLanguageCommand : IRequest, IBaseRequest
```

## Inheritance

object ← AddLanguageCommand

## Implements

IRequest, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Code

```
public required string Code { get; set; }
```

## Property Value

[string](#) ↗

## DisplayName

```
public required string DisplayName { get; set; }
```

## Property Value

[string](#) ↗

# Class AddLanguageCommandHandler

Namespace: [Application.Lookups.Commands.Languages](#)

Assembly: Application.dll

```
public class AddLanguageCommandHandler : IRequestHandler<AddLanguageCommand>
```

## Inheritance

object ← AddLanguageCommandHandler

## Implements

IRequestHandler<[AddLanguageCommand](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### AddLanguageCommandHandler(IServiceManager)

```
public AddLanguageCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle/AddLanguageCommand, CancellationToken)

Handles a request

```
public Task Handle(AddLanguageCommand request, CancellationToken cancellationToken)
```

## Parameters

request [AddLanguageCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

Returns

[Task](#)

Response from the request

# Namespace Application.Lookups.Commands. Sources.UpdateSource

## Classes

[UpdateSourceCommand](#)

[UpdateSourceCommandHandler](#)

# Class UpdateSourceCommand

Namespace: [Application.Lookups.Commands.Sources.UpdateSource](#)

Assembly: Application.dll

```
public class UpdateSourceCommand : IRequest, IBaseRequest
```

## Inheritance

object ← UpdateSourceCommand

## Implements

IRequest, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Description

```
public string? Description { get; set; }
```

### Property Value

[string](#) ↗

## Id

```
public Guid Id { get; set; }
```

### Property Value

[Guid](#) ↗

## Tags

```
public List<Guid>? Tags { get; set; }
```

### Property Value

[List](#) <[Guid](#)>

## Title

```
public string Title { get; set; }
```

### Property Value

[string](#)

## Type

```
public SourceType Type { get; set; }
```

### Property Value

[SourceType](#)

# Class UpdateSourceCommandHandler

Namespace: [Application.Lookups.Commands.Sources.UpdateSource](#)

Assembly: Application.dll

```
public class UpdateSourceCommandHandler : IRequestHandler<UpdateSourceCommand>
```

## Inheritance

object ← UpdateSourceCommandHandler

## Implements

IRequestHandler<[UpdateSourceCommand](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### UpdateSourceCommandHandler(IServiceManager)

```
public UpdateSourceCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(UpdateSourceCommand, CancellationToken)

Handles a request

```
public Task Handle(UpdateSourceCommand request, CancellationToken cancellationToken)
```

## Parameters

request [UpdateSourceCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

Returns

[Task](#)

Response from the request

# Namespace Application.Lookups.Queries. Languages.GetLanguages

## Classes

[GetLanguagesQuery](#)

[GetLanguagesQueryHandler](#)

[GetLanguagesQueryResult](#)

# Class GetLanguagesQuery

Namespace: [Application.Lookups.Queries.Languages.GetLanguages](#)

Assembly: Application.dll

```
public class GetLanguagesQuery : IRequest<IEnumerable<GetLanguagesQueryResult>>,  
IBaseRequest
```

## Inheritance

object ← GetLanguagesQuery

## Implements

IRequest<[IEnumerable](#)<[GetLanguagesQueryResult](#)>>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Class GetLanguagesQueryHandler

Namespace: [Application.Lookups.Queries.Languages.GetLanguages](#)

Assembly: Application.dll

```
public class GetLanguagesQueryHandler : IRequestHandler<GetLanguagesQuery,  
IEnumerable<GetLanguagesQueryResult>>
```

## Inheritance

object ← GetLanguagesQueryHandler

## Implements

IRequestHandler<[GetLanguagesQuery](#), [IEnumerable](#)<[GetLanguagesQueryResult](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### GetLanguagesQueryHandler(IServiceManager)

```
public GetLanguagesQueryHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(GetLanguagesQuery, CancellationToken)

Handles a request

```
public Task<IEnumerable<GetLanguagesQueryResult>> Handle(GetLanguagesQuery request,  
CancellationToken cancellationToken)
```

## Parameters

request [GetLanguagesQuery](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[IEnumerable](#) <[GetLanguagesQueryResult](#)>>

Response from the request

# Class GetLanguagesQueryResult

Namespace: [Application.Lookups.Queries.Languages.GetLanguages](#)

Assembly: Application.dll

```
public class GetLanguagesQueryResult
```

## Inheritance

object ← GetLanguagesQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Code

```
public required string Code { get; set; }
```

#### Property Value

[string](#) ↗

### DisplayName

```
public required string DisplayName { get; set; }
```

#### Property Value

[string](#) ↗

### Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

# Namespace Application.Lookups.Queries.Tags. GetAllTags

## Classes

[GetAllTagsQuery](#)

[GetAllTagsQueryHandler](#)

[GetAllTagsQueryResult](#)

# Class GetAllTagsQuery

Namespace: [Application.Lookups.Queries.Tags.GetAllTags](#)

Assembly: Application.dll

```
public class GetAllTagsQuery : PageRequest,  
 IRequest<Result<PageResponse<GetAllTagsQueryResult>>>, IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← GetAllTagsQuery

## Implements

IRequest<[Result<PageResponse<GetAllTagsQueryResult>>](#)>, IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## BackgroundColorGroup

```
public ColorCategory? BackgroundColorGroup { get; set; }
```

## Property Value

[ColorCategory?](#)

## IsArchived

```
public bool? IsArchived { get; set; }
```

## Property Value

[bool](#)?

## Search

`public string? Search { get; set; }`

Property Value

[string](#)

## TextColorGroup

`public ColorCategory? TextColorGroup { get; set; }`

Property Value

[ColorCategory](#)?

# Class GetAllTagsQueryHandler

Namespace: [Application.Lookups.Queries.Tags.GetAllTags](#)

Assembly: Application.dll

```
public class GetAllTagsQueryHandler : IRequestHandler<GetAllTagsQuery,
Result<PageResponse<GetAllTagsQueryResult>>>
```

## Inheritance

object ← GetAllTagsQueryHandler

## Implements

IRequestHandler<[GetAllTagsQuery](#), [Result<PageResponse<GetAllTagsQueryResult>>](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### GetAllTagsQueryHandler(IRepositoryManager, IMapper)

```
public GetAllTagsQueryHandler(IRepositoryManager repositoryManager, IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

mapper IMapper

## Methods

### Handle(GetAllTagsQuery, CancellationToken)

Handles a request

```
public Task<Result<PageResponse<GetAllTagsQueryResult>>> Handle(GetAllTagsQuery request,
```

```
CancellationToken cancellationToken)
```

## Parameters

request [GetAllTagsQuery](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[PageResponse](#)<[GetAllTagsQueryResult](#)>>>

Response from the request

# Class GetAllTagsQueryResult

Namespace: [Application.Lookups.Queries.Tags.GetAllTags](#)

Assembly: Application.dll

```
public class GetAllTagsQueryResult
```

## Inheritance

object ← GetAllTagsQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### BackgroundColorCode

```
public string BackgroundColorCode { get; set; }
```

Property Value

[string](#) ↗

### BackgroundColorGroup

```
public ColorCategory BackgroundColorGroup { get; set; }
```

Property Value

[ColorCategory](#)

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

IsArchived

```
public bool IsArchived { get; set; }
```

Property Value

[bool](#)

Name

```
public string Name { get; set; }
```

Property Value

[string](#)

TextColorCode

```
public string TextColorCode { get; set; }
```

Property Value

[string](#)

TextColorGroup

```
public ColorCategory TextColorGroup { get; set; }
```

## Property Value

[ColorCategory](#)

# Namespace Application.Questions.Commands. AddQuestion

## Classes

[AddQuestionCommand](#)

[AddQuestionCommandResult](#)

# Class AddQuestionCommand

Namespace: [Application.Questions.Commands.AddQuestion](#)

Assembly: Application.dll

```
public class AddQuestionCommand : IRequest<AddQuestionCommandResult>, IBaseRequest
```

## Inheritance

object ← AddQuestionCommand

## Implements

IRequest<[AddQuestionCommandResult](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Category

```
public Guid? Category { get; set; }
```

## Property Value

[Guid](#)?

## Difficulty

```
public QuestionDifficulty Difficulty { get; set; }
```

## Property Value

[QuestionDifficulty](#)

## Language

```
public Guid? Language { get; set; }
```

### Property Value

[Guid](#)?

## LongAnswer

```
public LongAnswerQuestionCommand? LongAnswer { get; set; }
```

### Property Value

[LongAnswerQuestionCommand](#)

## Mark

```
public required int Mark { get; set; }
```

### Property Value

[int](#)

## Options

```
public IEnumerable<MultipleChoiseQuestionCommand>? Options { get; set; }
```

### Property Value

[IEnumerable](#) <[MultipleChoiseQuestionCommand](#)>

## QuestionText

Question text.

```
public required string QuestionText { get; set; }
```

Property Value

[string](#)

## QuestionType

Question Type. witch can be the following 1- Multiple Choise. 2- True and false.

```
public required QuestionType QuestionType { get; set; }
```

Property Value

[QuestionType](#)

## Reordering

```
public IEnumerable<ReorderingQuestionCommand>? Reordering { get; set; }
```

Property Value

[IEnumerable](#) <[ReorderingQuestionCommand](#)>

## RequireManualReview

```
public required bool RequireManualReview { get; set; }
```

Property Value

[bool](#)

## ShortAnswer

```
public ShortAnswerQuestionCommand? ShortAnswer { get; set; }
```

### Property Value

[ShortAnswerQuestionCommand](#)

## Sources

```
public List<Guid>? Sources { get; set; }
```

### Property Value

[List<Guid>](#)

## Tags

```
public List<Guid>? Tags { get; set; }
```

### Property Value

[List<Guid>](#)

## TrueAndFalseAnswer

```
public TrueFalseQuestionCommand? TrueAndFalseAnswer { get; set; }
```

### Property Value

[TrueFalseQuestionCommand](#)

## Variants

List of the same question but different style

```
public List<string>? Variants { get; set; }
```

Property Value

List <string>

# Class AddQuestionCommandResult

Namespace: [Application.Questions.Commands.AddQuestion](#)

Assembly: Application.dll

```
public class AddQuestionCommandResult
```

## Inheritance

object ← AddQuestionCommandResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

# Namespace Application.Questions.Commands. AddQuestion.CommandModels

## Classes

[FeedbackCommand](#)

[LongAnswerQuestionCommand](#)

[MultipleChoiseQuestionCommand](#)

[QuestionSourceCommand](#)

[ReorderingQuestionCommand](#)

[ShortAnswerQuestionCommand](#)

[TrueFalseQuestionCommand](#)

# Class FeedbackCommand

Namespace: [Application.Questions.Commands.AddQuestion.CommandModels](#)

Assembly: Application.dll

```
public class FeedbackCommand
```

## Inheritance

object ← FeedbackCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### CorrectAnswerFeedback

```
public string CorrectAnswerFeedback { get; set; }
```

Property Value

[string](#) ↗

### WrongAnswerFeedback

```
public string WrongAnswerFeedback { get; set; }
```

Property Value

[string](#) ↗

# Class LongAnswerQuestionCommand

Namespace: [Application.Questions.Commands.AddQuestion.CommandModels](#)

Assembly: Application.dll

```
public class LongAnswerQuestionCommand
```

## Inheritance

object ← LongAnswerQuestionCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Answer

```
public string Answer { get; set; }
```

#### Property Value

[string](#) ↗

### GeneralFeedback

```
public string? GeneralFeedback { get; set; }
```

#### Property Value

[string](#) ↗

### MaximinWords

```
public int? MaximinWords { get; set; }
```

Property Value

int?

## MinimanWords

```
public int? MinimanWords { get; set; }
```

Property Value

int?

# Class MultipleChoiseQuestionCommand

Namespace: [Application.Questions.Commands.AddQuestion.CommandModels](#)

Assembly: Application.dll

```
public class MultipleChoiseQuestionCommand
```

## Inheritance

object ← MultipleChoiseQuestionCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### FeedBack

```
public string? FeedBack { get; set; }
```

Property Value

[string](#) ↗

### IsCorrect

```
public bool IsCorrect { get; set; }
```

Property Value

[bool](#) ↗

### OptionText

```
public string OptionText { get; set; }
```

Property Value

[string](#)

## Weight

```
public float Weight { get; set; }
```

Property Value

[float](#)

# Class QuestionSourceCommand

Namespace: [Application.Questions.Commands.AddQuestion.CommandModels](#)

Assembly: Application.dll

```
public class QuestionSourceCommand
```

## Inheritance

object ← QuestionSourceCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Description

```
public string? Description { get; set; }
```

#### Property Value

[string](#) ↗

### Title

```
public string Title { get; set; }
```

#### Property Value

[string](#) ↗

### Type

```
public SourceType Type { get; set; }
```

Property Value

[SourceType](#)

URL

```
public string URL { get; set; }
```

Property Value

[string](#) ↗

# Class ReorderingQuestionCommand

Namespace: [Application.Questions.Commands.AddQuestion.CommandModels](#)

Assembly: Application.dll

```
public class ReorderingQuestionCommand
```

## Inheritance

object ← ReorderingQuestionCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Feedback

```
public FeedbackCommand? Feedback { get; set; }
```

Property Value

[FeedbackCommand](#)

### Order

```
public int Order { get; set; }
```

Property Value

[int↗](#)

### Value

```
public string Value { get; set; }
```

Property Value

[string](#) ↗

# Class ShortAnswerQuestionCommand

Namespace: [Application.Questions.Commands.AddQuestion.CommandModels](#)

Assembly: Application.dll

```
public class ShortAnswerQuestionCommand
```

## Inheritance

object ← ShortAnswerQuestionCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### CorrectAnswer

```
public required string CorrectAnswer { get; set; }
```

#### Property Value

[string](#) ↗

### Feedback

```
public FeedbackCommand? Feedback { get; set; }
```

#### Property Value

[FeedbackCommand](#)

### PossibleAnswers

```
public List<string>? PossibleAnswers { get; set; }
```

Property Value

[List](#) <[string](#)>

# Class TrueFalseQuestionCommand

Namespace: [Application.Questions.Commands.AddQuestion.CommandModels](#)

Assembly: Application.dll

```
public class TrueFalseQuestionCommand
```

## Inheritance

object ← TrueFalseQuestionCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Feedback

```
public FeedbackCommand? Feedback { get; set; }
```

Property Value

[FeedbackCommand](#)

### IsCorrect

```
public bool IsCorrect { get; set; }
```

Property Value

[bool](#) ↗

# Namespace Application.Questions.Commands. DeleteAllQuestion

## Classes

[DeleteAllQuestionCommand](#)

# Class DeleteAllQuestionCommand

Namespace: [Application.Questions.Commands.DeleteAllQuestion](#)

Assembly: Application.dll

```
public class DeleteAllQuestionCommand : IRequest, IBaseRequest
```

## Inheritance

object ← DeleteAllQuestionCommand

## Implements

IRequest, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Namespace Application.Questions.Commands. DeleteQuestion

## Classes

[DeleteQuestionCommand](#)

# Class DeleteQuestionCommand

Namespace: [Application.Questions.Commands.DeleteQuestion](#)

Assembly: Application.dll

```
public class DeleteQuestionCommand : IRequest, IBaseRequest
```

## Inheritance

object ← DeleteQuestionCommand

## Implements

IRequest, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

# Namespace Application.Questions.Commands. UpdateQuestion

## Classes

[UpdateQuestionCommand](#)

[UpdateQuestionCommandHandler](#)

# Class UpdateQuestionCommand

Namespace: [Application.Questions.Commands.UpdateQuestion](#)

Assembly: Application.dll

```
public class UpdateQuestionCommand : IRequest, IBaseRequest
```

## Inheritance

object ← UpdateQuestionCommand

## Implements

IRequest, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Difficulty

```
public QuestionDifficulty Difficulty { get; set; }
```

## Property Value

[QuestionDifficulty](#)

## Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

## Mark

```
public int Mark { get; set; }
```

### Property Value

[int ↗](#)

## QuestionText

```
public string QuestionText { get; set; }
```

### Property Value

[string ↗](#)

## RequireManulReview

```
public bool RequireManulReview { get; set; }
```

### Property Value

[bool ↗](#)

## Sources

```
public List<QuestionSourceUpdateCommand>? Sources { get; set; }
```

### Property Value

[List ↗ <QuestionSourceUpdateCommand>](#)

## Tags

```
public List<string>? Tags { get; set; }
```

Property Value

[List](#) <[string](#)>

## Variants

```
public List<string>? Variants { get; set; }
```

Property Value

[List](#) <[string](#)>

# Class UpdateQuestionCommandHandler

Namespace: [Application.Questions.Commands.UpdateQuestion](#)

Assembly: Application.dll

```
public class UpdateQuestionCommandHandler : IRequestHandler<UpdateQuestionCommand>
```

## Inheritance

object ← UpdateQuestionCommandHandler

## Implements

IRequestHandler<[UpdateQuestionCommand](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### UpdateQuestionCommandHandler(IServiceManager)

```
public UpdateQuestionCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(UpdateQuestionCommand, CancellationToken)

Handles a request

```
public Task Handle(UpdateQuestionCommand request, CancellationToken cancellationToken)
```

## Parameters

request [UpdateQuestionCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

Returns

[Task](#)

Response from the request

# Namespace Application.Questions.Commands. UpdateQuestion.Models

## Classes

[QuestionSourceUpdateCommand](#)

# Class QuestionSourceUpdateCommand

Namespace: [Application.Questions.Commands.UpdateQuestion.Models](#)

Assembly: Application.dll

```
public class QuestionSourceUpdateCommand
```

## Inheritance

object ← QuestionSourceUpdateCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Description

```
public string? Description { get; set; }
```

#### Property Value

[string](#) ↗

### ISBN

```
public string? ISBN { get; set; }
```

#### Property Value

[string](#) ↗

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

Title

```
public string Title { get; set; }
```

Property Value

[string](#) ↗

Type

```
public SourceType Type { get; set; }
```

Property Value

[SourceType](#)

URL

```
public string URL { get; set; }
```

Property Value

[string](#) ↗

# Namespace Application.Questions.Queries.GetAllQuestions

## Classes

[GetAllQuestionsQuery](#)

[GetAllQuestionsQueryHandler](#)

[GetAllQuestionsQueryResult](#)

# Class GetAllQuestionsQuery

Namespace: [Application.Questions.Questions.GetAllQuestions](#)

Assembly: Application.dll

```
public class GetAllQuestionsQuery : PageRequest,  
 IRequest<PageResponse<GetAllQuestionsQueryResult>>, IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← GetAllQuestionsQuery

## Implements

IRequest<[PageResponse](#)<[GetAllQuestionsQueryResult](#)>>, IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Category

```
public string? Category { get; set; }
```

## Property Value

[string](#) ↗

## QuestionType

```
public QuestionType? QuestionType { get; set; }
```

## Property Value

[QuestionType](#)?

## RequireManualReview

`public bool? RequireManualReview { get; set; }`

Property Value

[bool](#)?

## Search

`public string? Search { get; set; }`

Property Value

[string](#)?

## Tags

`public string? Tags { get; set; }`

Property Value

[string](#)?

# Class GetAllQuestionsQueryHandler

Namespace: [Application.Questions.Questions.GetAllQuestions](#)

Assembly: Application.dll

```
public class GetAllQuestionsQueryHandler : IRequestHandler<GetAllQuestionsQuery,
PageResponse<GetAllQuestionsQueryResult>>
```

## Inheritance

object ← GetAllQuestionsQueryHandler

## Implements

IRequestHandler<[GetAllQuestionsQuery](#), [PageResponse<GetAllQuestionsQueryResult>](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### GetAllQuestionsQueryHandler(IRepositoryManager)

```
public GetAllQuestionsQueryHandler(IRepositoryManager repositoryManager)
```

## Parameters

repositoryManager [IRepositoryManager](#)

## Methods

### Handle(GetAllQuestionsQuery, CancellationToken)

Handles a request

```
public Task<PageResponse<GetAllQuestionsQueryResult>> Handle(GetAllQuestionsQuery request,
CancellationToken cancellationToken)
```

## Parameters

request  [GetAllQuestionsQuery](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) < [PageResponse](#)< [GetAllQuestionsQueryResult](#)>>

Response from the request

# Class GetAllQuestionsQueryResult

Namespace: [Application.Questions.Quires.GetAllQuestions](#)

Assembly: Application.dll

```
public class GetAllQuestionsQueryResult
```

## Inheritance

object ← GetAllQuestionsQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Category

```
public string? Category { get; set; }
```

#### Property Value

[string](#) ↗

### Difficulty

```
public QuestionDifficulty Difficulty { get; set; }
```

#### Property Value

[QuestionDifficulty](#)

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

Mark

```
public int Mark { get; set; }
```

Property Value

[int](#) ↗

QuestionText

```
public string QuestionText { get; set; }
```

Property Value

[string](#) ↗

QuestionType

```
public QuestionType QuestionType { get; set; }
```

Property Value

[QuestionType](#)

RequireManualReview

```
public bool RequireManualReview { get; set; }
```

Property Value

[bool](#)

Tags

```
public IEnumerable<TagResult>? Tags { get; set; }
```

Property Value

[IEnumerable](#) <[TagResult](#)>

# Namespace Application.Questions.Queries.GetQuestionsById

## Classes

[GetQuestionsByIdQuery](#)

[GetQuestionsByIdQueryResult](#)

# Class GetQuestionsByIdQuery

Namespace: [Application.Questions.Questions.GetById](#)

Assembly: Application.dll

```
public class GetQuestionsByIdQuery : IRequest<GetQuestionsByIdQueryResult>, IBaseRequest
```

## Inheritance

object ← GetQuestionsByIdQuery

## Implements

IRequest<[GetQuestionsByIdQueryResult](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

# Class GetQuestionsByIdQueryResult

Namespace: [Application.Questions.Questions.GetById](#)

Assembly: Application.dll

```
public class GetQuestionsByIdQueryResult
```

## Inheritance

object ← GetQuestionsByIdQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Category

```
public string? Category { get; set; }
```

#### Property Value

[string](#) ↗

### Difficulty

```
public QuestionDifficultyLookup Difficulty { get; set; }
```

#### Property Value

[QuestionDifficultyLookup](#)

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid ↗](#)

## Language

```
public LanguageResult? Language { get; set; }
```

Property Value

[LanguageResult](#)

## LongAnswer

```
public LongAnswerQuestionQueryResult? LongAnswer { get; set; }
```

Property Value

[LongAnswerQuestionQueryResult](#)

## Mark

```
public int Mark { get; set; }
```

Property Value

[int ↗](#)

## MultipleChoiceOptions

```
public IEnumerable<MultipleChoiseQuestionResult>? MultipleChoiceOptions { get; set; }
```

Property Value

[IEnumerable](#) <[MultipleChoiseQuestionResult](#)>

## QuestionText

```
public string QuestionText { get; set; }
```

Property Value

[string](#)

## QuestionType

```
public QuestionTypeLookup QuestionType { get; set; }
```

Property Value

[QuestionTypeLookup](#)

## Reordering

```
public IEnumerable<ReorderingQuestionQueryResult>? Reordering { get; set; }
```

Property Value

[IEnumerable](#) <[ReorderingQuestionQueryResult](#)>

## RequireManualReview

```
public bool RequireManualReview { get; set; }
```

Property Value

[bool](#)

## ShortAnswer

```
public ShortAnswerQuestionQueryResult? ShortAnswer { get; set; }
```

Property Value

[ShortAnswerQuestionQueryResult](#)

## Sources

```
public List<QuestionSourceResult>? Sources { get; set; }
```

Property Value

[List](#) <[QuestionSourceResult](#)>

## Tags

```
public List<TagResult>? Tags { get; set; }
```

Property Value

[List](#) <[TagResult](#)>

## TrueAndFalse

```
public TrueFalseQuestionQueryResult? TrueAndFalse { get; set; }
```

Property Value

[TrueFalseQuestionQueryResult](#)

## Variants

```
public List<string>? Variants { get; set; }
```

Property Value

[List](#) <[string](#)>

## VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int](#)

# Namespace Application.Questions.Queries.GetQuestionsById.ResultModel

## Classes

[FeedbackQueryResult](#)

[LongAnswerQuestionQueryResult](#)

[MultipleChoiseQuestionResult](#)

[ReorderingQuestionQueryResult](#)

[ShortAnswerQuestionQueryResult](#)

[TrueFalseQuestionQueryResult](#)

# Class FeedbackQueryResult

Namespace: [Application.Questions.QuestionsById.ResultModel](#)

Assembly: Application.dll

```
public class FeedbackQueryResult
```

## Inheritance

object ← FeedbackQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### CorrectAnswerFeedback

```
public string CorrectAnswerFeedback { get; set; }
```

Property Value

[string](#) ↗

### WrongAnswerFeedback

```
public string WrongAnswerFeedback { get; set; }
```

Property Value

[string](#) ↗

# Class LongAnswerQuestionQueryResult

Namespace: [Application.Questions.QuestionsById.ResultModel](#)

Assembly: Application.dll

```
public class LongAnswerQuestionQueryResult
```

## Inheritance

object ← LongAnswerQuestionQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Answer

```
public string Answer { get; set; }
```

#### Property Value

[string](#) ↗

### GeneralFeedback

```
public string? GeneralFeedback { get; set; }
```

#### Property Value

[string](#) ↗

### MaximinWords

```
public int? MaximinWords { get; set; }
```

Property Value

int?

MinimanWords

```
public int? MinimanWords { get; set; }
```

Property Value

int?

# Class MultipleChoiseQuestionResult

Namespace: [Application.Questions.QuestionsById.ResultModel](#)

Assembly: Application.dll

```
public class MultipleChoiseQuestionResult
```

## Inheritance

object ← MultipleChoiseQuestionResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### FeedBack

```
public string? FeedBack { get; set; }
```

Property Value

[string](#) ↗

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

### IsCorrect

```
public bool IsCorrect { get; set; }
```

Property Value

[bool](#)

## OptionText

```
public string OptionText { get; set; }
```

Property Value

[string](#)

## Weight

```
public float Weight { get; set; }
```

Property Value

[float](#)

# Class ReorderingQuestionQueryResult

Namespace: [Application.Questions.QuestionsById.ResultModel](#)

Assembly: Application.dll

```
public class ReorderingQuestionQueryResult
```

## Inheritance

object ← ReorderingQuestionQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Feedback

```
public FeedbackQueryResult? Feedback { get; set; }
```

#### Property Value

[FeedbackQueryResult](#)

### Id

```
public Guid Id { get; set; }
```

#### Property Value

[Guid](#) ↗

### Order

```
public int Order { get; set; }
```

Property Value

[int ↗](#)

Value

```
public string Value { get; set; }
```

Property Value

[string ↗](#)

# Class ShortAnswerQuestionQueryResult

Namespace: [Application.Questions.QuestionsById.ResultModel](#)

Assembly: Application.dll

```
public class ShortAnswerQuestionQueryResult
```

## Inheritance

object ← ShortAnswerQuestionQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### CorrectAnswer

```
public required string CorrectAnswer { get; set; }
```

#### Property Value

[string](#) ↗

### Feedback

```
public FeedbackQueryResult? Feedback { get; set; }
```

#### Property Value

[FeedbackQueryResult](#)

## PossibleAnswers

```
public List<string>? PossibleAnswers { get; set; }
```

Property Value

[List](#) <[string](#)>

# Class TrueFalseQuestionQueryResult

Namespace: [Application.Questions.QuestionsById.ResultModel](#)

Assembly: Application.dll

```
public class TrueFalseQuestionQueryResult
```

## Inheritance

object ← TrueFalseQuestionQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Feedback

```
public FeedbackQueryResult? Feedback { get; set; }
```

#### Property Value

[FeedbackQueryResult](#)

### Id

```
public Guid Id { get; set; }
```

#### Property Value

[Guid](#) ↗

### IsCorrect

```
public bool IsCorrect { get; set; }
```

Property Value

[bool](#) ↗

# Namespace Application.Questions.Queries.GetRandomQuestions

## Classes

[GetRandomQuestionsQuery](#)

# Class GetRandomQuestionsQuery

Namespace: [Application.Questions.Questions](#)

Assembly: Application.dll

```
public class GetRandomQuestionsQuery
```

## Inheritance

object ← GetRandomQuestionsQuery

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### CategoryId

```
public Guid CategoryId { get; set; }
```

Property Value

[Guid](#) ↗

# Namespace Application.Sources.Commands. AddBulkTagsToSource

## Classes

[AddBulkTagsToSourceCommand](#)

[AddBulkTagsToSourceCommandHandler](#)

# Class AddBulkTagsToSourceCommand

Namespace: [Application.Sources.Commands.AddBulkTagsToSource](#)

Assembly: Application.dll

```
public class AddBulkTagsToSourceCommand : IRequest<Result<PartialsSuccessResult>>,  
IBaseRequest
```

## Inheritance

object ← AddBulkTagsToSourceCommand

## Implements

IRequest<[Result<PartialsSuccessResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### SourceId

```
[JsonIgnore]  
public Guid SourceId { get; set; }
```

### Property Value

[Guid](#)

### TagIds

```
public IEnumerable<Guid> TagIds { get; set; }
```

### Property Value

[IEnumerable](#) <[Guid](#)>

# Class AddBulkTagsToSourceCommandHandler

Namespace: [Application.Sources.Commands.AddBulkTagsToSource](#)

Assembly: Application.dll

```
public class AddBulkTagsToSourceCommandHandler : IRequestHandler<AddBulkTagsToSourceCommand,  
Result<PartialsSuccessResult>>
```

## Inheritance

object ← AddBulkTagsToSourceCommandHandler

## Implements

IRequestHandler<[AddBulkTagsToSourceCommand](#), [Result<PartialsSuccessResult>](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### AddBulkTagsToSourceCommandHandler(IServiceManager)

```
public AddBulkTagsToSourceCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(AddBulkTagsToSourceCommand, CancellationToken)

Handles a request

```
public Task<Result<PartialsSuccessResult>> Handle(AddBulkTagsToSourceCommand request,  
CancellationToken cancellationToken)
```

## Parameters

request [AddBulkTagsToSourceCommand](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[PartialsSuccessResult](#)>>

Response from the request

# Namespace Application.Sources.Commands. AddReference

## Classes

[AddReferenceCommand](#)

[AddReferenceCommandHandler](#)

# Class AddReferenceCommand

Namespace: [Application.Sources.Commands.AddReference](#)

Assembly: Application.dll

```
public class AddReferenceCommand : IRequest, IBaseRequest
```

## Inheritance

object ← AddReferenceCommand

## Implements

IRequest, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Metadata

```
public List<AddMetadataCommand>? Metadata { get; set; }
```

## Property Value

[List](#) <[AddMetadataCommand](#)>

## Notes

```
public string? Notes { get; set; }
```

## Property Value

[string](#)

## SourceId

```
public Guid SourceId { get; set; }
```

### Property Value

[Guid ↗](#)

# Class AddReferenceCommandHandler

Namespace: [Application.Sources.Commands.AddReference](#)

Assembly: Application.dll

```
public class AddReferenceCommandHandler
```

## Inheritance

object ← AddReferenceCommandHandler

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Namespace Application.Sources.Commands. AddSource

## Classes

[AddSourceCommandHandler](#)

# Class AddSourceCommandHandler

Namespace: [Application.Sources.Commands.AddSource](#)

Assembly: Application.dll

```
public class AddSourceCommandHandler : IRequestHandler<AddSourceCommand,  
Result<AddSourceCommandResult>>
```

## Inheritance

object ← AddSourceCommandHandler

## Implements

IRequestHandler<[AddSourceCommand](#), [Result<AddSourceCommandResult>](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### AddSourceCommandHandler(IServiceManager, IMapper)

```
public AddSourceCommandHandler(IServiceManager serviceManager, IMapper mapper)
```

## Parameters

serviceManager [IServiceManager](#)

mapper IMapper

## Methods

### Handle(AddSourceCommand, CancellationToken)

Handles a request

```
public Task<Result<AddSourceCommandResult>> Handle(AddSourceCommand request,
```

```
CancellationToken cancellationToken)
```

## Parameters

request [AddSourceCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[AddSourceCommandResult](#)>>

Response from the request

# Namespace Application.Sources.Commands. AddSource.Requests

## Classes

[AddSourceCommand](#)

[AddSourceReferenceCommand](#)

# Class AddSourceCommand

Namespace: [Application.Sources.Commands.AddSource.Requests](#)

Assembly: Application.dll

```
public class AddSourceCommand : IRequest<Result<AddSourceCommandResult>>, IBaseRequest
```

## Inheritance

object ← AddSourceCommand

## Implements

IRequest<[Result<AddSourceCommandResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### CategoryId

```
public Guid? CategoryId { get; set; }
```

#### Property Value

[Guid](#)?

### Description

```
public string? Description { get; set; }
```

#### Property Value

[string](#)?

## FileExtension

```
public string? FileExtension { get; set; }
```

Property Value

[string](#) ↗

## FilePath

```
public string? FilePath { get; set; }
```

Property Value

[string](#) ↗

## HasAttachment

```
public bool HasAttachment { get; set; }
```

Property Value

[bool](#) ↗

## Metadata

```
public List<AddMetadataCommand>? Metadata { get; set; }
```

Property Value

[List](#) ↗ <[AddMetadataCommand](#)>

## References

```
public List<AddSourceReferenceCommand>? References { get; set; }
```

## Property Value

[List](#) <[AddSourceReferenceCommand](#)>

## Tags

```
public List<Guid>? Tags { get; set; }
```

## Property Value

[List](#) <[Guid](#)>

## Title

```
public string Title { get; set; }
```

## Property Value

[string](#)

## Type

```
public SourceType Type { get; set; }
```

## Property Value

[SourceType](#)

# Class AddSourceReferenceCommand

Namespace: [Application.Sources.Commands.AddSource.Requests](#)

Assembly: Application.dll

```
public class AddSourceReferenceCommand
```

## Inheritance

object ← AddSourceReferenceCommand

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Metadata

```
public List<AddMetadataCommand> Metadata { get; set; }
```

### Property Value

[List](#) <[AddMetadataCommand](#)>

### Notes

```
public string? Notes { get; set; }
```

### Property Value

[string](#)

# Namespace Application.Sources.Commands. AddSource.Results

## Classes

[AddSourceCommandResult](#)

[SourceReferenceResult](#)

# Class AddSourceCommandResult

Namespace: [Application.Sources.Commands.AddSource.Results](#)

Assembly: Application.dll

```
public class AddSourceCommandResult
```

## Inheritance

object ← AddSourceCommandResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Description

```
public string? Description { get; set; }
```

#### Property Value

[string](#) ↗

### FileExtension

```
public string? FileExtension { get; set; }
```

#### Property Value

[string](#) ↗

### FilePath

```
public string? FilePath { get; set; }
```

Property Value

[string](#)

## HasAttachment

```
public bool HasAttachment { get; set; }
```

Property Value

[bool](#)

## Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

## Metadata

```
public IEnumerable<MetadataResult>? Metadata { get; set; }
```

Property Value

[IEnumerable](#)<[MetadataResult](#)>

## References

```
public IEnumerable<AddSourceReferenceCommand>? References { get; set; }
```

## Property Value

[IEnumerable](#) <[AddSourceReferenceCommand](#)>

## Tags

```
public IEnumerable<Guid>? Tags { get; set; }
```

## Property Value

[IEnumerable](#) <[Guid](#)>

## Title

```
public string Title { get; set; }
```

## Property Value

[string](#)

## Type

```
public SourceType Type { get; set; }
```

## Property Value

[SourceType](#)

# Class SourceReferenceResult

Namespace: [Application.Sources.Commands.AddSource.Results](#)

Assembly: Application.dll

```
public class SourceReferenceResult
```

## Inheritance

object ← SourceReferenceResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Description

```
public string? Description { get; set; }
```

#### Property Value

[string](#) ↗

### FileExtension

```
public string? FileExtension { get; set; }
```

#### Property Value

[string](#) ↗

### FilePath

```
public string? FilePath { get; set; }
```

Property Value

[string](#)

## HasAttachment

```
public bool HasAttachment { get; set; }
```

Property Value

[bool](#)

## Metadata

```
public List<MetadataResult>? Metadata { get; set; }
```

Property Value

[List](#)<[MetadataResult](#)>

## References

```
public List<AddSourceReferenceCommand>? References { get; set; }
```

Property Value

[List](#)<[AddSourceReferenceCommand](#)>

## Tags

```
public List<Guid>? Tags { get; set; }
```

Property Value

[List](#) <[Guid](#)>

Title

```
public string Title { get; set; }
```

Property Value

[string](#)

Type

```
public SourceType Type { get; set; }
```

Property Value

[SourceType](#)

# Namespace Application.Sources.Commands. AddTagToSource

## Classes

[AddTagsToSourceCommand](#)

[AddTagsToSourceCommandHandler](#)

# Class AddTagsToSourceCommand

Namespace: [Application.Sources.Commands.AddTagToSource](#)

Assembly: Application.dll

```
public class AddTagsToSourceCommand : IRequest<Result>, IBaseRequest
```

## Inheritance

object ← AddTagsToSourceCommand

## Implements

IRequest<[Result](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### SourceId

```
public Guid SourceId { get; set; }
```

#### Property Value

[Guid](#) ↗

### TagId

```
public Guid TagId { get; set; }
```

#### Property Value

[Guid](#) ↗

# Class AddTagsToSourceCommandHandler

Namespace: [Application.Sources.Commands.AddTagToSource](#)

Assembly: Application.dll

```
public class AddTagsToSourceCommandHandler : IRequestHandler<AddTagsToSourceCommand, Result>
```

## Inheritance

object ← AddTagsToSourceCommandHandler

## Implements

IRequestHandler<[AddTagsToSourceCommand](#), [Result](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### AddTagsToSourceCommandHandler(IServiceManager)

```
public AddTagsToSourceCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(AddTagsToSourceCommand, CancellationToken)

Handles a request

```
public Task<Result> Handle(AddTagsToSourceCommand request, CancellationToken cancellationToken)
```

## Parameters

request [AddTagsToSourceCommand](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

Returns

[Task](#) <[Result](#)>

Response from the request

# Namespace Application.Sources.Commands. RemoveBulkTagFromSource

## Classes

[RemoveBulkTagFromSourceCommand](#)

# Class RemoveBulkTagFromSourceCommand

Namespace: [Application.Sources.Commands.RemoveBulkTagFromSource](#)

Assembly: Application.dll

```
public class RemoveBulkTagFromSourceCommand : IRequest<Result<PartialsSuccessResult>>,  
IBaseRequest
```

## Inheritance

object ← RemoveBulkTagFromSourceCommand

## Implements

IRequest<[Result<PartialsSuccessResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### SourceId

```
[JsonIgnore]  
public Guid SourceId { get; set; }
```

### Property Value

[Guid](#)

### TagIds

```
public IEnumerable<Guid> TagIds { get; set; }
```

### Property Value

[IEnumerable](#) <[Guid](#)>

# Namespace Application.Sources.Commands. RemoveTagFromSource

## Classes

[RemoveTagFromSourceCommand](#)

[RemoveTagFromSourceCommandHandler](#)

# Class RemoveTagFromSourceCommand

Namespace: [Application.Sources.Commands.RemoveTagFromSource](#)

Assembly: Application.dll

```
public class RemoveTagFromSourceCommand : IRequest<Result>, IBaseRequest
```

## Inheritance

object ← RemoveTagFromSourceCommand

## Implements

IRequest<[Result](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### SourceId

```
public Guid SourceId { get; set; }
```

#### Property Value

[Guid](#) ↗

### TagId

```
public Guid TagId { get; set; }
```

#### Property Value

[Guid](#) ↗

# Class RemoveTagFromSourceCommandHandler

Namespace: [Application.Sources.Commands.RemoveTagFromSource](#)

Assembly: Application.dll

```
public class RemoveTagFromSourceCommandHandler : IRequestHandler<RemoveTagFromSourceCommand, Result>
```

## Inheritance

object ← RemoveTagFromSourceCommandHandler

## Implements

IRequestHandler<[RemoveTagFromSourceCommand](#), [Result](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### RemoveTagFromSourceCommandHandler(IServiceManager)

```
public RemoveTagFromSourceCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(RemoveTagFromSourceCommand, CancellationToken)

Handles a request

```
public Task<Result> Handle(RemoveTagFromSourceCommand request, CancellationToken cancellationToken)
```

## Parameters

request [RemoveTagFromSourceCommand](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)>

Response from the request

# Namespace Application.Sources.Queries.GetAllSources

## Classes

[GetAllSourceQuery](#)

[GetAllSourceQueryHandler](#)

[GetAllSourceQueryResult](#)

# Class GetAllSourceQuery

Namespace: [Application.Sources.Queries.GetAllSources](#)

Assembly: Application.dll

```
public class GetAllSourceQuery : PageRequest,  
 IRequest<Result<PageResponse<GetAllSourceQueryResult>>>, IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← GetAllSourceQuery

## Implements

IRequest<[Result<PageResponse<GetAllSourceQueryResult>>](#)>, IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Tags

```
public Guid? Tags { get; set; }
```

## Property Value

[Guid](#)?

## Title

```
public string? Title { get; set; }
```

## Property Value

[string](#) ↗

## TypeId

```
public SourceType? TypeId { get; set; }
```

Property Value

[SourceType?](#)

# Class GetAllSourceQueryHandler

Namespace: [Application.Sources.Questions.GetAllSources](#)

Assembly: Application.dll

```
public class GetAllSourceQueryHandler : IRequestHandler<GetAllSourceQuery,
Result<PageResponse< GetAllSourceQueryResult>>>
```

## Inheritance

object ← GetAllSourceQueryHandler

## Implements

IRequestHandler<[GetAllSourceQuery](#), [Result<PageResponse< GetAllSourceQueryResult>>](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### GetAllSourceQueryHandler(IRepositoryManager, IMapper)

```
public GetAllSourceQueryHandler(IRepositoryManager repositoryManager, IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

mapper IMapper

## Methods

### Handle(GetAllSourceQuery, CancellationToken)

Handles a request

```
public Task<Result<PageResponse< GetAllSourceQueryResult>>> Handle(GetAllSourceQuery request,
```

```
CancellationToken cancellationToken)
```

## Parameters

request [GetAllSourceQuery](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[PageResponse](#)<[GetAllSourceQueryResult](#)>>>

Response from the request

# Class GetAllSourceQueryResult

Namespace: [Application.Sources.Questions.GetAllSources](#)

Assembly: Application.dll

```
[SwaggerSchema(null, Description = "Success response to get all the sources")]
public class GetAllSourceQueryResult
```

## Inheritance

object ← GetAllSourceQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Properties

## Description

```
[SwaggerSchema(null, Description = "short description for a source", Nullable = true, Title
= "Source description")]
[SwaggerExample("Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo
ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient
montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium
quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec,
vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam
dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum
semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat
vitae, eleifend ac, enim. Aliquam lorem ante, dapibus in, viverra quis, feugiat a, tellus.
Phasellus viverra nulla ut metus varius laoreet. Quisque rutrum. Aenean imperdiet. Etiam
ultricies nisi vel augue. Curabitur ullamcorper ultricies nisi. Nam eget dui. Etiam rhoncus.
Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet
adipiscing sem neque sed ipsum. Nam quam nunc, blandit vel, luctus pulvinar, hendrerit id,
lorem. Maecenas nec odio et ante tincidunt tempus. Donec vitae sapien ut libero venenatis
faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt. Duis leo. Sed
fringilla mauris sit amet nibh. Donec sodales sagittis magna. Sed consequat, leo eget
bibendum sodales, augue velit cursus nunc")]
[StringLength(2500, MinimumLength = 1)]
public string? Description { get; set; }
```

## Property Value

[string](#) ↗

## Id

```
[SwaggerSchema(null, Description = "An UUID unique id for source", ReadOnly = true, Format =
"UUID", Nullable = false, Title = "Source Unique Identifier")]
[SwaggerExample("01979c44-b446-7380-ad8b-df6b2a2be0bc")]
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

## Tags

```
[SwaggerSchema(null, Description = "An collection of tags", Nullable = true, Title =
"Source Tags")]
[Range(1, 50)]
public List<TagResult>? Tags { get; set; }
```

## Property Value

[List](#) ↗ <[TagResult](#)>

## Title

```
[SwaggerSchema(null, Description = "A source Title", Nullable = false, Title =
"Source title")]
[SwaggerExample("Intro programming")]
[StringLength(100, MinimumLength = 1)]
public string Title { get; set; }
```

## Property Value

[string](#) ↗

## Type

```
[SwaggerSchema(null, Description = "An type of a source", ReadOnly = true, Format = "UUID",
Nullable = false, Title = "Source Unique Identifier")]
[SwaggerExample(SourceType.Article)]
[StringLength(50, MinimumLength = 1)]
public SourceType Type { get; set; }
```

## Property Value

[SourceType](#)

# Namespace Application.Sources.Queries.GetSourceById

## Classes

[GetSourceByIdQuery](#)

[GetSourceByIdQueryHandler](#)

[GetSourceByIdQueryResult](#)

# Class GetSourceByIdQuery

Namespace: [Application.Sources.Queries.GetSourceById](#)

Assembly: Application.dll

```
public class GetSourceByIdQuery : IRequest<Result<GetSourceByIdQueryResult>>, IBaseRequest
```

## Inheritance

object ← GetSourceByIdQuery

## Implements

IRequest<[Result<GetSourceByIdQueryResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Properties

## Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

# Class GetSourceByIdQueryHandler

Namespace: [Application.Sources.Queries.GetSourceById](#)

Assembly: Application.dll

```
public class GetSourceByIdQueryHandler : IRequestHandler<GetSourceByIdQuery, Result<GetSourceByIdQueryResult>>
```

## Inheritance

object ← GetSourceByIdQueryHandler

## Implements

IRequestHandler<[GetSourceByIdQuery](#), [Result<GetSourceByIdQueryResult>](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### GetSourceByIdQueryHandler(IRepositoryManager, IMapper)

```
public GetSourceByIdQueryHandler(IRepositoryManager repositoryManager, IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

mapper IMapper

## Methods

### Handle(GetSourceByIdQuery, CancellationToken)

Handles a request

```
public Task<Result<GetSourceByIdQueryResult>> Handle(GetSourceByIdQuery request,
```

```
CancellationToken cancellationToken)
```

## Parameters

request [GetSourceByIdQuery](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[GetSourceByIdQueryResult](#)>>

Response from the request

# Class GetSourceByIdQueryResult

Namespace: [Application.Sources.Queries.GetSourceById](#)

Assembly: Application.dll

```
public class GetSourceByIdQueryResult
```

## Inheritance

object ← GetSourceByIdQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### ArchivedTags

```
public IEnumerable<TagResult>? ArchivedTags { get; set; }
```

#### Property Value

[IEnumerable](#) <[TagResult](#)>

### CreationDate

```
public DateTimeOffset CreationDate { get; set; }
```

#### Property Value

[DateTimeOffset](#)

### Description

```
public string? Description { get; set; }
```

Property Value

[string](#) ↗

## FileExtension

```
public string? FileExtension { get; set; }
```

Property Value

[string](#) ↗

## FilePath

```
public string? FilePath { get; set; }
```

Property Value

[string](#) ↗

## HasAttachment

```
public bool HasAttachment { get; set; }
```

Property Value

[bool](#) ↗

## Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

IsArchived

```
public bool IsArchived { get; set; }
```

Property Value

[bool](#)

LastArchiveDate

```
public DateTimeOffset? LastArchiveDate { get; set; }
```

Property Value

[DateTimeOffset](#)?

LastModifiedDate

```
public DateTimeOffset? LastModifiedDate { get; set; }
```

Property Value

[DateTimeOffset](#)?

Metadata

```
public IEnumerable<MetadataResult>? Metadata { get; set; }
```

Property Value

[IEnumerable](#) <[MetadataResult](#)>

Tags

```
public IEnumerable<TagResult>? Tags { get; set; }
```

Property Value

[IEnumerable](#) <[TagResult](#)>

Title

```
public string Title { get; set; }
```

Property Value

[string](#)

Type

```
public SourceType Type { get; set; }
```

Property Value

[SourceType](#)

VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int ↗](#)

# Namespace Application.Sources.Queries.GetSourceReferences

## Classes

[GetSourceReferencesQuery](#)

[GetSourceReferencesQueryResult](#)

# Class GetSourceReferencesQuery

Namespace: [Application.Sources.Queries.GetSourceReferences](#)

Assembly: Application.dll

```
public class GetSourceReferencesQuery
```

## Inheritance

object ← GetSourceReferencesQuery

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Note

```
public string? Note { get; set; }
```

Property Value

[string](#) ↗

### SourceId

```
public int SourceId { get; set; }
```

Property Value

[int](#) ↗

# Class GetSourceReferencesQueryResult

Namespace: [Application.Sources.Queries.GetSourceReferences](#)

Assembly: Application.dll

```
public class GetSourceReferencesQueryResult
```

## Inheritance

object ← GetSourceReferencesQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Id

```
public int Id { get; set; }
```

Property Value

[int↗](#)

# Namespace Application.Tags.Commands.Add Tag

## Classes

[AddTagCommand](#)

[AddTagCommandHandler](#)

[AddTagCommandResult](#)

[AddTagCommandValidator](#)

# Class AddTagCommand

Namespace: [Application.Tags.Commands.AddTag](#)

Assembly: Application.dll

```
[SwaggerSchema(null, Description = "a command for creating a new tag", Nullable = false,
Required = new string[] { "name" })]
public class AddTagCommand : IRequest<Result<AddTagCommandResult>>, IBaseRequest
```

## Inheritance

object ← AddTagCommand

## Implements

IRequest<[Result<AddTagCommandResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Properties

## BackgroundColorCode

```
[SwaggerSchema(null, Description = "Hexcode color format should start with #", Nullable =
true, Title = "Tag Color")]
[SwaggerExample("#1756a7")]
public string? BackgroundColorCode { get; set; }
```

## Property Value

[string](#) ↗

## Name

```
[SwaggerSchema(null, Description = "the Value for tag", Nullable = false, Title =
"Tag Color")]
[SwaggerExample("Math-101")]
public string Name { get; set; }
```

Property Value

[string](#) ↗

## TextColorCode

```
[SwaggerSchema(null, Description = "Hexcode color format should start with #", Nullable =  
true, Title = "Tag Color")]  
[SwaggerExample("#1756a7")]  
public string? TextColorCode { get; set; }
```

Property Value

[string](#) ↗

# Class AddTagCommandHandler

Namespace: [Application.Tags.Commands.AddTag](#)

Assembly: Application.dll

```
public class AddTagCommandHandler : IRequestHandler<AddTagCommand,  
Result<AddTagCommandResult>>
```

## Inheritance

object ← AddTagCommandHandler

## Implements

IRequestHandler<[AddTagCommand](#), [Result<AddTagCommandResult>](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

AddTagCommandHandler(IServiceManager, IMapper)

```
public AddTagCommandHandler(IServiceManager serviceManager, IMapper mapper)
```

## Parameters

serviceManager [IServiceManager](#)

mapper IMapper

## Methods

Handle(AddTagCommand, CancellationToken)

Handles a request

```
public Task<Result<AddTagCommandResult>> Handle(AddTagCommand request,
```

```
CancellationToken cancellationToken)
```

## Parameters

request [AddTagCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[AddTagCommandResult](#)>>

Response from the request

# Class AddTagCommandResult

Namespace: [Application.Tags.Commands.AddTag](#)

Assembly: Application.dll

```
public class AddTagCommandResult
```

## Inheritance

object ← AddTagCommandResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### BackgroundColorCode

```
public string BackgroundColorCode { get; set; }
```

Property Value

[string](#) ↗

### BackgroundColorGroup

```
public ColorCategory BackgroundColorGroup { get; set; }
```

Property Value

[ColorCategory](#)

### CreationDate

```
public DateTimeOffset CreationDate { get; set; }
```

Property Value

[DateTimeOffset](#)

Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

Name

```
public string Name { get; set; }
```

Property Value

[string](#)

TextColorCode

```
public string TextColorCode { get; set; }
```

Property Value

[string](#)

TextColorGroup

```
public ColorCategory TextColorGroup { get; set; }
```

Property Value

[ColorCategory](#)

VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int](#)

# Class AddTagCommandValidator

Namespace: [Application.Tags.Commands.AddTag](#)

Assembly: Application.dll

```
public class AddTagCommandValidator : AbstractValidator<AddTagCommand>,
IValidator<AddTagCommand>, IValidator, IEnumerable<IValidationRule>, IEnumerable
```

## Inheritance

object ← AbstractValidator<[AddTagCommand](#)> ← AddTagCommandValidator

## Implements

IValidator<[AddTagCommand](#)>, IValidator, [IEnumerable](#) ↴ <IValidationRule>, [IEnumerable](#) ↴

## Inherited Members

AbstractValidator<AddTagCommand>.Validate(AddTagCommand) ,  
[AbstractValidator<AddTagCommand>.ValidateAsync\(AddTagCommand, CancellationToken\)](#) ↴ ,  
AbstractValidator<AddTagCommand>.Validate(ValidationContext<AddTagCommand>) ,  
[AbstractValidator<AddTagCommand>.ValidateAsync\(ValidationContext<AddTagCommand>, CancellationToken\)](#) ↴ ,  
AbstractValidator<AddTagCommand>.CreateDescriptor() ,  
[AbstractValidator<AddTagCommand>.RuleFor<TPROPERTY>\(Expression<Func<AddTagCommand, TPROPERTY>>\)](#) ↴ ,  
[AbstractValidator<AddTagCommand>.Transform<TPROPERTY, TTransformed>\(Expression<Func<AddTagCommand, TPROPERTY>>, Func<TPROPERTY, TTransformed>\)](#) ↴ ,  
[AbstractValidator<AddTagCommand>.Transform<TPROPERTY, TTransformed>\(Expression<Func<AddTagCommand, TPROPERTY>>, Func<AddTagCommand, TPROPERTY, TTransformed>\)](#) ↴ ,  
[AbstractValidator<AddTagCommand>.RuleForEach<TELEMENT>\(Expression<Func<AddTagCommand, IEnumerable<TELEMENT>>\)](#) ↴ ,  
[AbstractValidator<AddTagCommand>.TransformForEach<TELEMENT, TTransformed>\(Expression<Func<AddTagCommand, IEnumerable<TELEMENT>>, Func<TELEMENT, TTransformed>\)](#) ↴ ,  
[AbstractValidator<AddTagCommand>.TransformForEach<TELEMENT, TTransformed>\(Expression<Func<AddTagCommand, IEnumerable<TELEMENT>>, Func<AddTagCommand, TELEMENT, TTransformed>\)](#) ↴ ,  
[AbstractValidator<AddTagCommand>.RuleSet\(string, Action\)](#) ↴ ,  
[AbstractValidator<AddTagCommand>.When\(Func<AddTagCommand, bool>, Action\)](#) ↴ ,  
[AbstractValidator<AddTagCommand>.When\(Func<AddTagCommand, ValidationContext<AddTagCommand>, bool>, Action\)](#) ↴ ,

[AbstractValidator<AddTagCommand>.Unless\(Func<AddTagCommand, bool>, Action\)](#) ,  
[AbstractValidator<AddTagCommand>.Unless\(Func<AddTagCommand,](#)  
[ValidationContext<AddTagCommand>, bool>, Action\)](#) ,  
[AbstractValidator<AddTagCommand>.WhenAsync\(Func<AddTagCommand, CancellationToken,](#)  
[Task<bool>, Action\)](#) ,  
[AbstractValidator<AddTagCommand>.WhenAsync\(Func<AddTagCommand,](#)  
[ValidationContext<AddTagCommand>, CancellationToken, Task<bool>, Action\)](#) ,  
[AbstractValidator<AddTagCommand>.UnlessAsync\(Func<AddTagCommand, CancellationToken,](#)  
[Task<bool>, Action\)](#) ,  
[AbstractValidator<AddTagCommand>.UnlessAsync\(Func<AddTagCommand,](#)  
[ValidationContext<AddTagCommand>, CancellationToken, Task<bool>, Action\)](#) ,  
AbstractValidator<AddTagCommand>.Include(IValidator<AddTagCommand>) ,  
[AbstractValidator<AddTagCommand>.Include<TValidator>\(Func<AddTagCommand, TValidator>\)](#) ,  
AbstractValidator<AddTagCommand>.GetEnumerator() ,  
AbstractValidator<AddTagCommand>.EnsureInstanceNotNull(object) ,  
AbstractValidator<AddTagCommand>.PreValidate(ValidationContext<AddTagCommand>,  
ValidationResult) ,  
AbstractValidator<AddTagCommand>.RaiseValidationException(ValidationContext<AddTagCommand>,  
ValidationResult) ,  
AbstractValidator<AddTagCommand>.OnRuleAdded(IValidationRule<AddTagCommand>) ,  
AbstractValidator<AddTagCommand>.CascadeMode ,  
AbstractValidator<AddTagCommand>.ClassLevelCascadeMode ,  
AbstractValidator<AddTagCommand>.RuleLevelCascadeMode

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Constructors

## AddTagCommandValidator()

```
public AddTagCommandValidator()
```

# Namespace Application.Tags.Commands. ArchiveAllTags

## Classes

[ArchiveAllTagsCommand](#)

[ArchiveAllTagsCommandHandler](#)

# Class ArchiveAllTagsCommand

Namespace: [Application.Tags.Commands.ArchiveAllTags](#)

Assembly: Application.dll

```
public class ArchiveAllTagsCommand : IRequest<Result>, IBaseRequest
```

## Inheritance

object ← ArchiveAllTagsCommand

## Implements

IRequest<[Result](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Class ArchiveAllTagsCommandHandler

Namespace: [Application.Tags.Commands.ArchiveAllTags](#)

Assembly: Application.dll

```
public class ArchiveAllTagsCommandHandler : IRequestHandler<ArchiveAllTagsCommand, Result>
```

## Inheritance

object ← ArchiveAllTagsCommandHandler

## Implements

IRequestHandler<[ArchiveAllTagsCommand](#), [Result](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### ArchiveAllTagsCommandHandler(IServiceManager)

```
public ArchiveAllTagsCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(ArchiveAllTagsCommand, CancellationToken)

Handles a request

```
public Task<Result> Handle(ArchiveAllTagsCommand request, CancellationToken cancellationToken)
```

## Parameters

request [ArchiveAllTagsCommand](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

Returns

[Task](#) <[Result](#)>

Response from the request

# Namespace Application.Tags.Commands. ArchiveTag

## Classes

[ArchiveTagCommand](#)

[ArchiveTagCommandHandler](#)

# Class ArchiveTagCommand

Namespace: [Application.Tags.Commands.ArchiveTag](#)

Assembly: Application.dll

```
public class ArchiveTagCommand : IRequest<Result>, IBaseRequest
```

## Inheritance

object ← ArchiveTagCommand

## Implements

IRequest<[Result](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Properties

## Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

# Class ArchiveTagCommandHandler

Namespace: [Application.Tags.Commands.ArchiveTag](#)

Assembly: Application.dll

```
public class ArchiveTagCommandHandler : IRequestHandler<ArchiveTagCommand, Result>
```

## Inheritance

object ← ArchiveTagCommandHandler

## Implements

IRequestHandler<[ArchiveTagCommand](#), [Result](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### ArchiveTagCommandHandler(IServiceManager)

```
public ArchiveTagCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(ArchiveTagCommand, CancellationToken)

Handles a request

```
public Task<Result> Handle(ArchiveTagCommand request, CancellationToken cancellationToken)
```

## Parameters

request [ArchiveTagCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

Returns

[Task](#) <[Result](#)>

Response from the request

# Namespace Application.Tags.Commands.Delete Tag

## Classes

[DeleteTagCommand](#)

[DeleteTagCommandHandler](#)

# Class DeleteTagCommand

Namespace: [Application.Tags.Commands.DeleteTag](#)

Assembly: Application.dll

```
public class DeleteTagCommand : IRequest<Result>, IBaseRequest
```

## Inheritance

object ← DeleteTagCommand

## Implements

IRequest<[Result](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

# Class DeleteTagCommandHandler

Namespace: [Application.Tags.Commands.DeleteTag](#)

Assembly: Application.dll

```
public class DeleteTagCommandHandler : IRequestHandler<DeleteTagCommand, Result>
```

## Inheritance

object ← DeleteTagCommandHandler

## Implements

IRequestHandler<[DeleteTagCommand](#), [Result](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### DeleteTagCommandHandler(IServiceManager)

```
public DeleteTagCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(DeleteTagCommand, CancellationToken)

Handles a request

```
public Task<Result> Handle(DeleteTagCommand request, CancellationToken cancellationToken)
```

## Parameters

`request` [DeleteTagCommand](#)

The request

`cancellationToken` [CancellationToken](#)

Cancellation token

Returns

[Task](#) <[Result](#)>

Response from the request

# Namespace Application.Tags.Commands.UnArchiveTag

## Classes

[UnArchiveTagCommandHandler](#)

[UnarchiveTagCommand](#)

# Class UnArchiveTagCommandHandler

Namespace: [Application.Tags.Commands.UnArchiveTag](#)

Assembly: Application.dll

```
public class UnArchiveTagCommandHandler : IRequestHandler<UnarchiveTagCommand, Result>
```

## Inheritance

object ← UnArchiveTagCommandHandler

## Implements

IRequestHandler<[UnarchiveTagCommand](#), [Result](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### UnArchiveTagCommandHandler(IServiceManager)

```
public UnArchiveTagCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(UnarchiveTagCommand, CancellationToken)

Handles a request

```
public Task<Result> Handle(UnarchiveTagCommand request, CancellationToken cancellationToken)
```

## Parameters

request [UnarchiveTagCommand](#)

The request

cancellationToken [CancellationToken](#)

Cancellation token

Returns

[Task](#) <[Result](#)>

Response from the request

# Class UnarchiveTagCommand

Namespace: [Application.Tags.Commands.UnArchiveTag](#)

Assembly: Application.dll

```
public class UnarchiveTagCommand : IRequest<Result>, IBaseRequest
```

## Inheritance

object ← UnarchiveTagCommand

## Implements

IRequest<[Result](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Id

```
public Guid Id { get; set; }
```

## Property Value

[Guid](#) ↗

# Namespace Application.Tags.Commands. UpdateTag

## Classes

[UpdateTagCommand](#)

[UpdateTagCommandHandler](#)

[UpdateTagCommandValidator](#)

# Class UpdateTagCommand

Namespace: [Application.Tags.Commands.UpdateTag](#).

Assembly: Application.dll

```
public class UpdateTagCommand : IRequest<Result>, IBaseRequest
```

## Inheritance

object ← UpdateTagCommand

## Implements

IRequest<[Result](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Properties

## BackgroundColorCode

```
public string? BackgroundColorCode { get; set; }
```

### Property Value

[string](#) ↗

## Id

```
public Guid Id { get; set; }
```

### Property Value

[Guid](#) ↗

## Name

```
public string Name { get; set; }
```

## Property Value

[string](#) ↗

## TextColorCode

```
public string? TextColorCode { get; set; }
```

## Property Value

[string](#) ↗

# Class UpdateTagCommandHandler

Namespace: [Application.Tags.Commands.UpdateTag](#).

Assembly: Application.dll

```
public class UpdateTagCommandHandler : IRequestHandler<UpdateTagCommand, Result>
```

## Inheritance

object ← UpdateTagCommandHandler

## Implements

IRequestHandler<[UpdateTagCommand](#), [Result](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### UpdateTagCommandHandler(IServiceManager)

```
public UpdateTagCommandHandler(IServiceManager serviceManager)
```

## Parameters

serviceManager [IServiceManager](#)

## Methods

### Handle(UpdateTagCommand, CancellationToken)

Handles a request

```
public Task<Result> Handle(UpdateTagCommand request, CancellationToken cancellationToken)
```

## Parameters

request [UpdateTagCommand](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

Returns

[Task](#) <[Result](#)>

Response from the request

# Class UpdateTagCommandValidator

Namespace: [Application.Tags.Commands.UpdateTag](#).

Assembly: Application.dll

```
public class UpdateTagCommandValidator : AbstractValidator<UpdateTagCommand>,
IValidator<UpdateTagCommand>, IValidator, IEnumerable<IValidationRule>, IEnumerable
```

## Inheritance

object ← AbstractValidator<[UpdateTagCommand](#)> ← UpdateTagCommandValidator

## Implements

IValidator<[UpdateTagCommand](#)>, IValidator, [IEnumerable](#) ↴ <IValidationRule>, [IEnumerable](#) ↴

## Inherited Members

```
AbstractValidator<UpdateTagCommand>.Validate(UpdateTagCommand) ,
AbstractValidator<UpdateTagCommand>.ValidateAsync(UpdateTagCommand, CancellationToken) ↴ ,
AbstractValidator<UpdateTagCommand>.Validate(ValidationContext<UpdateTagCommand>) ,
AbstractValidator<UpdateTagCommand>.ValidateAsync(ValidationContext<UpdateTagCommand>,
CancellationToken) ↴ ,
AbstractValidator<UpdateTagCommand>.CreateDescriptor() ,
AbstractValidator<UpdateTagCommand>.RuleFor<TPROPERTY>(Expression<Func<UpdateTagCommand,
TPROPERTY>>) ↴ ,
AbstractValidator<UpdateTagCommand>.Transform<TPROPERTY, TTransformed>
(Expression<Func<UpdateTagCommand, TPROPERTY>>, Func<TPROPERTY, TTransformed>) ↴ ,
AbstractValidator<UpdateTagCommand>.Transform<TPROPERTY, TTransformed>
(Expression<Func<UpdateTagCommand, TPROPERTY>>, Func<UpdateTagCommand, TPROPERTY,
TTransformed>) ↴ ,
AbstractValidator<UpdateTagCommand>.RuleForEach<TELEMENT>
(Expression<Func<UpdateTagCommand, IEnumerable<TELEMENT>>>) ↴ ,
AbstractValidator<UpdateTagCommand>.TransformForEach<TELEMENT, TTransformed>
(Expression<Func<UpdateTagCommand, IEnumerable<TELEMENT>>>, Func<TELEMENT, TTransformed>)
, ↴ ,
AbstractValidator<UpdateTagCommand>.TransformForEach<TELEMENT, TTransformed>
(Expression<Func<UpdateTagCommand, IEnumerable<TELEMENT>>>, Func<UpdateTagCommand,
TELEMENT, TTransformed>) ↴ ,
AbstractValidator<UpdateTagCommand>.RuleSet(string, Action) ↴ ,
AbstractValidator<UpdateTagCommand>.When(Func<UpdateTagCommand, bool>, Action) ↴ ,
```

[AbstractValidator<UpdateTagCommand>.When\(Func<UpdateTagCommand, ValidationContext<UpdateTagCommand>, bool, Action\)](#) ,  
[AbstractValidator<UpdateTagCommand>.Unless\(Func<UpdateTagCommand, bool, Action\)](#) ,  
[AbstractValidator<UpdateTagCommand>.Unless\(Func<UpdateTagCommand, ValidationContext<UpdateTagCommand>, bool, Action\)](#) ,  
[AbstractValidator<UpdateTagCommand>.WhenAsync\(Func<UpdateTagCommand, CancellationToken, Task<bool>, Action\)](#) ,  
[AbstractValidator<UpdateTagCommand>.WhenAsync\(Func<UpdateTagCommand, ValidationContext<UpdateTagCommand>, CancellationToken, Task<bool>, Action\)](#) ,  
[AbstractValidator<UpdateTagCommand>.UnlessAsync\(Func<UpdateTagCommand, CancellationToken, Task<bool>, Action\)](#) ,  
[AbstractValidator<UpdateTagCommand>.UnlessAsync\(Func<UpdateTagCommand, ValidationContext<UpdateTagCommand>, CancellationToken, Task<bool>, Action\)](#) ,  
AbstractValidator<UpdateTagCommand>.Include(IValidator<UpdateTagCommand>) ,  
[AbstractValidator<UpdateTagCommand>.Include<TValidator>\(Func<UpdateTagCommand, TValidator>\)](#) ,  
AbstractValidator<UpdateTagCommand>.GetEnumerator() ,  
AbstractValidator<UpdateTagCommand>.EnsureInstanceNotNull(object) ,  
AbstractValidator<UpdateTagCommand>.PreValidate(ValidationContext<UpdateTagCommand>, ValidationResult) ,  
AbstractValidator<UpdateTagCommand>.RaiseValidationException(ValidationContext<UpdateTagCommand>, ValidationResult) ,  
AbstractValidator<UpdateTagCommand>.OnRuleAdded(IValidationRule<UpdateTagCommand>) ,  
AbstractValidator<UpdateTagCommand>.CascadeMode ,  
AbstractValidator<UpdateTagCommand>.ClassLevelCascadeMode ,  
AbstractValidator<UpdateTagCommand>.RuleLevelCascadeMode

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Constructors

## UpdateTagCommandValidator()

```
public UpdateTagCommandValidator()
```

# Namespace Application.Tags.Queries.AutoCompleteTags

## Classes

[AutoCompleteTagsQuery](#)

[AutoCompleteTagsQueryHandler](#)

[AutoCompleteTagsQueryResult](#)

# Class AutoCompleteTagsQuery

Namespace: [Application.Tags.Questions.AutoCompleteTags](#)

Assembly: Application.dll

```
public class AutoCompleteTagsQuery :  
    IRequest<Result<IEnumerable<AutoCompleteTagsQueryResult>>>, IBaseRequest
```

## Inheritance

object ← AutoCompleteTagsQuery

## Implements

IRequest<[Result<IEnumerable<AutoCompleteTagsQueryResult>>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Properties

## Name

```
public string Name { get; set; }
```

## Property Value

[string](#)

# Class AutoCompleteTagsQueryHandler

Namespace: [Application.Tags.Questions.AutoCompleteTags](#)

Assembly: Application.dll

```
public class AutoCompleteTagsQueryHandler : IRequestHandler<AutoCompleteTagsQuery, Result<IEnumerable<AutoCompleteTagsQueryResult>>>
```

## Inheritance

object ← AutoCompleteTagsQueryHandler

## Implements

IRequestHandler<[AutoCompleteTagsQuery](#), [Result<IEnumerable<AutoCompleteTagsQueryResult>>](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

AutoCompleteTagsQueryHandler(IRepositoryManager, IMapper)

```
public AutoCompleteTagsQueryHandler(IRepositoryManager repositoryManager, IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

mapper IMapper

## Methods

Handle(AutoCompleteTagsQuery, CancellationToken)

Handles a request

```
public Task<Result<IEnumerable<AutoCompleteTagsQueryResult>>> Handle(AutoCompleteTagsQuery
```

```
request, CancellationToken cancellationToken)
```

## Parameters

request [AutoCompleteTagsQuery](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) < [Result](#) < [IEnumerable](#) < [AutoCompleteTagsQueryResult](#) > >

Response from the request

# Class AutoCompleteTagsQueryResult

Namespace: [Application.Tags.Queries.AutoCompleteTags](#)

Assembly: Application.dll

```
public class AutoCompleteTagsQueryResult
```

## Inheritance

object ← AutoCompleteTagsQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

### Name

```
public string Name { get; set; }
```

Property Value

[string](#) ↗

# Namespace Application.Tags.Queries.Export Tags

## Classes

[ExportTagsQuery](#)

[ExportTagsQueryHandler](#)

[ExportTagsQueryResult](#)

# Class ExportTagsQuery

Namespace: [Application.Tags.Queries.ExportTags](#)

Assembly: Application.dll

```
public class ExportTagsQuery : IRequest<Result<ExportTagsQueryResult>>, IBaseRequest
```

## Inheritance

object ← ExportTagsQuery

## Implements

IRequest<[Result<ExportTagsQueryResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Format

```
public FileFormat Format { get; set; }
```

## Property Value

[FileFormat](#)

## IncludeArchived

```
public bool IncludeArchived { get; set; }
```

## Property Value

[bool](#) ↗

# Class ExportTagsQueryHandler

Namespace: [Application.Tags.Questions.ExportTags](#)

Assembly: Application.dll

```
public class ExportTagsQueryHandler : IRequestHandler<ExportTagsQuery,  
Result<ExportTagsQueryResult>>
```

## Inheritance

object ← ExportTagsQueryHandler

## Implements

IRequestHandler<[ExportTagsQuery](#), [Result<ExportTagsQueryResult>](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### ExportTagsQueryHandler(IRepositoryManager)

```
public ExportTagsQueryHandler(IRepositoryManager repositoryManager)
```

## Parameters

repositoryManager [IRepositoryManager](#)

## Methods

### Handle(ExportTagsQuery, CancellationToken)

Handles a request

```
public Task<Result<ExportTagsQueryResult>> Handle(ExportTagsQuery request,  
CancellationToken cancellationToken)
```

## Parameters

request [ExportTagsQuery](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[ExportTagsQueryResult](#)>>

Response from the request

# Class ExportTagsQueryResult

Namespace: [Application.Tags.Queries.ExportTags](#)

Assembly: Application.dll

```
public class ExportTagsQueryResult
```

## Inheritance

object ← ExportTagsQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### ContentType

```
public string ContentType { get; set; }
```

Property Value

[string](#) ↗

### FileContent

```
public byte[] FileContent { get; set; }
```

Property Value

[byte](#) ↗ []

### FileName

```
public string FileName { get; set; }
```

Property Value

[string](#) ↗

# Namespace Application.Tags.Queries.GetAllTags

## Classes

[GetAllTagsQueryValidator](#)

# Class GetAllTagsQueryValidator

Namespace: [Application.Tags.Questions.GetAllTags](#)

Assembly: Application.dll

```
public class GetAllTagsQueryValidator : AbstractValidator<GetAllTagsQuery>,
IValidator<GetAllTagsQuery>, IValidator, IEnumerable<IValidationRule>, IEnumerable
```

## Inheritance

object ← AbstractValidator<[GetAllTagsQuery](#)> ← GetAllTagsQueryValidator

## Implements

IValidator<[GetAllTagsQuery](#)>, IValidator, [IEnumerable](#) ↴ <IValidationRule>, [IEnumerable](#) ↴

## Inherited Members

AbstractValidator<GetAllTagsQuery>.Validate(GetAllTagsQuery) ,  
[AbstractValidator<GetAllTagsQuery>.ValidateAsync\(GetAllTagsQuery, CancellationToken\)](#) ↴ ,  
AbstractValidator<GetAllTagsQuery>.Validate(ValidationContext<GetAllTagsQuery>) ,  
[AbstractValidator<GetAllTagsQuery>.ValidateAsync\(ValidationContext<GetAllTagsQuery>, CancellationToken\)](#) ↴ ,  
AbstractValidator<GetAllTagsQuery>.CreateDescriptor() ,  
[AbstractValidator<GetAllTagsQuery>.RuleFor<TProperty>\(Expression<Func<GetAllTagsQuery, TProperty>>\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.Transform<TProperty, TTransformed>\(Expression<Func<GetAllTagsQuery, TProperty>>, Func<TProperty, TTransformed>\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.Transform<TProperty, TTransformed>\(Expression<Func<GetAllTagsQuery, TProperty>>, Func<GetAllTagsQuery, TProperty, TTransformed>\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.RuleForEach<TElement>\(Expression<Func<GetAllTagsQuery, IEnumerable<TElement>>\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.TransformForEach<TElement, TTransformed>\(Expression<Func<GetAllTagsQuery, IEnumerable<TElement>>, Func<TElement, TTransformed>\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.TransformForEach<TElement, TTransformed>\(Expression<Func<GetAllTagsQuery, IEnumerable<TElement>>, Func<GetAllTagsQuery, TElement, TTransformed>\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.RuleSet\(string, Action\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.When\(Func<GetAllTagsQuery, bool>, Action\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.When\(Func<GetAllTagsQuery, ValidationContext<GetAllTagsQuery>, bool>, Action\)](#) ↴ ,  
[AbstractValidator<GetAllTagsQuery>.Unless\(Func<GetAllTagsQuery, bool>, Action\)](#) ↴ ,

```
AbstractValidator<GetAllTagsQuery>.Unless(Func<GetAllTagsQuery,>
ValidationContext<GetAllTagsQuery>, bool>, Action) ,
AbstractValidator<GetAllTagsQuery>.WhenAsync(Func<GetAllTagsQuery, CancellationToken,>
Task<bool>, Action) ,
AbstractValidator<GetAllTagsQuery>.WhenAsync(Func<GetAllTagsQuery,>
ValidationContext<GetAllTagsQuery>, CancellationToken, Task<bool>, Action) ,
AbstractValidator<GetAllTagsQuery>.UnlessAsync(Func<GetAllTagsQuery, CancellationToken,>
Task<bool>, Action) ,
AbstractValidator<GetAllTagsQuery>.UnlessAsync(Func<GetAllTagsQuery,>
ValidationContext<GetAllTagsQuery>, CancellationToken, Task<bool>, Action) ,
AbstractValidator<GetAllTagsQuery>.Include(IValidator<GetAllTagsQuery>) ,
AbstractValidator<GetAllTagsQuery>.Include<TValidator>(Func<GetAllTagsQuery, TValidator>) ,
AbstractValidator<GetAllTagsQuery>.GetEnumerator() ,
AbstractValidator<GetAllTagsQuery>.EnsureInstanceNotNull(object) ,
AbstractValidator<GetAllTagsQuery>.PreValidate(ValidationContext<GetAllTagsQuery>, ValidationResult)
,
AbstractValidator<GetAllTagsQuery>.RaiseValidationException(ValidationContext<GetAllTagsQuery>,
ValidationResult) ,
AbstractValidator<GetAllTagsQuery>.OnRuleAdded(IValidationRule<GetAllTagsQuery>) ,
AbstractValidator<GetAllTagsQuery>.CascadeMode ,
AbstractValidator<GetAllTagsQuery>.ClassLevelCascadeMode ,
AbstractValidator<GetAllTagsQuery>.RuleLevelCascadeMode
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### GetAllTagsQueryValidator()

```
public GetAllTagsQueryValidator()
```

# Namespace Application.Tags.Queries.GetRelatedSources

## Classes

[GetRelatedSourcesQuery](#)

[GetRelatedSourcesQueryHandler](#)

[GetRelatedSourcesQueryResult](#)

# Class GetRelatedSourcesQuery

Namespace: [Application.Tags.Queries.GetRelatedSources](#)

Assembly: Application.dll

```
public class GetRelatedSourcesQuery : PageRequest,  
 IRequest<Result<PageResponse<GetRelatedSourcesQueryResult>>>, IBaseRequest
```

## Inheritance

object ← [PageRequest](#) ← GetRelatedSourcesQuery

## Implements

IRequest<[Result<PageResponse<GetRelatedSourcesQueryResult>>>](#), IBaseRequest

## Inherited Members

[PageRequest.PageNumber](#) , [PageRequest.PageSize](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## TagId

```
public Guid TagId { get; set; }
```

## Property Value

[Guid](#) ↗

## Title

```
public string? Title { get; set; }
```

## Property Value

[string](#) ↗

## TypeId

```
public SourceType? TypeId { get; set; }
```

Property Value

[SourceType?](#)

# Class GetRelatedSourcesQueryHandler

Namespace: [Application.Tags.Queries.GetRelatedSources](#)

Assembly: Application.dll

```
public class GetRelatedSourcesQueryHandler : IRequestHandler<GetRelatedSourcesQuery, Result<PageResponse<GetRelatedSourcesQueryResult>>>
```

## Inheritance

object ← GetRelatedSourcesQueryHandler

## Implements

IRequestHandler<[GetRelatedSourcesQuery](#), [Result<PageResponse<GetRelatedSourcesQueryResult>>](#)>>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### GetRelatedSourcesQueryHandler(IRepositoryManager, IMapper)

```
public GetRelatedSourcesQueryHandler(IRepositoryManager repositoryManager, IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

mapper IMapper

## Methods

### Handle(GetRelatedSourcesQuery, CancellationToken)

Handles a request

```
public Task<Result<PageResponse<GetRelatedSourcesQueryResult>>>
```

```
Handle(GetRelatedSourcesQuery request, CancellationToken cancellationToken)
```

## Parameters

**request** [GetRelatedSourcesQuery](#)

The request

**cancellationToken**  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) < [Result](#) < [PageResponse](#) < [GetRelatedSourcesQueryResult](#) >>>

Response from the request

# Class GetRelatedSourcesQueryResult

Namespace: [Application.Tags.Queries.GetRelatedSources](#)

Assembly: Application.dll

```
public class GetRelatedSourcesQueryResult
```

## Inheritance

object ← GetRelatedSourcesQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Id

```
[SwaggerSchema(null, Description = "An UUID unique id for source", ReadOnly = true, Format =  
"UUID", Nullable = false, Title = "Source Unique Identifier")]  
[SwaggerExample("01979c44-b446-7380-ad8b-df6b2a2be0bc")]  
public Guid Id { get; set; }
```

### Property Value

[Guid](#) ↗

### IsArchived

```
[SwaggerSchema(null, Description = "flag if the source is archived", Nullable = false, Title  
= "Source archived flag")]  
[SwaggerExample(false)]  
public bool IsArchived { get; set; }
```

### Property Value

[bool](#) ↗

## Title

```
[SwaggerSchema(null, Description = "A source Title", Nullable = false, Title =
"Source title")]
[SwaggerExample("Intro programming")]
[StringLength(100, MinimumLength = 1)]
public string Title { get; set; }
```

## Property Value

[string](#) ↗

## Type

```
[SwaggerSchema(null, Description = "An type of a source", ReadOnly = true, Format = "UUID",
Nullable = false, Title = "Source Unique Identifier")]
[SwaggerExample(SourceType.Article)]
[StringLength(50, MinimumLength = 1)]
public SourceType Type { get; set; }
```

## Property Value

[SourceType](#)

# Namespace Application.Tags.Queries.GetTagDetails

## Classes

[GetTagDetailsQuery](#)

[GetTagDetailsQueryHandler](#)

[GetTagDetailsQueryResult](#)

# Class GetTagDetailsQuery

Namespace: [Application.Tags.Queries.GetTagDetails](#)

Assembly: Application.dll

```
public class GetTagDetailsQuery : IRequest<Result<GetTagDetailsQueryResult>>, IBaseRequest
```

## Inheritance

object ← GetTagDetailsQuery

## Implements

IRequest<[Result<GetTagDetailsQueryResult>](#)>, IBaseRequest

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

# Class GetTagDetailsQueryHandler

Namespace: [Application.Tags.Questions.GetTagDetails](#)

Assembly: Application.dll

```
public class GetTagDetailsQueryHandler : IRequestHandler<GetTagDetailsQuery, Result<GetTagDetailsQueryResult>>
```

## Inheritance

object ← GetTagDetailsQueryHandler

## Implements

IRequestHandler<[GetTagDetailsQuery](#), [Result<GetTagDetailsQueryResult>](#)>

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### GetTagDetailsQueryHandler(IRepositoryManager, IMapper)

```
public GetTagDetailsQueryHandler(IRepositoryManager repositoryManager, IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

mapper IMapper

## Methods

### Handle(GetTagDetailsQuery, CancellationToken)

Handles a request

```
public Task<Result<GetTagDetailsQueryResult>> Handle(GetTagDetailsQuery request,
```

```
CancellationToken cancellationToken)
```

## Parameters

request [GetTagDetailsQuery](#)

The request

cancellationToken  [CancellationToken](#)

Cancellation token

## Returns

[Task](#) <[Result](#)<[GetTagDetailsQueryResult](#)>>

Response from the request

# Class GetTagDetailsQueryResult

Namespace: [Application.Tags.Queries.GetTagDetails](#)

Assembly: Application.dll

```
public class GetTagDetailsQueryResult
```

## Inheritance

object ← GetTagDetailsQueryResult

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### ColorGroup

```
public ColorCategory ColorGroup { get; set; }
```

Property Value

[ColorCategory](#)

### ColorHexCode

```
public string ColorHexCode { get; set; }
```

Property Value

[string](#) ↗

### CreationDate

```
public DateTimeOffset CreationDate { get; set; }
```

Property Value

[DateTimeOffset](#)

Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

IsArchived

```
public bool IsArchived { get; set; }
```

Property Value

[bool](#)

LastArchiveDate

```
public DateTimeOffset? LastArchiveDate { get; set; }
```

Property Value

[DateTimeOffset?](#)

LastModifiedDate

```
public DateTimeOffset? LastModifiedDate { get; set; }
```

Property Value

[DateTimeOffset](#)?

Name

```
public string Name { get; set; }
```

Property Value

[string](#)

VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int](#)

# Namespace Domain.Auditing

## Classes

[ApplicationLog](#)

[AuditTrail](#)

[EntityAudit](#)

Represents a base class for auditable entities, providing standard audit properties such as creation date, last modification timestamp, version tracking, and soft-archive support.

[ExceptionInfo](#)

[PropertyChange](#)

# Class ApplicationLog

Namespace: [Domain.Auditing](#)

Assembly: Domain.dll

```
public class ApplicationLog
```

## Inheritance

object ← ApplicationLog

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### Exception

```
public ExceptionInfo? Exception { get; set; }
```

Property Value

[ExceptionInfo](#)

### Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

### Level

```
public LogEventLevel Level { get; set; }
```

Property Value

LogEventLevel

## Message

```
public string Message { get; set; }
```

Property Value

[string](#)

## Properties

```
public IDictionary<string, string> Properties { get; set; }
```

Property Value

[IDictionary](#) <[string](#), [string](#)>

## Timestamp

```
public DateTimeOffset Timestamp { get; set; }
```

Property Value

[DateTimeOffset](#)

# Class AuditTrail

Namespace: [Domain.Auditing](#)

Assembly: Domain.dll

```
public class AuditTrail
```

## Inheritance

object ← AuditTrail

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

AuditTrail(EntityName, Guid, ActionType, ActionBy, int, string?)

```
public AuditTrail(EntityName entityName, Guid entityId, ActionType operation, ActionBy  
changedBy, int versionNumber, string? comment = null)
```

## Parameters

entityName [EntityName](#)

entityId [Guid](#)

operation [ActionType](#)

changedBy [ActionBy](#)

versionNumber [int](#)

comment [string](#)

## Properties

ChangedBy

```
public ActionBy ChangedBy { get; set; }
```

Property Value

[ActionBy](#)

## Changes

```
public List<PropertyChange> Changes { get; set; }
```

Property Value

[List](#) <[PropertyChange](#)>

## Comment

```
public string? Comment { get; set; }
```

Property Value

[string](#)

## EntityId

```
public Guid EntityId { get; set; }
```

Property Value

[Guid](#)

## EntityName

```
public EntityName EntityName { get; set; }
```

Property Value

[EntityName](#)

Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

Operation

```
public ActionType Operation { get; set; }
```

Property Value

[ActionType](#)

Timestamp

```
public DateTimeOffset Timestamp { get; set; }
```

Property Value

[DateTimeOffset](#)

VersionNumber

```
public int VersionNumber { get; set; }
```

Property Value

[int ↗](#)

## Methods

GetChanges<T>(T, bool)

```
public void GetChanges<T>(T obj, bool isDelete)
```

Parameters

**obj** T

**isDelete** [bool ↗](#)

Type Parameters

T

GetChanges<T>(T, T)

```
public void GetChanges<T>(T oldObj, T newObj)
```

Parameters

**oldObj** T

**newObj** T

Type Parameters

T

# Class EntityAudit

Namespace: [Domain.Auditing](#)

Assembly: Domain.dll

Represents a base class for auditable entities, providing standard audit properties such as creation date, last modification timestamp, version tracking, and soft-archive support.

```
public class EntityAudit
```

## Inheritance

object ← EntityAudit

## Derived

[Category](#), [Tag](#), [Question](#), [Source](#), [SourceReference](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Properties

## CreationDate

Gets the timestamp when the entity was created.

```
public DateTimeOffset CreationDate { get; protected set; }
```

## Property Value

[DateTimeOffset](#)

## Id

Gets or sets the unique identifier of the entity.

```
public Guid Id { get; set; }
```

## PropertyValue

[Guid ↗](#)

## IsArchived

Gets a value indicating whether the entity is archived (soft-deleted).

```
public bool IsArchived { get; protected set; }
```

## PropertyValue

[bool ↗](#)

## LastArchiveDate

Gets the timestamp when the entity was archived.

```
public DateTimeOffset? LastArchiveDate { get; protected set; }
```

## PropertyValue

[DateTimeOffset ↗?](#)

## LastModifiedDate

Gets the timestamp of the last modification made to the entity.

```
public DateTimeOffset? LastModifiedDate { get; protected set; }
```

## PropertyValue

[DateTimeOffset ↗?](#)

## VersionNumber

Gets the current version number of the entity. Incremented with each update.

```
public int VersionNumber { get; protected set; }
```

Property Value

[int](#)

## Methods

### Archive()

Archives the entity by setting [IsArchived](#) to [true](#) and updating the [LastArchiveDate](#). Also updates the modification timestamp and version.

```
protected void Archive()
```

### Created()

Sets the entity as newly created. Initializes [CreationDate](#), sets [VersionNumber](#) to 1, and ensures the entity is not archived.

```
protected void Created()
```

### UnArchive()

Unarchives the entity by setting [IsArchived](#) to [false](#) and updating the modification state.

```
protected void UnArchive()
```

### Updated()

Updates the [LastModifiedDate](#) to the current timestamp and increments the [VersionNumber](#). Should be called whenever the entity is modified.

```
protected void Updated()
```

# Class ExceptionInfo

Namespace: [Domain.Auditing](#)

Assembly: Domain.dll

```
public class ExceptionInfo
```

## Inheritance

object ← ExceptionInfo

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### InnerException

```
public ExceptionInfo? InnerException { get; set; }
```

#### Property Value

[ExceptionInfo](#)

### Message

```
public string Message { get; set; }
```

#### Property Value

[string](#) ↗

### StackTrace

```
public string? StackTrace { get; set; }
```

Property Value

[string](#)

Type

```
public string Type { get; set; }
```

Property Value

[string](#)

## Methods

FromException(Exception)

```
public static ExceptionInfo FromException(Exception ex)
```

Parameters

ex [Exception](#)

Returns

[ExceptionInfo](#)

# Class PropertyChange

Namespace: [Domain.Auditing](#)

Assembly: Domain.dll

```
public class PropertyChange
```

## Inheritance

object ← PropertyChange

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### NewValue

```
public object? NewValue { get; set; }
```

Property Value

object

### OldValue

```
public object? OldValue { get; set; }
```

Property Value

object

### PropertyName

```
public string PropertyName { get; set; }
```

PropertyValue

[string](#) ↗

PropertyType

```
public string PropertyType { get; set; }
```

PropertyValue

[string](#) ↗

# Namespace Domain.Constants

## Classes

[ColorsConsts](#)

[EnumValues](#)

[ErrorMessage](#)

[MaxLength](#)

[QuestionDifficultyMatrixes](#)

[RegexPattern](#)

# Class ColorsConsts

Namespace: [Domain.Constants](#)

Assembly: Domain.dll

```
public class ColorsConsts
```

## Inheritance

object ← ColorsConsts

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Fields

### White

```
public const string White = "#FFFFFF"
```

#### Field Value

[string](#) ↗

# Class EnumValues

Namespace: [Domain.Constants](#)

Assembly: Domain.dll

```
public static class EnumValues
```

## Inheritance

object ← EnumValues

## Fields

### LongAnswer

```
public const string LongAnswer = "long answer question"
```

Field Value

[string](#) ↗

### MultipleChoise

```
public const string MultipleChoise = "multiple choice question"
```

Field Value

[string](#) ↗

### Reordering

```
public const string Reordering = "reordering question"
```

Field Value

[string](#) ↗

## ShortAnswer

```
public const string ShortAnswer = "short answer question"
```

Field Value

[string](#) ↗

## TrueAndFalse

```
public const string TrueAndFalse = "true & false question"
```

Field Value

[string](#) ↗

# Class ErrorMessage

Namespace: [Domain.Constants](#)

Assembly: Domain.dll

```
public class ErrorMessage
```

## Inheritance

object ← ErrorMessage

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Fields

### ENTITY\_IS\_ARCHIVED

```
public const string ENTITY_IS_ARCHIVED = "ENTITY_IS_ARCHIVED"
```

Field Value

[string](#) ↗

### NOT\_FOUND\_ENTITY

```
public const string NOT_FOUND_ENTITY = "NOT_FOUND_ENTITY"
```

Field Value

[string](#) ↗

### TAG\_WITH\_SAME\_NAME\_EXISTS

```
public const string TAG_WITH_SAME_NAME_EXISTS = "TAG_WITH_SAME_NAME_EXISTS"
```

Field Value

[string](#) ↗

# Class MaxLength

Namespace: [Domain.Constants](#)

Assembly: Domain.dll

```
public static class MaxLength
```

## Inheritance

object ← MaxLength

## Fields

### PAGESIZE\_MAX\_VALUE

```
public const int PAGESIZE_MAX_VALUE = 250
```

Field Value

[int↗](#)

### QUESTION\_TEXT\_MAX\_LENGTH

```
public const int QUESTION_TEXT_MAX_LENGTH = 1000
```

Field Value

[int↗](#)

### TAG\_COLOR\_CODE\_MAX\_LENGTH

```
public const int TAG_COLOR_CODE_MAX_LENGTH = 7
```

Field Value

[int](#) ↗

## TAG\_NAME\_MAX\_LENGTH

```
public const int TAG_NAME_MAX_LENGTH = 100
```

Field Value

[int](#) ↗

# Class QuestionDifficultyMatrixes

Namespace: [Domain.Constants](#)

Assembly: Domain.dll

```
public static class QuestionDifficultyMatrixes
```

## Inheritance

object ← QuestionDifficultyMatrixes

## Fields

### Average

```
public const short Average = 45
```

Field Value

[short](#) ↗

### Basic

```
public const short Basic = 0
```

Field Value

[short](#) ↗

### Challenging

```
public const short Challenging = 60
```

Field Value

[short ↴](#)

## Extreme

```
public const short Extreme = 90
```

Field Value

[short ↴](#)

## Manageable

```
public const short Manageable = 30
```

Field Value

[short ↴](#)

## Simple

```
public const short Simple = 15
```

Field Value

[short ↴](#)

## Tough

```
public const short Tough = 75
```

Field Value

[short ↴](#)

# Class RegexPattern

Namespace: [Domain.Constants](#)

Assembly: Domain.dll

```
public class RegexPattern
```

## Inheritance

object ← RegexPattern

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Fields

### MatchHexCode

```
public const string MatchHexCode = "^(?:[0-9a-fA-F]{3}|[0-9a-fA-F]{6})$"
```

#### Field Value

[string](#) ↗

# Namespace Domain.Entities.EntityLookup

## Classes

[Category](#)

[Language](#)

translate

[Tag](#)

Represents a categorized tag entity with color properties and archiving support.

# Class Category

Namespace: [Domain.Entities.EntityLookup](#)

Assembly: Domain.dll

```
public class Category : EntityAudit
```

## Inheritance

object ← [EntityAudit](#) ← Category

## Inherited Members

[EntityAudit.LastModifiedDate](#) , [EntityAudit.LastArchiveDate](#) , [EntityAudit.CreationDate](#) ,  
[EntityAudit.IsArchived](#) , [EntityAudit.VersionNumber](#) , [EntityAudit.Created\(\)](#) , [EntityAudit.Updated\(\)](#) ,  
[EntityAudit.Archive\(\)](#) , [EntityAudit.UnArchive\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### Category(string, string?)

```
public Category(string name, string? description)
```

#### Parameters

name [string](#) ↗

description [string](#) ↗

### Category(string, string?, Guid, int)

```
public Category(string name, string? description, Guid parentId, int level)
```

#### Parameters

`name` [string](#)

`description` [string](#)

`parentId` [Guid](#)

`level` [int](#)

## Properties

### Description

`public string? Description { get; }`

### Property Value

[string](#)

### HasChildren

`public bool HasChildren { get; }`

### Property Value

[bool](#)

### Id

`public Guid Id { get; }`

### Property Value

[Guid](#)

### IsRoot

```
[BsonIgnore]  
public bool IsRoot { get; }
```

Property Value

[bool](#)

Level

```
public int Level { get; }
```

Property Value

[int](#)

Name

```
public string Name { get; }
```

Property Value

[string](#)

ParentId

```
public Guid? ParentId { get; }
```

Property Value

[Guid](#)?

Methods

## EnableChildrenFlag()

```
public void EnableChildrenFlag()
```

## UpdateCategory(string, string?)

```
public void UpdateCategory(string name, string? description)
```

### Parameters

name [string](#)

description [string](#)

# Class Language

Namespace: [Domain.Entities.EntityLookup](#)

Assembly: Domain.dll

translate

```
public class Language
```

Inheritance

object ← Language

Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

Language(string, string)

```
public Language(string code, string displayName)
```

Parameters

code [string](#)

displayName [string](#)

## Properties

Code

```
public string Code { get; }
```

Property Value

[string](#)

## DisplayName

```
public string DisplayName { get; }
```

### Property Value

[string](#) ↗

## Id

```
public Guid Id { get; }
```

### Property Value

[Guid](#) ↗

# Class Tag

Namespace: [Domain.Entities.EntityLookup](#)

Assembly: Domain.dll

Represents a categorized tag entity with color properties and archiving support.

```
public class Tag : EntityAudit
```

## Inheritance

object ← [EntityAudit](#) ← Tag

## Inherited Members

[EntityAudit.LastModifiedDate](#) , [EntityAudit.LastArchiveDate](#) , [EntityAudit.CreationDate](#) ,  
[EntityAudit.IsArchived](#) , [EntityAudit.VersionNumber](#) , [EntityAudit.Created\(\)](#) , [EntityAudit.Updated\(\)](#) ,  
[EntityAudit.Archive\(\)](#) , [EntityAudit.UnArchive\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Remarks

The [Tag](#) class supports creation, updating, and soft-archiving functionality. It enforces domain rules to ensure that all tags have valid names and color codes in hexadecimal format.

## Constructors

### Tag(string, string, string)

Initializes a new instance of the [Tag](#) class with the specified name and color codes.

```
public Tag(string name, string backgroundColorHexCode, string textColorCode = null)
```

## Parameters

**name** [string](#) ↗

The name of the tag. Must be non-empty and non-null.

## `backgroundColorHexCode` [string](#)

The background color of the tag in hexadecimal format (e.g., `#FFFFFF`). Must be a valid 6-digit hex code with a leading hash.

## `textColorCode` [string](#)

The optional text color of the tag in hexadecimal format (e.g., `#000000`). If not provided, the default color [White](#) will be used.

## Exceptions

### [ArgumentNullException](#)

Thrown if `name`, `backgroundColorHexCode`, or resolved `textColorCode` is null or empty.

### [ArgumentException](#)

Thrown if `backgroundColorHexCode` or `textColorCode` are not valid hex color codes (must match format `#RRGGBB`).

# Properties

## BackgroundColorCode

Gets the background color code of the tag in hexadecimal format.

```
public string BackgroundColorCode { get; }
```

## Property Value

[string](#)

## BackgroundColorGroup

Gets the color category group derived from the background color.

```
public ColorCategory BackgroundColorGroup { get; }
```

## Property Value

[ColorCategory](#)

## Id

Gets the unique identifier of the tag.

```
public Guid Id { get; }
```

## Property Value

[Guid](#)

## Name

Gets the name of the tag.

```
public string Name { get; }
```

## Property Value

[string](#)

## TextColorCode

Gets the text color code of the tag in hexadecimal format.

```
public string TextColorCode { get; }
```

## Property Value

[string](#)

## TextColorGroup

Gets the color category group derived from the text color.

```
public ColorCategory TextColorGroup { get; }
```

Property Value

[ColorCategory](#)

## Methods

### ArchiveTag()

Archives the tag by marking it as no longer active.

```
public void ArchiveTag()
```

Remarks

This method sets [IsArchived](#) to [true](#) and records the archive timestamp.

### UnArchiveTag()

Restores the tag from an archived state.

```
public void UnArchiveTag()
```

Remarks

This method sets [IsArchived](#) to [false](#) and updates modification metadata.

### UpdateTag(string, string, string)

Updates the current [Tag](#) with a new name, background color, and text color.

```
public void UpdateTag(string name, string backgroundColorHexCode, string textColorCode)
```

## Parameters

### `name` [string](#)

The new name of the tag. Must not be null or empty.

### `backgroundColorHexCode` [string](#)

The new background color in hexadecimal format (e.g., `#FF5733`). Must be a valid 6-digit hex code with a leading hash.

### `textColorCode` [string](#)

The new text color in hexadecimal format (e.g., `#000000`). Must be a valid 6-digit hex code with a leading hash.

## Exceptions

### [ArgumentNullException](#)

Thrown if `name`, `backgroundColorHexCode`, or `textColorCode` is null or empty.

### [ArgumentException](#)

Thrown if `backgroundColorHexCode` or `textColorCode` are not valid hex color codes (must match format `#RRGGBB`).

# Namespace Domain.Entities.Examiner

## Classes

[LongAnswerQuestion](#)

[MultipleChoiceQuestionOption](#)

[MultipleChoiseQuestion](#)

[Question](#)

[QuestionAttachments](#)

[ReorderingQuestion](#)

[ShortAnswerQuestion](#)

[TrueFalseQuestion](#)

# Class LongAnswerQuestion

Namespace: [Domain.Entities.Examiner](#)

Assembly: Domain.dll

```
public class LongAnswerQuestion
```

## Inheritance

object ← LongAnswerQuestion

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

LongAnswerQuestion(int?, int?, string, string?)

```
public LongAnswerQuestion(int? maximinWords, int? minimanWords, string answer,  
string? feedback)
```

## Parameters

maximinWords [int](#)?

minimanWords [int](#)?

answer [string](#)

feedback [string](#)

## Properties

### Answer

```
public string Answer { get; set; }
```

Property Value

[string](#) ↗

Feedback

```
public string? Feedback { get; set; }
```

Property Value

[string](#) ↗

Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#) ↗

MaximinWords

```
public int? MaximinWords { get; set; }
```

Property Value

[int](#) ↗?

MinimanWords

```
public int? MinimanWords { get; set; }
```

Property Value

int?

# Class MultipleChoiceQuestionOption

Namespace: [Domain.Entities.Examiner](#)

Assembly: Domain.dll

```
public class MultipleChoiceQuestionOption
```

## Inheritance

object ← MultipleChoiceQuestionOption

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### MultipleChoiceQuestionOption(string, bool, float, string?)

```
public MultipleChoiceQuestionOption(string optionText, bool isCorrect, float weight, string?  
feedback = null)
```

## Parameters

optionText [string](#)

isCorrect [bool](#)

weight [float](#)

feedback [string](#)

## Properties

### FeedBack

```
public string? FeedBack { get; }
```

PropertyValue

[string](#) ↗

Id

```
public Guid Id { get; }
```

PropertyValue

[Guid](#) ↗

IsCorrect

```
public bool IsCorrect { get; }
```

PropertyValue

[bool](#) ↗

OptionText

```
public string OptionText { get; }
```

PropertyValue

[string](#) ↗

Weight

```
public float Weight { get; }
```

PropertyValue

float ↗

# Class MultipleChoiseQuestion

Namespace: [Domain.Entities.Examiner](#)

Assembly: Domain.dll

```
public class MultipleChoiseQuestion
```

## Inheritance

object ← MultipleChoiseQuestion

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

MultipleChoiseQuestion(List<MultipleChoiceQuestionOption>)

```
public MultipleChoiseQuestion(List<MultipleChoiceQuestionOption> options)
```

## Parameters

options [List](#)<[MultipleChoiceQuestionOption](#)>

## Properties

### Options

```
public List<MultipleChoiceQuestionOption> Options { get; set; }
```

## Property Value

[List](#)<[MultipleChoiceQuestionOption](#)>

## Type

```
public MultipleChoiseQuestionType Type { get; set; }
```

Property Value

[MultipleChoiseQuestionType](#)

## Methods

CreateOption(string, bool, float, string?)

```
public static MultipleChoiceQuestionOption CreateOption(string optionText, bool isCorrect,  
float weight, string? feedback)
```

Parameters

optionText [string](#)

isCorrect [bool](#)

weight [float](#)

feedback [string](#)

Returns

[MultipleChoiceQuestionOption](#)

# Class Question

Namespace: [Domain.Entities.Examiner](#)

Assembly: Domain.dll

```
public class Question : EntityAudit
```

## Inheritance

object ← [EntityAudit](#) ← Question

## Inherited Members

[EntityAudit.LastModifiedDate](#) , [EntityAudit.LastArchiveDate](#) , [EntityAudit.CreationDate](#) ,  
[EntityAudit.IsArchived](#) , [EntityAudit.VersionNumber](#) , [EntityAudit.Created\(\)](#) , [EntityAudit.Updated\(\)](#) ,  
[EntityAudit.Archive\(\)](#) , [EntityAudit.UnArchive\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Constructors

Question(string, List<string>?, QuestionType, int, bool,  
QuestionDifficulty, Guid?, List<Guid>?, List<Guid>?, Guid?)

```
public Question(string questionText, List<string>? variants, QuestionType questionType, int  
mark, bool requireManualReview, QuestionDifficulty difficulty, Guid? language = null,  
List<Guid>? tags = null, List<Guid>? sources = null, Guid? category = null)
```

## Parameters

questionText [string](#)

variants [List](#)<[string](#)>

questionType [QuestionType](#)

mark [int](#)

requireManualReview [bool](#)

**difficulty** [QuestionDifficulty](#)

**language** [Guid](#)?

**tags** [List](#)<[Guid](#)>

**sources** [List](#)<[Guid](#)>

**category** [Guid](#)?

## Properties

### Category

The category under which this question falls, if any.

```
public Guid? Category { get; }
```

### Property Value

[Guid](#)?

### DifficultyIndex

An index (0–100) representing how difficult the question is perceived to be.

```
public short DifficultyIndex { get; }
```

### Property Value

[short](#)

### HasAttachments

Determine if the

```
public bool HasAttachments { get; }
```

Property Value

[bool](#)

Id

Unique identifier for the question.

```
public Guid Id { get; }
```

Property Value

[Guid](#)

ImagesFilePaths

File path for an image for handling images

```
public List<string> ImagesFilePaths { get; }
```

Property Value

[List](#)<[string](#)>

Language

```
public Guid? Language { get; }
```

Property Value

[Guid](#)?

LongAnswerQuestion

Details specific to long answer questions (if applicable).

```
public LongAnswerQuestion? LongAnswerQuestion { get; }
```

## Property Value

[LongAnswerQuestion](#)

## Mark

The number of points (marks) assigned to this question.

```
public int Mark { get; }
```

## Property Value

[int](#)

## MultipleChoiceQuestion

Details specific to multiple choice questions (if applicable).

```
public MultipleChoiceQuestion? MultipleChoiceQuestion { get; }
```

## Property Value

[MultipleChoiceQuestion](#)

## QuestionText

The actual text of the question displayed to the user.

```
public string QuestionText { get; }
```

## Property Value

[string](#)

## QuestionType

The type of the question, e.g., MultipleChoice, TrueFalse, ShortAnswer.

```
public QuestionType QuestionType { get; }
```

Property Value

[QuestionType](#)

## ReorderingQuestions

Details specific to reordering questions (if applicable).

```
public List<ReorderingQuestion>? ReorderingQuestions { get; }
```

Property Value

[List](#) <[ReorderingQuestion](#)>

## RequireManualReview

Indicates whether the question needs manual grading (e.g., short answer).

```
public bool RequireManualReview { get; }
```

Property Value

[bool](#)

## ShortAnswerQuestion

Details specific to short answer questions (if applicable).

```
public ShortAnswerQuestion? ShortAnswerQuestion { get; }
```

## Property Value

[ShortAnswerQuestion](#)

## Sources

Optional list of sources (e.g., books, websites) from which the question originates.

```
public List<Guid>? Sources { get; }
```

## Property Value

[List<Guid>](#)

## Tags

```
public List<Guid>? Tags { get; }
```

## Property Value

[List<Guid>](#)

## TrueFalseQuestion

Details specific to true/false questions (if applicable).

```
public TrueFalseQuestion? TrueFalseQuestion { get; }
```

## Property Value

[TrueFalseQuestion](#)

## Variants

Different phrase of questions

```
public List<string>? Variants { get; }
```

Property Value

[List](#) <[string](#)>

## Methods

CreateLongAnswer(int?, int?, string, string?)

```
public void CreateLongAnswer(int? maximinWords, int? minimalWords, string answer,  
    string? feedback)
```

Parameters

maximinWords [int](#)?

minimalWords [int](#)?

answer [string](#)

feedback [string](#)

CreateMultipleChoiceQuestion(MultipleChoiseQuestion)

```
public void CreateMultipleChoiceQuestion(MultipleChoiseQuestion multipleChoiseQuestion)
```

Parameters

multipleChoiseQuestion [MultipleChoiseQuestion](#)

CreateReordering(IEnumerable<ReorderingQuestion>)

```
public void CreateReordering(IEnumerable<ReorderingQuestion> reorderingQuestions)
```

Parameters

reorderingQuestions [IEnumerable](#)<[ReorderingQuestion](#)>

## CreateShortAnswer(string, List<string>?, string?, string?)

```
public void CreateShortAnswer(string correctAnswer, List<string>? possibleAnswers, string?  
wrongAnswerFeedback, string? correctAnswerFeedBack)
```

Parameters

correctAnswer [string](#)

possibleAnswers [List](#)<[string](#)>

wrongAnswerFeedback [string](#)

correctAnswerFeedBack [string](#)

## CreateTrueAndFalse(TrueFalseQuestion)

```
public void CreateTrueAndFalse(TrueFalseQuestion trueFalseQuestion)
```

Parameters

trueFalseQuestion [TrueFalseQuestion](#)

## RemoveTag(Guid)

```
public void RemoveTag(Guid tagId)
```

Parameters

tagId [Guid](#)

## UpdateBasicQueastion(string, List<string>?, int, bool, List<string>, short)

```
public void UpdateBasicQueastion(string questionText, List<string>? variants, int mark, bool requireManualReview, List<string> tags, short difficultyIndex)
```

### Parameters

questionText [string](#)

variants [List](#)<[string](#)>

mark [int](#)

requireManualReview [bool](#)

tags [List](#)<[string](#)>

difficultyIndex [short](#)

# Class QuestionAttachments

Namespace: [Domain.Entities.Examiner](#)

Assembly: Domain.dll

```
public class QuestionAttachments
```

## Inheritance

object ← QuestionAttachments

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Properties

### FilePath

```
public string FilePath { get; set; }
```

Property Value

[string](#) ↗

### MyProperty

```
public int MyProperty { get; set; }
```

Property Value

[int](#) ↗

# Class ReorderingQuestion

Namespace: [Domain.Entities.Examiner](#)

Assembly: Domain.dll

```
public class ReorderingQuestion
```

## Inheritance

object ← ReorderingQuestion

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### ReorderingQuestion(string, int, string?, string?)

```
public ReorderingQuestion(string value, int order, string? wrongAnswerFeedBack,  
string? correctAnswerFeedBack)
```

## Parameters

value [string](#)

order [int](#)

wrongAnswerFeedBack [string](#)

correctAnswerFeedBack [string](#)

## Properties

### CorrectAnswerFeedBack

```
public string? CorrectAnswerFeedBack { get; }
```

PropertyValue

[string](#) ↗

Id

```
public Guid Id { get; }
```

PropertyValue

[Guid](#) ↗

Order

```
public int Order { get; }
```

PropertyValue

[int](#) ↗

Value

```
public string Value { get; }
```

PropertyValue

[string](#) ↗

WrongAnswerFeedBack

```
public string? WrongAnswerFeedBack { get; }
```

PropertyValue

[string](#) ↗

# Class ShortAnswerQuestion

Namespace: [Domain.Entities.Examiner](#)

Assembly: Domain.dll

```
public class ShortAnswerQuestion
```

## Inheritance

object ← ShortAnswerQuestion

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

ShortAnswerQuestion(string, List<string>?, string?, string?)

```
public ShortAnswerQuestion(string correctAnswer, List<string>? possibleAnswers, string?  
wrongAnswerFeedBack, string? correctAnswerFeedBack)
```

## Parameters

correctAnswer [string](#)

possibleAnswers [List](#)<[string](#)>

wrongAnswerFeedBack [string](#)

correctAnswerFeedBack [string](#)

## Properties

### CorrectAnswer

```
public string CorrectAnswer { get; }
```

PropertyValue

[string](#)

## CorrectAnswerFeedBack

```
public string? CorrectAnswerFeedBack { get; }
```

PropertyValue

[string](#)

## Id

```
public Guid Id { get; set; }
```

PropertyValue

[Guid](#)

## PossibleAnswers

```
public List<string>? PossibleAnswers { get; }
```

PropertyValue

[List](#)<[string](#)>

## WrongAnswerFeedBack

```
public string? WrongAnswerFeedBack { get; }
```

PropertyValue

[string](#) ↗

# Class TrueFalseQuestion

Namespace: [Domain.Entities.Examiner](#)

Assembly: Domain.dll

```
public class TrueFalseQuestion
```

## Inheritance

object ← TrueFalseQuestion

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### TrueFalseQuestion(bool, string?, string?)

```
public TrueFalseQuestion(bool isCorrect, string? wrongAnswerFeedBack,  
string? answerFeedBack)
```

## Parameters

isCorrect [bool](#)

wrongAnswerFeedBack [string](#)

answerFeedBack [string](#)

## Properties

### AnswerFeedBack

```
public string? AnswerFeedBack { get; set; }
```

## Property Value

[string](#)

## Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

## IsCorrect

```
public bool IsCorrect { get; set; }
```

Property Value

[bool](#)

## Justification

```
public string? Justification { get; set; }
```

Property Value

[string](#)

## RequireJustification

```
public bool RequireJustification { get; set; }
```

Property Value

[bool](#)

## WrongAnswerFeedBack

```
public string? WrongAnswerFeedBack { get; set; }
```

Property Value

[string](#) ↗

# Namespace Domain.Entities.Sources

## Classes

[Metadata](#)

[Source](#)

[SourceReference](#)

# Class Metadata

Namespace: [Domain.Entities.Sources](#)

Assembly: Domain.dll

```
public class Metadata
```

## Inheritance

object ← Metadata

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

Metadata(string, string, FiledType)

```
public Metadata(string filedName, string value, FiledType fileType)
```

### Parameters

fileName [string](#)

value [string](#)

filedType [FiledType](#)

## Properties

FileName

```
public string FileName { get; set; }
```

### Property Value

[string](#)

## FileType

```
public FileType FileType { get; set; }
```

Property Value

[FileType](#)

## Id

```
public Guid Id { get; set; }
```

Property Value

[Guid](#)

## Value

```
public string Value { get; set; }
```

Property Value

[string](#)

# Class Source

Namespace: [Domain.Entities.Sources](#)

Assembly: Domain.dll

```
public class Source : EntityAudit
```

## Inheritance

object ← [EntityAudit](#) ← Source

## Inherited Members

[EntityAudit.Id](#) , [EntityAudit.LastModifiedDate](#) , [EntityAudit.LastArchiveDate](#) , [EntityAudit.CreationDate](#) ,  
[EntityAudit.IsArchived](#) , [EntityAudit.VersionNumber](#) , [EntityAudit.Created\(\)](#) , [EntityAudit.Updated\(\)](#) ,  
[EntityAudit.Archive\(\)](#) , [EntityAudit.UnArchive\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

# Constructors

Source(SourceType, string, string?, bool, string, string,  
IEnumerable<Guid>?, Guid?)

```
public Source(SourceType type, string title, string? description, bool hasAttachment, string  
fileExtension, string filePath, IEnumerable<Guid>? tags, Guid? categoryId)
```

## Parameters

type [SourceType](#)

title [string](#) ↴

description [string](#) ↴

hasAttachment [bool](#) ↴

fileExtension [string](#) ↴

filePath [string](#) ↴

tags [IEnumerable](#)<[Guid](#)>

categoryId [Guid](#)?

## Properties

### CategoryId

`public Guid? CategoryId { get; }`

Property Value

[Guid](#)?

### Description

`public string? Description { get; }`

Property Value

[string](#)

### FileExtension

`public string? FileExtension { get; }`

Property Value

[string](#)

### FilePath

`public string? FilePath { get; }`

Property Value

[string](#)

## HasAttachment

`public bool HasAttachment { get; }`

Property Value

[bool](#)

## Metadata

`public List<Metadata>? Metadata { get; }`

Property Value

[List](#) <[Metadata](#)>

## Tags

`public List<Guid>? Tags { get; }`

Property Value

[List](#) <[Guid](#)>

## Title

`public string Title { get; }`

Property Value

[string](#)

## Type

```
public SourceType Type { get; }
```

### Property Value

[SourceType](#)

## Methods

### AddMetadata(Metadata)

```
public void AddMetadata(Metadata metadata)
```

#### Parameters

metadata [Metadata](#)

### AddMetadata(IEnumerable<Metadata>)

```
public void AddMetadata(IEnumerable<Metadata> metadata)
```

#### Parameters

metadata [IEnumerable](#)<[Metadata](#)>

### AddMetadata(string, string, FiledType)

```
public void AddMetadata(string fileName, string value, FiledType fileType)
```

#### Parameters

`fileName` [string](#)

`value` [string](#)

`filedType` [FiledType](#)

## AddNewTag(IEnumerable<Guid>)

`public void AddNewTag(IEnumerable<Guid> tagIds)`

### Parameters

`tagIds` [IEnumerable](#)<[Guid](#)>

## AddNewTag(Guid)

`public void AddNewTag(Guid tagId)`

### Parameters

`tagId` [Guid](#)

## RemoveTag(Guid)

`public void RemoveTag(Guid id)`

### Parameters

`id` [Guid](#)

## UpdateSource(SourceType, string, string?, IEnumerable<Guid>?)

`public void UpdateSource(SourceType type, string title, string? description, IEnumerable<Guid>? tags)`

## Parameters

**type** [SourceType](#)

**title** [string](#)

**description** [string](#)

**tags** [IEnumerable](#) <[Guid](#)>

# Class SourceReference

Namespace: [Domain.Entities.Sources](#)

Assembly: Domain.dll

```
public class SourceReference : EntityAudit
```

## Inheritance

object ← [EntityAudit](#) ← SourceReference

## Inherited Members

[EntityAudit.LastModifiedDate](#) , [EntityAudit.LastArchiveDate](#) , [EntityAudit.CreationDate](#) ,  
[EntityAudit.IsArchived](#) , [EntityAudit.VersionNumber](#) , [EntityAudit.Created\(\)](#) , [EntityAudit.Updated\(\)](#) ,  
[EntityAudit.Archive\(\)](#) , [EntityAudit.UnArchive\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### SourceReference(string?, Guid)

```
public SourceReference(string? notes, Guid sourceId)
```

## Parameters

notes [string](#) ↗

sourceId [Guid](#) ↗

## Properties

### Id

```
public Guid Id { get; }
```

Property Value

[Guid ↗](#)

## Metadata

```
public List<Metadata>? Metadata { get; }
```

Property Value

[List ↗ <Metadata>](#)

## Notes

```
public string? Notes { get; }
```

Property Value

[string ↗](#)

## SourceId

```
public Guid SourceId { get; }
```

Property Value

[Guid ↗](#)

## Methods

### AddMetadata(Metadata)

```
public void AddMetadata(Metadata metadata)
```

Parameters

**metadata** [Metadata](#)

## AddMetadata(IEnumerable<Metadata>)

```
public void AddMetadata(IEnumerable<Metadata> metadata)
```

Parameters

**metadata** [IEnumerable](#)<[Metadata](#)>

## AddMetadata(string, string, FiledType)

```
public void AddMetadata(string fileName, string value, FiledType fileType)
```

Parameters

**fileName** [string](#)

**value** [string](#)

**filedType** [FiledType](#)

# Namespace Domain.Enums

## Enums

[ActionBy](#)

[ActionType](#)

[ColorCategory](#)

[EntityName](#)

[FileFormat](#)

[FileType](#)

[MultipleChoiseQuestionType](#)

[OrderType](#)

[QuestionDifficulty](#)

[QuestionType](#)

[SourceType](#)

# Enum ActionBy

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum ActionBy
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

System = 2

User = 1

# Enum ActionType

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum ActionType
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

AddNewTag = 6

Added = 1

Archived = 4

Deleted = 3

Modified = 2

RemoveTag = 7

UnArchived = 5

# Enum ColorCategory

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum ColorCategory
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

Black = 10

Blue = 6

Cyan = 5

Gray = 11

Green = 4

Orange = 2

Pink = 8

Purple = 7

Red = 1

White = 9

Yellow = 3

# Enum EntityName

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum EntityName
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

Category = 4

Reference = 3

Source = 2

Tag = 1

# Enum FileFormat

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum FileFormat
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

CSV = 2

JSON = 1

# Enum FiledType

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum FiledType
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

Date = 3

Number = 2

Text = 1

# Enum MultipleChoiseQuestionType

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum MultipleChoiseQuestionType
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

MuktipleChoise = 1

SingleChoise = 2

# Enum OrderType

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum OrderType
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

Ascending = 1

Descending = 2

# Enum QuestionDifficulty

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum QuestionDifficulty
```

## Extension Methods

[QuestionDifficultyExtenstion.GetMattrix\(QuestionDifficulty\)](#) , [ObjectExtention.IsNotNull\(object\)](#) ,  
[ObjectExtention.IsNull\(object\)](#)

## Fields

Average = 4

Basic = 7

Challenging = 3

Extreme = 1

Manageable = 5

Simple = 6

Tough = 2

# Enum QuestionType

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum QuestionType
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

LongAnswer = 4

MultipleChoice = 1

Reordering = 5

ShortAnswer = 3

TrueAndFalse = 2

# Enum SourceType

Namespace: [Domain.Enums](#)

Assembly: Domain.dll

```
public enum SourceType
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Fields

Article = 3

Book = 4

Podcast = 2

Video = 1

# Namespace Domain.Extentions

## Classes

[ColorExtension](#)

[ObjectExtention](#)

[StringExtension](#)

[TypeExtensions](#)

# Class ColorExtension

Namespace: [Domain.Extensions](#)

Assembly: Domain.dll

```
public static class ColorExtension
```

## Inheritance

object ← ColorExtension

## Methods

### GenerateRandomHexColor()

```
public static string GenerateRandomHexColor()
```

Returns

[string](#) ↗

### GetColorGroup(string)

```
public static ColorCategory GetColorGroup(this string hexColor)
```

Parameters

hexColor [string](#) ↗

Returns

[ColorCategory](#).

# Class ObjectExtention

Namespace: [Domain.Extentions](#)

Assembly: Domain.dll

```
public static class ObjectExtention
```

## Inheritance

object ← ObjectExtention

## Methods

### IsNotNull(object)

```
public static bool IsNotNull(this object value)
```

Parameters

**value** object

Returns

[bool](#)

### IsNull(object)

```
public static bool IsNull(this object value)
```

Parameters

**value** object

Returns

[bool](#)

# Class StringExtension

Namespace: [Domain.Extensions](#)

Assembly: Domain.dll

```
public static class StringExtension
```

## Inheritance

object ← StringExtension

## Methods

### IsHexColor(string)

```
public static bool IsHexColor(this string input)
```

Parameters

**input** [string](#) ↗

Returns

[bool](#) ↗

### IsNotNullOrEmpty(string)

```
public static bool IsNotNullOrEmpty(this string value)
```

Parameters

**value** [string](#) ↗

Returns

[bool](#) ↗

# Class TypeExtensions

Namespace: [Domain.Extensions](#)

Assembly: Domain.dll

```
public static class TypeExtensions
```

## Inheritance

object ← TypeExtensions

## Methods

### GetFriendlyName(Type)

```
public static string GetFriendlyName(this Type type)
```

#### Parameters

type [Type](#) ↗

#### Returns

[string](#) ↗

# Namespace Domain.Lookups

## Classes

[EntityHistoryTypeLookup](#)

[QuestionDifficultyLookup](#)

[QuestionTypeLookup](#)

[SourceTypeLookup](#)

# Class EntityHistoryTypeLookup

Namespace: [Domain.Lookups](#)

Assembly: Domain.dll

```
public class EntityHistoryTypeLookup
```

## Inheritance

object ← EntityHistoryTypeLookup

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Id

```
public ActionType Id { get; set; }
```

Property Value

[ActionType](#)

### Value

```
public string Value { get; }
```

Property Value

[string](#) ↗

# Class QuestionDifficultyLookup

Namespace: [Domain.Lookups](#)

Assembly: Domain.dll

```
public class QuestionDifficultyLookup
```

## Inheritance

object ← QuestionDifficultyLookup

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Id

```
public QuestionDifficulty Id { get; set; }
```

Property Value

[QuestionDifficulty](#)

### Value

```
public string Value { get; }
```

Property Value

[string](#) ↗

# Class QuestionTypeLookup

Namespace: [Domain.Lookups](#)

Assembly: Domain.dll

```
public class QuestionTypeLookup
```

## Inheritance

object ← QuestionTypeLookup

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### Id

```
public QuestionType Id { get; set; }
```

Property Value

[QuestionType](#)

### Value

```
public string Value { get; }
```

Property Value

[string](#) ↗

# Class SourceTypeLookup

Namespace: [Domain.Lookups](#)

Assembly: Domain.dll

```
public class SourceTypeLookup
```

## Inheritance

object ← SourceTypeLookup

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### SourceTypeLookup(SourceType)

```
public SourceTypeLookup(SourceType id)
```

## Parameters

**id** [SourceType](#)

## Properties

### Id

```
public SourceType Id { get; set; }
```

## Property Value

[SourceType](#)

## Value

```
public string Value { get; }
```

Property Value

[string](#) ↗

# Namespace Domain.Managers

## Interfaces

[IRepositoryManager](#)

# Interface IRepositoryManager

Namespace: [Domain.Managers](#)

Assembly: Domain.dll

```
public interface IRepositoryManager : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Properties

### CategoryRepository

```
ICategoryRepository CategoryRepository { get; }
```

#### Property Value

[ICategoryRepository](#)

### LanguageRepository

```
ILanguageRepository LanguageRepository { get; }
```

#### Property Value

[ILanguageRepository](#)

### QuestionRepository

```
IQuestionRepository QuestionRepository { get; }
```

Property Value

[IQuestionRepository](#)

## ReferenceRepository

```
IReferenceRepository ReferenceRepository { get; }
```

Property Value

[IReferenceRepository](#)

## SourceRepository

```
ISourceRepository SourceRepository { get; }
```

Property Value

[ISourceRepository](#)

## TagRepository

```
ITagRepository TagRepository { get; }
```

Property Value

[ITagRepository](#)

# Namespace Domain.Repositories

## Interfaces

[IApplicationLogRepository](#)

[ICategoryRepository](#)

[ILanguageRepository](#)

[IQuestionRepository](#)

[IReferenceRepository](#)

[ISourceRepository](#)

[ITagRepository](#)

# Interface IApplicationLogRepository

Namespace: [Domain.Repositories](#)

Assembly: Domain.dll

```
public interface IApplicationLogRepository
```

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Methods

### Count()

```
int Count()
```

Returns

[int](#)

### GetAll(Func<ApplicationLog, bool>, int, int)

```
IEnumerable<ApplicationLog> GetAll(Func<ApplicationLog, bool> func, int pageNumber,  
int pageSize)
```

Parameters

func [Func](#)<[ApplicationLog](#), [bool](#)>

pageNumber [int](#)

pageSize [int](#)

Returns

[IEnumerable](#)<[ApplicationLog](#)>



# Interface ICategoryRepository

Namespace: [Domain.Repositories](#)

Assembly: Domain.dll

```
public interface ICategoryRepository : IBaseRepository<Category>, IDisposable
```

## Inherited Members

[IBaseRepository<Category>.Insert\(Category\)](#) ,  
[IBaseRepository<Category>.Insert\(IEnumerable<Category>\)](#) ,  
[IBaseRepository<Category>.GetById\(Guid\)](#) , [IBaseRepository<Category>.GetAll\(\)](#) ,  
[IBaseRepository<Category>.GetAll\(Func<Category, bool>\)](#) ,  
[IBaseRepository<Category>.GetAll\(Func<Category, bool>, int, int\)](#) ,  
[IBaseRepository<Category>.GetAllAsync\(Func<Category, bool>, int, int\)](#) ,  
[IBaseRepository<Category>.GetAll\(int, int\)](#) , [IBaseRepository<Category>.GetCollection\(\)](#) ,  
[IBaseRepository<Category>.Count\(\)](#) , [IBaseRepository<Category>.CountBy\(Func<Category, bool>\)](#) ,  
[IBaseRepository<Category>.CountAsync\(\)](#) , [IBaseRepository<Category>.DeleteById\(Guid\)](#) ,  
[IBaseRepository<Category>.DeleteAll\(\)](#) , [IBaseRepository<Category>.Update\(Category\)](#) ,  
[IBaseRepository<Category>.Update\(IEnumerable<Category>\)](#) , [IDisposable.Dispose\(\)](#) ↗

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### IsNotExists(Guid)

```
bool IsNotExists(Guid id)
```

#### Parameters

**id** [Guid](#) ↗

#### Returns

[bool](#) ↗

# Interface ILanguageRepository

Namespace: [Domain.Repositories](#)

Assembly: Domain.dll

```
public interface ILanguageRepository : IBaseRepository<Language>, IDisposable
```

## Inherited Members

[IBaseRepository<Language>.Insert\(Language\)](#) ,  
[IBaseRepository<Language>.Insert\(IEnumerable<Language>\)](#) ,  
[IBaseRepository<Language>.GetById\(Guid\)](#) , [IBaseRepository<Language>.GetAll\(\)](#) ,  
[IBaseRepository<Language>.GetAll\(Func<Language, bool>\)](#) ,  
[IBaseRepository<Language>.GetAll\(Func<Language, bool>, int, int\)](#) ,  
[IBaseRepository<Language>.GetAllAsync\(Func<Language, bool>, int, int\)](#) ,  
[IBaseRepository<Language>.GetAll\(int, int\)](#) , [IBaseRepository<Language>.GetCollection\(\)](#) ,  
[IBaseRepository<Language>.Count\(\)](#) , [IBaseRepository<Language>.CountBy\(Func<Language, bool>\)](#) ,  
[IBaseRepository<Language>.CountAsync\(\)](#) , [IBaseRepository<Language>.DeleteById\(Guid\)](#) ,  
[IBaseRepository<Language>.DeleteAll\(\)](#) , [IBaseRepository<Language>.Update\(Language\)](#) ,  
[IBaseRepository<Language>.Update\(IEnumerable<Language>\)](#) , [IDisposable.Dispose\(\)](#) ↗

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNotNull\(object\)](#)

## Methods

### GetLanguageReference(string)

Guid **GetLanguageReference**(string code)

#### Parameters

code [string](#) ↗

#### Returns

[Guid](#) ↗

## IsLangCodeExists(string)

```
bool IsLangCodeExists(string code)
```

Parameters

code [string](#)

Returns

[bool](#)

# Interface IQuestionRepository

Namespace: [Domain.Repositories](#)

Assembly: Domain.dll

```
public interface IQuestionRepository : IBaseRepository<Question>, IDisposable
```

## Inherited Members

[IBaseRepository<Question>.Insert\(Question\)](#) ,  
[IBaseRepository<Question>.Insert\(IEnumerable<Question>\)](#) ,  
[IBaseRepository<Question>.GetById\(Guid\)](#) , [IBaseRepository<Question>.GetAll\(\)](#) ,  
[IBaseRepository<Question>.GetAll\(Func<Question, bool>\)](#) ,  
[IBaseRepository<Question>.GetAll\(Func<Question, bool>, int, int\)](#) ,  
[IBaseRepository<Question>.GetAllAsync\(Func<Question, bool>, int, int\)](#) ,  
[IBaseRepository<Question>.GetAll\(int, int\)](#) , [IBaseRepository<Question>.GetCollection\(\)](#) ,  
[IBaseRepository<Question>.Count\(\)](#) , [IBaseRepository<Question>.CountBy\(Func<Question, bool>\)](#) ,  
[IBaseRepository<Question>.CountAsync\(\)](#) , [IBaseRepository<Question>.DeleteById\(Guid\)](#) ,  
[IBaseRepository<Question>.DeleteAll\(\)](#) , [IBaseRepository<Question>.Update\(Question\)](#) ,  
[IBaseRepository<Question>.Update\(IEnumerable<Question>\)](#) , [IDisposable.Dispose\(\)](#) ↗

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Interface IReferenceRepository

Namespace: [Domain.Repositories](#)

Assembly: Domain.dll

```
public interface IReferenceRepository : IBaseRepository<SourceReference>, IDisposable
```

## Inherited Members

[IBaseRepository<SourceReference>.Insert\(SourceReference\)](#) ,  
[IBaseRepository<SourceReference>.Insert\(IEnumerable<SourceReference>\)](#) ,  
[IBaseRepository<SourceReference>.GetById\(Guid\)](#) , [IBaseRepository<SourceReference>.GetAll\(\)](#) ,  
[IBaseRepository<SourceReference>.GetAll\(Func<SourceReference, bool>\)](#) ,  
[IBaseRepository<SourceReference>.GetAll\(Func<SourceReference, bool>, int, int\)](#) ,  
[IBaseRepository<SourceReference>.GetAllAsync\(Func<SourceReference, bool>, int, int\)](#) ,  
[IBaseRepository<SourceReference>.GetAll\(int, int\)](#) , [IBaseRepository<SourceReference>.GetCollection\(\)](#) ,  
[IBaseRepository<SourceReference>.Count\(\)](#) ,  
[IBaseRepository<SourceReference>.CountBy\(Func<SourceReference, bool>\)](#) ,  
[IBaseRepository<SourceReference>.CountAsync\(\)](#) ,  
[IBaseRepository<SourceReference>.DeleteById\(Guid\)](#) , [IBaseRepository<SourceReference>.DeleteAll\(\)](#) ,  
[IBaseRepository<SourceReference>.Update\(SourceReference\)](#) ,  
[IBaseRepository<SourceReference>.Update\(IEnumerable<SourceReference>\)](#) , [IDisposable.Dispose\(\)](#) ↗

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNotNull\(object\)](#).

# Interface ISourceRepository

Namespace: [Domain.Repositories](#)

Assembly: Domain.dll

```
public interface ISourceRepository : IBaseRepository<Source>, IDisposable
```

## Inherited Members

[IBaseRepository<Source>.Insert\(Source\)](#) , [IBaseRepository<Source>.Insert\(IEnumerable<Source>\)](#) ,  
[IBaseRepository<Source>.GetById\(Guid\)](#) , [IBaseRepository<Source>.GetAll\(\)](#) ,  
[IBaseRepository<Source>.GetAll\(Func<Source, bool>\)](#) ,  
[IBaseRepository<Source>.GetAll\(Func<Source, bool>, int, int\)](#) ,  
[IBaseRepository<Source>.GetAllAsync\(Func<Source, bool>, int, int\)](#) ,  
[IBaseRepository<Source>.GetAll\(int, int\)](#) , [IBaseRepository<Source>.GetCollection\(\)](#) ,  
[IBaseRepository<Source>.Count\(\)](#) , [IBaseRepository<Source>.CountBy\(Func<Source, bool>\)](#) ,  
[IBaseRepository<Source>.CountAsync\(\)](#) , [IBaseRepository<Source>.DeleteById\(Guid\)](#) ,  
[IBaseRepository<Source>.DeleteAll\(\)](#) , [IBaseRepository<Source>.Update\(Source\)](#) ,  
[IBaseRepository<Source>.Update\(IEnumerable<Source>\)](#) , [IDisposable.Dispose\(\)](#) ↴

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### IsExist(Guid)

```
bool IsExist(Guid id)
```

#### Parameters

**id** [Guid](#) ↴

#### Returns

[bool](#) ↴

## IsNotExist(Guid)

```
bool IsNotExist(Guid id)
```

Parameters

**id** [Guid](#)

Returns

[bool](#)

# Interface ITagRepository

Namespace: [Domain.Repositories](#)

Assembly: Domain.dll

```
public interface ITagRepository : IBaseRepository<Tag>, IDisposable
```

## Inherited Members

[IBaseRepository<Tag>.Insert\(Tag\)](#) , [IBaseRepository<Tag>.Insert\(IEnumerable<Tag>\)](#) ,  
[IBaseRepository<Tag>.GetById\(Guid\)](#) , [IBaseRepository<Tag>.GetAll\(\)](#) ,  
[IBaseRepository<Tag>.GetAll\(Func<Tag, bool>\)](#) , [IBaseRepository<Tag>.GetAll\(Func<Tag, bool>, int, int\)](#) ,  
[IBaseRepository<Tag>.GetAllAsync\(Func<Tag, bool>, int, int\)](#) , [IBaseRepository<Tag>.GetAll\(int, int\)](#) ,  
[IBaseRepository<Tag>.GetCollection\(\)](#) , [IBaseRepository<Tag>.Count\(\)](#) ,  
[IBaseRepository<Tag>.CountBy\(Func<Tag, bool>\)](#) , [IBaseRepository<Tag>.CountAsync\(\)](#) ,  
[IBaseRepository<Tag>.DeleteById\(Guid\)](#) , [IBaseRepository<Tag>.DeleteAll\(\)](#) ,  
[IBaseRepository<Tag>.Update\(Tag\)](#) , [IBaseRepository<Tag>.Update\(IEnumerable<Tag>\)](#) ,  
[IDisposable.Dispose\(\)](#) ↴

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### ArchiveTag(Guid)

```
void ArchiveTag(Guid id)
```

#### Parameters

**id** [Guid](#) ↴

### GetAllAvailableTags(IEnumerable<string>)

```
IEnumerable<string> GetAllAvailableTags(IEnumerable<string> tags)
```

Parameters

**tags** [IEnumerable<string>](#)

Returns

[IEnumerable<string>](#)

## GetNotFoundTags(IEnumerable<Guid>)

`IEnumerable<Guid> GetNotFoundTags(IEnumerable<Guid> inputIds)`

Parameters

**inputIds** [IEnumerable<Guid>](#)

Returns

[IEnumerable<Guid>](#)

## GetTagReference(string)

`Guid GetTagReference(string tag)`

Parameters

**tag** [string](#)

Returns

[Guid](#)

## GetTagsReference(IEnumerable<string>)

`IEnumerable<Guid> GetTagsReference(IEnumerable<string> tags)`

Parameters

**tags** [IEnumerable<string>](#)

Returns

[IEnumerable<Guid>](#)

## IsExist(IEnumerable<Guid>)

**bool** **IsExist**([IEnumerable<Guid>](#) ids)

Parameters

**ids** [IEnumerable<Guid>](#)

Returns

[bool](#)

## IsExist(Guid)

**bool** **IsExist**([Guid](#) id)

Parameters

**id** [Guid](#)

Returns

[bool](#)

## IsExist(string)

**bool** **IsExist**([string](#) name)

Parameters

`name` [string](#)

Returns

[bool](#)

## IsExist(string, Guid)

`bool IsExist(string name, Guid id)`

Parameters

`name` [string](#)

`id` [Guid](#)

Returns

[bool](#)

## IsNotExist(IEnumerable<Guid>)

`bool IsNotExist(IEnumerable<Guid> ids)`

Parameters

`ids` [IEnumerable](#) <[Guid](#)>

Returns

[bool](#)

## IsNotExist(Guid)

```
bool IsNotExist(Guid id)
```

Parameters

**id** [Guid](#)

Returns

[bool](#)

## UnArchiveTag(Guid)

```
void UnArchiveTag(Guid id)
```

Parameters

**id** [Guid](#)

# Namespace Domain.Repositories.Repository Base

## Interfaces

[IBaseRepository<T>](#)

# Interface IBaseRepository<T>

Namespace: [Domain.Repositories.RepositoryBase](#)

Assembly: Domain.dll

```
public interface IBaseRepository<T> : IDisposable where T : class
```

## Type Parameters

T

### Inherited Members

[IDisposable.Dispose\(\)](#)

### Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Methods

### Count()

```
int Count()
```

#### Returns

[int](#)

### CountAsync()

```
Task<int> CountAsync()
```

#### Returns

[Task](#)<[int](#)>

## CountBy(Func<T, bool>)

`int CountBy(Func<T, bool> func)`

Parameters

`func Func<T, bool>`

Returns

`int`

## DeleteAll()

`void DeleteAll()`

## DeleteById(Guid)

`void DeleteById(Guid id)`

Parameters

`id Guid`

## GetAll()

`IEnumerable<T> GetAll()`

Returns

`IEnumerable<T>`

## GetAll(Func<T, bool>)

```
IEnumerable<T> GetAll(Func<T, bool> func)
```

Parameters

func [Func<T, bool>](#)

Returns

[IEnumerable<T>](#)

## GetAll(Func<T, bool>, int, int)

```
IEnumerable<T> GetAll(Func<T, bool> func, int pageNumber, int pageSize)
```

Parameters

func [Func<T, bool>](#)

pageNumber [int](#)

pageSize [int](#)

Returns

[IEnumerable<T>](#)

## GetAll(int, int)

```
IEnumerable<T> GetAll(int pageNumber, int pageSize)
```

Parameters

pageNumber [int](#)

pageSize [int](#)

Returns

[IEnumerable](#)<T>

## GetAllAsync(Func<T, bool>, int, int)

Task<IEnumerable<T>> GetAllAsync(Func<T, bool> func, int pageNumber, int pageSize)

Parameters

func [Func](#)<T, bool>

pageNumber [int](#)

pageSize [int](#)

Returns

[Task](#)<IEnumerable<T>>

## GetById(Guid)

T [GetById](#)(Guid id)

Parameters

id [Guid](#)

Returns

T

## GetCollection()

[ILiteCollection](#)<T> [GetCollection](#)()

Returns

ILiteCollection<T>

## Insert(IEnumerable<T>)

```
void Insert(IEnumerable<T> entity)
```

Parameters

entity [IEnumerable](#)<T>

## Insert(T)

```
void Insert(T entity)
```

Parameters

entity T

## Update(IEnumerable<T>)

```
void Update(IEnumerable<T> entity)
```

Parameters

entity [IEnumerable](#)<T>

## Update(T)

```
void Update(T entity)
```

Parameters

entity T

# Namespace Infrastructure

## Classes

[InfrastructureRegisterServices](#)

# Class InfrastructureRegisterServices

Namespace: [Infrastructure](#)

Assembly: Infrastructure.dll

```
public static class InfrastructureRegisterServices
```

## Inheritance

object ← InfrastructureRegisterServices

## Methods

### RegisterInfrastructure(IServiceCollection)

```
public static IServiceCollection RegisterInfrastructure(this IServiceCollection services)
```

#### Parameters

services [IServiceCollection](#) ↗

#### Returns

[IServiceCollection](#) ↗

# Namespace Infrastructure.Audits

## Classes

[AuditTrailService](#)

# Class AuditTrailService

Namespace: [Infrastructure.Audits](#)

Assembly: Infrastructure.dll

```
public class AuditTrailService : BaseRepository<AuditTrail>, IBaseRepository<AuditTrail>,
IDisposable, IAuditTrailService
```

## Inheritance

object ← [BaseRepository<AuditTrail>](#) ← AuditTrailService

## Implements

[IBaseRepository<AuditTrail>](#), [IDisposable](#) ↗, [IAuditTrailService](#)

## Inherited Members

[BaseRepository<AuditTrail>.db](#), [BaseRepository<AuditTrail>.collection](#),  
[BaseRepository<AuditTrail>.collectionName](#), [BaseRepository<AuditTrail>.Insert\(AuditTrail\)](#),  
[BaseRepository<AuditTrail>.Insert\(IEnumerable<AuditTrail>\)](#),  
[BaseRepository<AuditTrail>.Update\(AuditTrail\)](#), [BaseRepository<AuditTrail>.DeleteByld\(Guid\)](#),  
[BaseRepository<AuditTrail>.DeleteAll\(\)](#), [BaseRepository<AuditTrail>.GetByld\(Guid\)](#),  
[BaseRepository<AuditTrail>.GetAll\(\)](#), [BaseRepository<AuditTrail>.GetAll\(Func<AuditTrail, bool>, int, int\)](#),  
[BaseRepository<AuditTrail>.GetAll\(Func<AuditTrail, bool>\)](#),  
[BaseRepository<AuditTrail>.GetAllAsync\(Func<AuditTrail, bool>, int, int\)](#),  
[BaseRepository<AuditTrail>.GetAll\(int, int\)](#), [BaseRepository<AuditTrail>.GetCollection\(\)](#),  
[BaseRepository<AuditTrail>.Count\(\)](#), [BaseRepository<AuditTrail>.CountBy\(Func<AuditTrail, bool>\)](#),  
[BaseRepository<AuditTrail>.CountAsync\(\)](#),  
[BaseRepository<AuditTrail>.Update\(IEnumerable<AuditTrail>\)](#), [BaseRepository<AuditTrail>.Dispose\(\)](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#), [ObjectExtention.IsNull\(object\)](#)

## Constructors

### AuditTrailService(string)

```
public AuditTrailService(string databasePath)
```

Parameters

databasePath [string](#)

## Methods

AddEntitiesBulk(IEnumerable<AuditTrail>)

```
public void AddEntitiesBulk(IEnumerable<AuditTrail> auditTrails)
```

Parameters

auditTrails [IEnumerable](#)<[AuditTrail](#)>

AddNewEntity<T>(EntityName, Guid, ActionBy, T, int, string?)

```
public void AddNewEntity<T>(EntityName entityName, Guid entityId, ActionBy actionBy, T entity, int versionNumber, string? comment = null)
```

Parameters

entityName [EntityName](#)

entityId [Guid](#)

actionBy [ActionBy](#)

entity T

versionNumber [int](#)

comment [string](#)

Type Parameters

T

## Count(Guid)

```
public int Count(Guid entityId)
```

Parameters

entityId [Guid](#)

Returns

[int](#)

## CountDeletedEntity()

```
public int CountDeletedEntity()
```

Returns

[int](#)

## DeleteEntity<T>(EntityName, Guid, ActionType, ActionBy, T, int, string?)

```
public void DeleteEntity<T>(EntityName entityName, Guid entityId, ActionType actionType,  
ActionBy actionBy, T entity, int versionNumber, string? comment = null)
```

Parameters

entityName [EntityName](#)

entityId [Guid](#)

actionType [ActionType](#)

actionBy [ActionBy](#)

entity T

versionNumber [int](#)

comment [string](#)

## Type Parameters

T

## EnqueueAudit(AuditTrail)

```
public void EnqueueAudit(AuditTrail audit)
```

## Parameters

audit [AuditTrail](#)

## GetDeletedEntities(Func<AuditTrail, bool>, int, int)

```
public IEnumerable<AuditTrail> GetDeletedEntities(Func<AuditTrail, bool> func, int  
pageNumber, int pageSize)
```

## Parameters

func [Func](#)<[AuditTrail](#), [bool](#)>

pageNumber [int](#)

pageSize [int](#)

## Returns

[IEnumerable](#)<[AuditTrail](#)>

## GetEntityTrail(Func<AuditTrail, bool>, int, int, Guid)

```
public IEnumerable<AuditTrail> GetEntityTrail(Func<AuditTrail, bool> func, int pageNumber,
```

```
int pageSize, Guid entityId)
```

Parameters

```
func Func<AuditTrail, bool>
```

pageNumber int

pageSize int

entityId Guid

Returns

```
IEnumerable<AuditTrail>
```

## GetEntityTrailDetails(Guid, EntityName, Guid)

```
public AuditTrail GetEntityTrailDetails(Guid trailId, EntityName entityName, Guid entityId)
```

Parameters

trailId Guid

entityName EntityName

entityId Guid

Returns

```
AuditTrail
```

## UpdateEntitiesBulk<T>(EntityName, IEnumerable<(Guid EntityId, ActionType ActionType, ActionBy ActionBy, T OldEntity, T NewEntity, int Version, string? Comment)>)

```
public void UpdateEntitiesBulk<T>(EntityName entityName, IEnumerable<(Guid EntityId, ActionType ActionType, ActionBy ActionBy, T OldEntity, T NewEntity, int Version, string?)>
```

```
Comment)> changes)
```

## Parameters

entityName [EntityName](#)

changes [IEnumerable](#)<(Guid [EntityId](#), [ActionType](#) [ActionType](#), [ActionBy](#) [ActionBy](#), T [OldEntity](#),  
T [NewEntity](#), int [Version](#), string [Comment](#))>

## Type Parameters

T

# UpdateEntity<T>(EntityName, Guid, ActionType, ActionBy, T, T, int, string?)

```
public void UpdateEntity<T>(EntityName entityName, Guid entityId, ActionType actionType,  
ActionBy actionBy, T oldEntity, T newEntity, int versionNumber, string? comment = null)
```

## Parameters

entityName [EntityName](#)

entityId [Guid](#)

actionType [ActionType](#)

actionBy [ActionBy](#)

oldEntity T

newEntity T

versionNumber [int](#)

comment [string](#)

## Type Parameters

T

# Namespace Infrastructure.Factories

## Classes

[JsonStringLocalizerFactory](#)

# Class JsonStringLocalizerFactory

Namespace: [Infrastructure.Factories](#)

Assembly: Infrastructure.dll

```
public class JsonStringLocalizerFactory : IStringLocalizerFactory
```

## Inheritance

object ← JsonStringLocalizerFactory

## Implements

[IStringLocalizerFactory](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### JsonStringLocalizerFactory(IDistributedCache)

```
public JsonStringLocalizerFactory(IDistributedCache cache)
```

## Parameters

cache [IDistributedCache](#)

## Methods

### Create(string, string)

Creates an [IStringLocalizer](#).

```
public IStringLocalizer Create(string basePath, string location)
```

## Parameters

**baseName** [string](#)

The base name of the resource to load strings from.

**location** [string](#)

The location to load resources from.

Returns

[IStringLocalizer](#)

The [IStringLocalizer](#).

## Create(Type)

Creates an [IStringLocalizer](#) using the [Assembly](#) and [FullName](#) of the specified [Type](#).

```
public IStringLocalizer Create(Type resourceSource)
```

Parameters

**resourceSource** [Type](#)

The [Type](#).

Returns

[IStringLocalizer](#)

The [IStringLocalizer](#).

# Namespace Infrastructure.Managers

## Classes

[AuditManager](#)

[RepositoryManager](#)

[ServiceManager](#)

[SystemManager](#)

# Class AuditManager

Namespace: [Infrastructure.Managers](#)

Assembly: Infrastructure.dll

```
public class AuditManager : IAuditManager
```

Inheritance

object ← AuditManager

Implements

[IAuditManager](#)

Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### AuditManager(IConfiguration)

```
public AuditManager(IConfiguration configuration)
```

Parameters

configuration  [IConfiguration](#)

## Properties

### ApplicationLogRepository

```
public IApplicationLogRepository ApplicationLogRepository { get; }
```

Property Value

[IApplicationLogRepository](#)

## AuditTrailService

```
public IAuditTrailService AuditTrailService { get; }
```

Property Value

[IAuditTrailService](#)

# Class RepositoryManager

Namespace: [Infrastructure.Managers](#)

Assembly: Infrastructure.dll

```
public class RepositoryManager : IRepositoryManager, IDisposable
```

Inheritance

object ← RepositoryManager

Implements

[IRepositoryManager](#), [IDisposable](#) ↗

Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### RepositoryManager(IConfiguration)

```
public RepositoryManager(IConfiguration configuration)
```

Parameters

configuration  [IConfiguration](#) ↗

## Properties

### CategoryRepository

```
public ICategoryRepository CategoryRepository { get; }
```

Property Value

[ICategoryRepository](#)

## LanguageRepository

```
public ILanguageRepository LanguageRepository { get; }
```

Property Value

[ILanguageRepository](#)

## QuestionRepository

```
public IQuestionRepository QuestionRepository { get; }
```

Property Value

[IQuestionRepository](#)

## ReferenceRepository

```
public IRepository ReferenceRepository { get; }
```

Property Value

[IReferenceRepository](#)

## SourceRepository

```
public IRepository SourceRepository { get; }
```

Property Value

[IRepository](#)

## TagRepository

```
public ITagRepository TagRepository { get; }
```

## Property Value

[ITagRepository](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

### Dispose(bool)

```
protected virtual void Dispose(bool disposing)
```

## Parameters

`disposing` [bool](#)

# Class ServiceManager

Namespace: [Infrastructure.Managers](#)

Assembly: Infrastructure.dll

```
public class ServiceManager : IServiceManager, IDisposable
```

## Inheritance

object ← ServiceManager

## Implements

[IServiceManager](#), [IDisposable](#) ↗

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

ServiceManager(IRepositoryManager, IAuditManager, IMapper, IServiceProvider)

```
public ServiceManager(IRepositoryManager repositoryManager, IAuditManager auditManager,
IMapper mapper, IServiceProvider serviceProvider)
```

## Parameters

repositoryManager [IRepositoryManager](#)

auditManager [IAuditManager](#)

mapper IMapper

serviceProvider [IServiceProvider](#) ↗

## Properties

CategoryService

```
public ICategoryService CategoryService { get; }
```

Property Value

[ICategoryService](#)

## LookupService

```
public ILookupService LookupService { get; }
```

Property Value

[ILookupService](#)

## QuestionService

```
public IQuestionService QuestionService { get; }
```

Property Value

[IQuestionService](#)

## SourceService

```
public ISourceService SourceService { get; }
```

Property Value

[ISourceService](#)

## TagService

```
public ITagService TagService { get; }
```

## Property Value

[ITagService](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

### Dispose(bool)

```
protected virtual void Dispose(bool disposing)
```

## Parameters

`disposing` [bool](#)

# Class SystemManager

Namespace: [Infrastructure.Managers](#)

Assembly: Infrastructure.dll

```
public class SystemManager : ISystemManager
```

Inheritance

object ← SystemManager

Implements

[ISystemManager](#)

Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### SystemManager(IConfiguration)

```
public SystemManager(IConfiguration configuration)
```

Parameters

configuration [IConfiguration](#) ↗

## Properties

### SystemService

```
public ISystemService SystemService { get; }
```

Property Value

[ISystemService](#)

# Namespace Infrastructure.Repositories

## Classes

[CategoryRepository](#)

[LanguageRepository](#)

[QuestionRepository](#)

[ReferenceRepository](#)

[SourceRepository](#)

[TagRepository](#)

# Class CategoryRepository

Namespace: [Infrastructure.Repositories](#)

Assembly: Infrastructure.dll

```
public class CategoryRepository : BaseRepository<Category>, ICategoryRepository,  
IBaseRepository<Category>, IDisposable
```

## Inheritance

object ← [BaseRepository<Category>](#) ← CategoryRepository

## Implements

[ICategoryRepository](#), [IBaseRepository<Category>](#), [IDisposable](#) ↗

## Inherited Members

[BaseRepository<Category>.db](#) , [BaseRepository<Category>.collection](#) ,  
[BaseRepository<Category>.collectionName](#) , [BaseRepository<Category>.Insert\(Category\)](#) .  
[BaseRepository<Category>.Insert\(IEnumerable<Category>\)](#) ,  
[BaseRepository<Category>.Update\(Category\)](#) . [BaseRepository<Category>.DeleteById\(Guid\)](#) ,  
[BaseRepository<Category>.DeleteAll\(\)](#) , [BaseRepository<Category>.GetById\(Guid\)](#) ,  
[BaseRepository<Category>.GetAll\(\)](#) , [BaseRepository<Category>.GetAll\(Func<Category, bool>, int, int\)](#) ,  
[BaseRepository<Category>.GetAll\(Func<Category, bool>\)](#) ,  
[BaseRepository<Category>.GetAllAsync\(Func<Category, bool>, int, int\)](#) ,  
[BaseRepository<Category>.GetAll\(int, int\)](#) , [BaseRepository<Category>.GetCollection\(\)](#) ,  
[BaseRepository<Category>.Count\(\)](#) , [BaseRepository<Category>.CountBy\(Func<Category, bool>\)](#) ,  
[BaseRepository<Category>.CountAsync\(\)](#) ,  
[BaseRepository<Category>.Update\(IEnumerable<Category>\)](#) , [BaseRepository<Category>.Dispose\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### CategoryRepository(string)

```
public CategoryRepository(string databasePath)
```

Parameters

`databasePath` [string](#)

## Methods

`IsNotExists(Guid)`

`public bool IsNotExists(Guid id)`

Parameters

`id` [Guid](#)

Returns

[bool](#)

# Class LanguageRepository

Namespace: [Infrastructure.Repositories](#)

Assembly: Infrastructure.dll

```
public class LanguageRepository : BaseRepository<Language>, ILanguageRepository,  
IBaseRepository<Language>, IDisposable
```

## Inheritance

object ← [BaseRepository<Language>](#) ← LanguageRepository

## Implements

[ILanguageRepository](#), [IBaseRepository<Language>](#), [IDisposable](#) ↗

## Inherited Members

[BaseRepository<Language>.db](#) , [BaseRepository<Language>.collection](#) ,  
[BaseRepository<Language>.collectionName](#) , [BaseRepository<Language>.Insert\(Language\)](#) ,  
[BaseRepository<Language>.Insert\(IEnumerable<Language>\)](#) ,  
[BaseRepository<Language>.Update\(Language\)](#) , [BaseRepository<Language>.DeleteById\(Guid\)](#) ,  
[BaseRepository<Language>.DeleteAll\(\)](#) , [BaseRepository<Language>.GetById\(Guid\)](#) ,  
[BaseRepository<Language>.GetAll\(\)](#) ,  
[BaseRepository<Language>.GetAll\(Func<Language, bool>, int, int\)](#) ,  
[BaseRepository<Language>.GetAll\(Func<Language, bool>\)](#) ,  
[BaseRepository<Language>.GetAllAsync\(Func<Language, bool>, int, int\)](#) ,  
[BaseRepository<Language>.GetAll\(int, int\)](#) , [BaseRepository<Language>.GetCollection\(\)](#) ,  
[BaseRepository<Language>.Count\(\)](#) , [BaseRepository<Language>.CountBy\(Func<Language, bool>\)](#) ,  
[BaseRepository<Language>.CountAsync\(\)](#) ,  
[BaseRepository<Language>.Update\(IEnumerable<Language>\)](#) , [BaseRepository<Language>.Dispose\(\)](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### LanguageRepository(string)

```
public LanguageRepository(string databasePath)
```

Parameters

databasePath [string](#)

## Methods

### GetLanguageReference(string)

```
public Guid GetLanguageReference(string code)
```

Parameters

code [string](#)

Returns

[Guid](#)

### IsLangCodeExists(string)

```
public bool IsLangCodeExists(string code)
```

Parameters

code [string](#)

Returns

[bool](#)

# Class QuestionRepository

Namespace: [Infrastructure.Repositories](#)

Assembly: Infrastructure.dll

```
public class QuestionRepository : BaseRepository<Question>, IQuestionRepository,  
IBaseRepository<Question>, IDisposable
```

## Inheritance

object ← [BaseRepository<Question>](#) ← QuestionRepository

## Implements

[IQuestionRepository](#), [IBaseRepository<Question>](#), [IDisposable](#) ↗

## Inherited Members

[BaseRepository<Question>.db](#) , [BaseRepository<Question>.collection](#) ,  
[BaseRepository<Question>.collectionName](#) , [BaseRepository<Question>.Insert\(Question\)](#) ,  
[BaseRepository<Question>.Insert\(IEnumerable<Question>\)](#) ,  
[BaseRepository<Question>.Update\(Question\)](#) , [BaseRepository<Question>.DeleteById\(Guid\)](#) ,  
[BaseRepository<Question>.DeleteAll\(\)](#) , [BaseRepository<Question>.GetById\(Guid\)](#) ,  
[BaseRepository<Question>.GetAll\(\)](#) , [BaseRepository<Question>.GetAll\(Func<Question, bool>, int, int\)](#) ,  
[BaseRepository<Question>.GetAll\(Func<Question, bool>\)](#) ,  
[BaseRepository<Question>.GetAllAsync\(Func<Question, bool>, int, int\)](#) ,  
[BaseRepository<Question>.GetAll\(int, int\)](#) , [BaseRepository<Question>.GetCollection\(\)](#) ,  
[BaseRepository<Question>.Count\(\)](#) , [BaseRepository<Question>.CountBy\(Func<Question, bool>\)](#) ,  
[BaseRepository<Question>.CountAsync\(\)](#) ,  
[BaseRepository<Question>.Update\(IEnumerable<Question>\)](#) , [BaseRepository<Question>.Dispose\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### QuestionRepository(string)

```
public QuestionRepository(string databasePath)
```

## Parameters

**databasePath** [string](#) ↗

# Class ReferenceRepository

Namespace: [Infrastructure.Repositories](#)

Assembly: Infrastructure.dll

```
public class ReferenceRepository : BaseRepository<SourceReference>, IReferenceRepository,  
IBaseRepository<SourceReference>, IDisposable
```

## Inheritance

object ← [BaseRepository<SourceReference>](#) ← ReferenceRepository

## Implements

[IReferenceRepository](#), [IBaseRepository<SourceReference>](#), [IDisposable](#) ↗

## Inherited Members

[BaseRepository<SourceReference>.db](#) , [BaseRepository<SourceReference>.collection](#) ,  
[BaseRepository<SourceReference>.collectionName](#) ,  
[BaseRepository<SourceReference>.Insert\(SourceReference\)](#) ,  
[BaseRepository<SourceReference>.Insert\(IEnumerable<SourceReference>\)](#) ,  
[BaseRepository<SourceReference>.Update\(SourceReference\)](#) ,  
[BaseRepository<SourceReference>.DeleteById\(Guid\)](#) , [BaseRepository<SourceReference>.DeleteAll\(\)](#) ,  
[BaseRepository<SourceReference>.GetById\(Guid\)](#) , [BaseRepository<SourceReference>.GetAll\(\)](#) ,  
[BaseRepository<SourceReference>.GetAll\(Func<SourceReference, bool>, int, int\)](#) ,  
[BaseRepository<SourceReference>.GetAll\(Func<SourceReference, bool>\)](#) ,  
[BaseRepository<SourceReference>.GetAllAsync\(Func<SourceReference, bool>, int, int\)](#) ,  
[BaseRepository<SourceReference>.GetAll\(int, int\)](#) , [BaseRepository<SourceReference>.GetCollection\(\)](#) ,  
[BaseRepository<SourceReference>.Count\(\)](#) ,  
[BaseRepository<SourceReference>.CountBy\(Func<SourceReference, bool>\)](#) ,  
[BaseRepository<SourceReference>.CountAsync\(\)](#) ,  
[BaseRepository<SourceReference>.Update\(IEnumerable<SourceReference>\)](#) ,  
[BaseRepository<SourceReference>.Dispose\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

# Constructors

ReferenceRepository(string)

```
public ReferenceRepository(string databasePath)
```

## Parameters

databasePath [string](#)

# Class SourceRepository

Namespace: [Infrastructure.Repositories](#)

Assembly: Infrastructure.dll

```
public class SourceRepository : BaseRepository<Source>, ISourceRepository,  
IBaseRepository<Source>, IDisposable
```

## Inheritance

object ← [BaseRepository<Source>](#) ← SourceRepository

## Implements

[ISourceRepository](#), [IBaseRepository<Source>](#), [IDisposable](#) ↗

## Inherited Members

[BaseRepository<Source>.db](#) , [BaseRepository<Source>.collection](#) ,  
[BaseRepository<Source>.collectionName](#) , [BaseRepository<Source>.Insert\(Source\)](#) ,  
[BaseRepository<Source>.Insert\(IEnumerable<Source>\)](#) , [BaseRepository<Source>.Update\(Source\)](#) ,  
[BaseRepository<Source>.DeleteById\(Guid\)](#) , [BaseRepository<Source>.DeleteAll\(\)](#) ,  
[BaseRepository<Source>.GetById\(Guid\)](#) , [BaseRepository<Source>.GetAll\(\)](#) ,  
[BaseRepository<Source>.GetAll\(Func<Source, bool>, int, int\)](#) ,  
[BaseRepository<Source>.GetAll\(Func<Source, bool>\)](#) ,  
[BaseRepository<Source>.GetAllAsync\(Func<Source, bool>, int, int\)](#) ,  
[BaseRepository<Source>.GetAll\(int, int\)](#) , [BaseRepository<Source>.GetCollection\(\)](#) ,  
[BaseRepository<Source>.Count\(\)](#) , [BaseRepository<Source>.CountBy\(Func<Source, bool>\)](#) ,  
[BaseRepository<Source>.CountAsync\(\)](#) , [BaseRepository<Source>.Update\(IEnumerable<Source>\)](#) ,  
[BaseRepository<Source>.Dispose\(\)](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### SourceRepository(string)

```
public SourceRepository(string databasePath)
```

Parameters

**databasePath** [string](#)

## Methods

### IsExist(Guid)

**public bool IsExist(Guid id)**

Parameters

**id** [Guid](#)

Returns

[bool](#)

### IsNotExist(Guid)

**public bool IsNotExist(Guid id)**

Parameters

**id** [Guid](#)

Returns

[bool](#)

# Class TagRepository

Namespace: [Infrastructure.Repositories](#)

Assembly: Infrastructure.dll

```
public class TagRepository : BaseRepository<Tag>, ITagRepository,  
IBaseRepository<Tag>, IDisposable
```

## Inheritance

object ← [BaseRepository<Tag>](#) ← TagRepository

## Implements

[ITagRepository](#), [IBaseRepository<Tag>](#), [IDisposable](#) ↗

## Inherited Members

[BaseRepository<Tag>.db](#), [BaseRepository<Tag>.collection](#), [BaseRepository<Tag>.collectionName](#),  
[BaseRepository<Tag>.Insert\(Tag\)](#), [BaseRepository<Tag>.Insert\(IEnumerable<Tag>\)](#),  
[BaseRepository<Tag>.Update\(Tag\)](#), [BaseRepository<Tag>.DeleteById\(Guid\)](#),  
[BaseRepository<Tag>.DeleteAll\(\)](#), [BaseRepository<Tag>.GetById\(Guid\)](#), [BaseRepository<Tag>.GetAll\(\)](#),  
[BaseRepository<Tag>.GetAll\(Func<Tag, bool>, int, int\)](#), [BaseRepository<Tag>.GetAll\(Func<Tag, bool>\)](#),  
[BaseRepository<Tag>.GetAllAsync\(Func<Tag, bool>, int, int\)](#), [BaseRepository<Tag>.GetAll\(int, int\)](#),  
[BaseRepository<Tag>.GetCollection\(\)](#), [BaseRepository<Tag>.Count\(\)](#),  
[BaseRepository<Tag>.CountBy\(Func<Tag, bool>\)](#), [BaseRepository<Tag>.CountAsync\(\)](#),  
[BaseRepository<Tag>.Update\(IEnumerable<Tag>\)](#), [BaseRepository<Tag>.Dispose\(\)](#).

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#), [ObjectExtention.IsNull\(object\)](#)

# Constructors

## TagRepository(string)

```
public TagRepository(string databasePath)
```

## Parameters

databasePath [string](#) ↗

# Methods

## ArchiveTag(Guid)

```
public void ArchiveTag(Guid id)
```

### Parameters

**id** [Guid](#)

## GetAllAvailableTags(IEnumerable<string>)

```
public IEnumerable<string> GetAllAvailableTags(IEnumerable<string> tags)
```

### Parameters

**tags** [IEnumerable](#)<[string](#)>

### Returns

[IEnumerable](#)<[string](#)>

## GetNotFoundTags(IEnumerable<Guid>)

```
public IEnumerable<Guid> GetNotFoundTags(IEnumerable<Guid> inputIds)
```

### Parameters

**inputIds** [IEnumerable](#)<[Guid](#)>

### Returns

[IEnumerable](#)<[Guid](#)>

## GetTagReference(string)

```
public Guid GetTagReference(string tag)
```

Parameters

tag [string](#)

Returns

[Guid](#)

## GetTagsReference(IEnumerable<string>)

```
public IEnumerable<Guid> GetTagsReference(IEnumerable<string> tags)
```

Parameters

tags [IEnumerable](#)<[string](#)>

Returns

[IEnumerable](#)<[Guid](#)>

## IsExist(IEnumerable<Guid>)

```
public bool IsExist(IEnumerable<Guid> ids)
```

Parameters

ids [IEnumerable](#)<[Guid](#)>

Returns

[bool](#)

## IsExist(Guid)

```
public bool IsExist(Guid id)
```

Parameters

id [Guid](#)

Returns

[bool](#)

## IsExist(string)

```
public bool IsExist(string name)
```

Parameters

name [string](#)

Returns

[bool](#)

## IsExist(string, Guid)

```
public bool IsExist(string name, Guid id)
```

Parameters

name [string](#)

id [Guid](#)

Returns

[bool](#)

## IsNotExist(IEnumerable<Guid>)

```
public bool IsNotExist(IEnumerable<Guid> ids)
```

Parameters

**ids** [IEnumerable<Guid>](#)

Returns

[bool](#)

## IsNotExist(Guid)

```
public bool IsNotExist(Guid id)
```

Parameters

**id** [Guid](#)

Returns

[bool](#)

## UnArchiveTag(Guid)

```
public void UnArchiveTag(Guid id)
```

Parameters

**id** [Guid](#)

# Namespace Infrastructure.Repositories.Base Repository

## Classes

[BaseRepository<T>](#)

# Class BaseRepository<T>

Namespace: [Infrastructure.Repositories.BaseRepository](#)

Assembly: Infrastructure.dll

```
public class BaseRepository<T> : IBaseRepository<T>, IDisposable where T : class
```

## Type Parameters

T

## Inheritance

object ← BaseRepository<T>

## Implements

[IBaseRepository](#)<T>, [IDisposable](#)

## Derived

[AuditTrailService](#), [CategoryRepository](#), [LanguageRepository](#), [QuestionRepository](#), [ReferenceRepository](#),  
[SourceRepository](#), [TagRepository](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#), [ObjectExtention.IsNull\(object\)](#)

# Constructors

## BaseRepository(string, string)

```
public BaseRepository(string databasePath, string collectionName)
```

## Parameters

databasePath [string](#)

collectionName [string](#)

# Fields

## \_collection

`protected readonly ILiteCollection<T> _collection`

### Field Value

`ILiteCollection<T>`

## \_collectionName

`protected readonly string _collectionName`

### Field Value

`string` ↗

## \_db

`protected readonly LiteDatabase _db`

### Field Value

`LiteDatabase`

## Methods

### Count()

`public virtual int Count()`

### Returns

`int` ↗

## CountAsync()

```
public virtual Task<int> CountAsync()
```

Returns

[Task](#) <[int](#)>

## CountBy(Func<T, bool>)

```
public virtual int CountBy(Func<T, bool> func)
```

Parameters

[func](#) [Func](#) <T, [bool](#)>

Returns

[int](#)

## DeleteAll()

```
public virtual void DeleteAll()
```

## DeleteById(Guid)

```
public virtual void DeleteById(Guid id)
```

Parameters

[id](#) [Guid](#)

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## GetAll()

```
public virtual IEnumerable<T> GetAll()
```

Returns

[IEnumerable](#) <T>

## GetAll(Func<T, bool>)

```
public virtual IEnumerable<T> GetAll(Func<T, bool> func)
```

Parameters

func [Func](#) <T, [bool](#)>

Returns

[IEnumerable](#) <T>

## GetAll(Func<T, bool>, int, int)

```
public virtual IEnumerable<T> GetAll(Func<T, bool> func, int pageNumber, int pageSize)
```

Parameters

func [Func](#) <T, [bool](#)>

pageNumber [int](#)

pageSize [int](#)

Returns

[IEnumerable](#) <T>

## GetAll(int, int)

```
public virtual IEnumerable<T> GetAll(int pageNumber, int pageSize)
```

Parameters

pageNumber [int](#)

pageSize [int](#)

Returns

[IEnumerable](#) <T>

## GetAllAsync(Func<T, bool>, int, int)

```
public virtual Task<IEnumerable<T>> GetAllAsync(Func<T, bool> func, int pageNumber,
int pageSize)
```

Parameters

func [Func](#) <T, [bool](#)>

pageNumber [int](#)

pageSize [int](#)

Returns

[Task](#) <[IEnumerable](#) <T>>

## GetById(Guid)

```
public virtual T GetById(Guid id)
```

Parameters

id [Guid](#)

Returns

T

## GetCollection()

```
public virtual ILiteCollection<T> GetCollection()
```

Returns

ILiteCollection<T>

## Insert(IEnumerable<T>)

```
public virtual void Insert(IEnumerable<T> entity)
```

Parameters

entity [IEnumerable](#)<T>

## Insert(T)

```
public virtual void Insert(T entity)
```

Parameters

entity T

## Update(IEnumerable<T>)

```
public void Update(IEnumerable<T> entity)
```

### Parameters

entity [IEnumerable](#)<T>

## Update(T)

```
public virtual void Update(T entity)
```

### Parameters

entity T

# Namespace Infrastructure.Services

## Classes

[CategoryService](#)

[JsonStringLocalizer](#)

[LoggingService](#)

[LookupService](#)

[QuestionService](#)

[SourceService](#)

[SystemService](#)

[TagService](#)

# Class CategoryService

Namespace: [Infrastructure.Services](#)

Assembly: Infrastructure.dll

```
public class CategoryService : ICategoryService
```

Inheritance

object ← CategoryService

Implements

[ICategoryService](#)

Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

CategoryService(IRepositoryManager, IAuditManager, IMapper)

```
public CategoryService(IRepositoryManager repositoryManager, IAuditManager auditManager,  
IMapper mapper)
```

Parameters

repositoryManager [IRepositoryManager](#)

auditManager [IAuditManager](#)

mapper IMapper

## Methods

AddCategory(string, string, Guid?)

```
public Result<Category> AddCategory(string name, string description, Guid? parentId)
```

## Parameters

`name` [string](#) ↗

`description` [string](#) ↗

`parentId` [Guid](#) ↗?

## Returns

[Result](#)<[Category](#)>

## UpdateCategory(Guid, string, string)

```
public Result UpdateCategory(Guid categoryId, string name, string description)
```

## Parameters

`categoryId` [Guid](#) ↗

`name` [string](#) ↗

`description` [string](#) ↗

## Returns

[Result](#)

# Class JsonStringLocalizer

Namespace: [Infrastructure.Services](#)

Assembly: Infrastructure.dll

```
public class JsonStringLocalizer : IStringLocalizer
```

## Inheritance

object ← JsonStringLocalizer

## Implements

[IStringLocalizer](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### JsonStringLocalizer(IDistributedCache)

```
public JsonStringLocalizer(IDistributedCache cache)
```

## Parameters

cache [IDistributedCache](#)

## Properties

### this[string]

Gets the string resource with the given name.

```
public LocalizedString this[string name] { get; }
```

## Parameters

**name** [string](#)

The name of the string resource.

## Property Value

[LocalizedString](#)

The string resource as a [LocalizedString](#).

## this[string, object[]]

Gets the string resource with the given name and formatted with the supplied arguments.

```
public LocalizedString this[string name, params object[] arguments] { get; }
```

## Parameters

**name** [string](#)

The name of the string resource.

**arguments** [object\[\]](#)

The values to format the string with.

## Property Value

[LocalizedString](#)

The formatted string resource as a [LocalizedString](#).

# Methods

## GetAllStrings(bool)

Gets all string resources.

```
public IEnumerable<LocalizedString> GetAllStrings(bool includeParentCultures)
```

## Parameters

`includeParentCultures bool`

A [bool](#) indicating whether to include strings from parent cultures.

## Returns

`IEnumerable<LocalizedString>`

The strings.

# Class LoggingService

Namespace: [Infrastructure.Services](#)

Assembly: Infrastructure.dll

```
public class LoggingService : ILogEventSink, IDisposable
```

## Inheritance

object ← LoggingService

## Implements

ILogEventSink, [IDisposable](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### LoggingService(string)

```
public LoggingService(string databasePath)
```

## Parameters

databasePath [string](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## Emit(LogEvent)

Emit the provided log event to the sink.

```
public void Emit(LogEvent logEvent)
```

### Parameters

**logEvent** LogEvent

The log event to write.

### Remarks

Implementers should allow exceptions to propagate when event emission fails. The logger will handle exceptions and produce diagnostics appropriately.

### See Also

[IBatchedLogEventSink](#)

## GetAllLogs()

```
public IEnumerable<ApplicationLog> GetAllLogs()
```

### Returns

[IEnumerable](#)<[ApplicationLog](#)>

# Class LookupService

Namespace: [Infrastructure.Services](#)

Assembly: Infrastructure.dll

```
public class LookupService : ILookupService, IDisposable
```

## Inheritance

object ← LookupService

## Implements

[ILookupService](#), [IDisposable](#) ↗

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### LookupService(IRepositoryManager, IAuditManager)

```
public LookupService(IRepositoryManager repositoryManager, IAuditManager auditManager)
```

## Parameters

repositoryManager [IRepositoryManager](#)

auditManager [IAuditManager](#)

## Methods

### AddLanguage(Language)

```
public void AddLanguage(Language language)
```

## Parameters

language [Language](#)

## AddSource(Source)

```
public void AddSource(Source source)
```

Parameters

source [Source](#)

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## Dispose(bool)

```
protected virtual void Dispose(bool disposing)
```

Parameters

disposing [bool](#)

## GetLanguageReference(string)

```
public Guid GetLanguageReference(string code)
```

Parameters

code [string](#)

Returns

[Guid ↗](#)

## GetLanguages()

```
public IEnumerable<Language> GetLanguages()
```

Returns

[IEnumerable ↗ <Language>](#)

## GetSource(Guid)

```
public Source GetSource(Guid id)
```

Parameters

**id** [Guid ↗](#)

Returns

[Source](#)

## UpdateSource(Source)

```
public void UpdateSource(Source source)
```

Parameters

**source** [Source](#)

# Class QuestionService

Namespace: [Infrastructure.Services](#)

Assembly: Infrastructure.dll

```
public class QuestionService : IQuestionService, IDisposable
```

Inheritance

object ← QuestionService

Implements

[IQuestionService](#), [IDisposable](#) ↗

Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#).

## Constructors

### QuestionService(IQuestionRepository)

```
public QuestionService(IQuestionRepository questionRepository)
```

Parameters

questionRepository [IQuestionRepository](#)

## Methods

### Count()

```
public int Count()
```

Returns

[int](#) ↗

## DeleteAllQuestions()

```
public void DeleteAllQuestions()
```

## DeleteQuestion(Guid)

```
public void DeleteQuestion(Guid id)
```

### Parameters

**id** [Guid](#)

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## Dispose(bool)

```
protected virtual void Dispose(bool disposing)
```

### Parameters

**disposing** [bool](#)

## GetQuestion(Guid)

```
public Question GetQuestion(Guid id)
```

### Parameters

**id** [Guid](#)

Returns

[Question](#)

## GetQuestions(int, int)

```
public IEnumerable<Question> GetQuestions(int pageNumber, int pageSize)
```

Parameters

pageNumber [int](#)

pageSize [int](#)

Returns

[IEnumerable](#) <[Question](#)>

## InsertQuestion(Question)

```
public Question InsertQuestion(Question question)
```

Parameters

question [Question](#)

Returns

[Question](#)

## UpdateQuestion(Question)

```
public void UpdateQuestion(Question question)
```

## Parameters

question [Question](#)

# Class SourceService

Namespace: [Infrastructure.Services](#)

Assembly: Infrastructure.dll

```
public class SourceService : ISourceService
```

## Inheritance

object ← SourceService

## Implements

[ISourceService](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

SourceService(IRepositoryManager, IAuditManager, IMapper)

```
public SourceService(IRepositoryManager repositoryManager, IAuditManager auditManager,  
IMapper mapper)
```

## Parameters

repositoryManager [IRepositoryManager](#)

auditManager [IAuditManager](#)

mapper IMapper

## Methods

AddBulkTag(Guid, IEnumerable<Guid>)

```
public Result<PartialsSuccessResult> AddBulkTag(Guid sourceId, IEnumerable<Guid> tagIds)
```

Parameters

`sourceId` [Guid](#)

`tagIds` [IEnumerable](#)<[Guid](#)>

Returns

[Result](#)<[PartialsSuccessResult](#)>

## AddReference(SourceReference)

```
public void AddReference(SourceReference reference)
```

Parameters

`reference` [SourceReference](#)

## AddReference(IEnumerable<AddSourceReferenceServiceModel>, Guid)

```
public Result<IEnumerable<SourceReference>>
AddReference(IEnumerable<AddSourceReferenceServiceModel> sourceReferences, Guid sourceId)
```

Parameters

`sourceReferences` [IEnumerable](#)<[AddSourceReferenceServiceModel](#)>

`sourceId` [Guid](#)

Returns

[Result](#)<[IEnumerable](#)<[SourceReference](#)>>

## AddSource(AddSourceServiceModel)

```
public Result<Source> AddSource(AddSourceServiceModel model)
```

Parameters

model [AddSourceServiceModel](#)

Returns

[Result<Source>](#)

## AddTag(Guid, Guid)

```
public Result AddTag(Guid sourceId, Guid tagId)
```

Parameters

sourceId [Guid](#)

tagId [Guid](#)

Returns

[Result](#)

## RemoveBulkTags(Guid, IEnumerable<Guid>)

```
public Result<PartialsSuccessResult> RemoveBulkTags(Guid sourceId, IEnumerable<Guid> tagIds)
```

Parameters

sourceId [Guid](#)

tagIds [IEnumerable<Guid>](#)

Returns

[Result<PartialsSuccessResult>](#)

## RemoveTag(Guid, Guid)

```
public Result RemoveTag(Guid sourceId, Guid tagId)
```

### Parameters

sourceId [Guid](#)

tagId [Guid](#)

### Returns

[Result](#)

# Class SystemService

Namespace: [Infrastructure.Services](#)

Assembly: Infrastructure.dll

```
public class SystemService : ISystemService
```

## Inheritance

object ← SystemService

## Implements

[IService](#)

## Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

### SystemService(string)

```
public SystemService(string databasePath)
```

## Parameters

databasePath [string](#) ↗

## Methods

### AnalyzeCollections(int)

```
public List<CollectionStorageStat> AnalyzeCollections(int sampleSize = 10)
```

## Parameters

sampleSize [int](#) ↗

Returns

[List](#) <[CollectionStorageStat](#)>

## GetDatabaseFileSizeBytes()

`public long GetDatabaseFileSizeBytes()`

Returns

[long](#)

# Class TagService

Namespace: [Infrastructure.Services](#)

Assembly: Infrastructure.dll

```
public class TagService : ITagService
```

Inheritance

object ← TagService

Implements

[ITagService](#)

Extension Methods

[ObjectExtention.IsNotNull\(object\)](#) , [ObjectExtention.IsNull\(object\)](#)

## Constructors

TagService(IRepositoryManager, IAuditManager,  
IStringLocalizer<TagService>)

```
public TagService(IRepositoryManager repositoryManager, IAuditManager auditManager,  
IStringLocalizer<TagService> localizer)
```

Parameters

repositoryManager  [IRepositoryManager](#)

auditManager  [IAuditManager](#)

localizer  [IStringLocalizer](#)< [TagService](#)>

## Methods

AddTag(string, string?, string)

```
public Result<Tag> AddTag(string name, string? backgroundColorCode = null, string testColorCode = null)
```

## Parameters

`name` [string](#)

`backgroundColorCode` [string](#)

`testColorCode` [string](#)

## Returns

[Result<Tag>](#)

## ArchiveAllTag()

```
public Result ArchiveAllTag()
```

## Returns

[Result](#)

## ArchiveTag(Guid)

```
public Result ArchiveTag(Guid id)
```

## Parameters

`id` [Guid](#)

## Returns

[Result](#)

## DeleteTag(Guid)

delete a tag and all related entities with a tag.

```
public Result DeleteTag(Guid id)
```

Parameters

[id](#) [Guid](#)

Returns

[Result](#)

## GetAllTags()

```
public IEnumerable<Tag> GetAllTags()
```

Returns

[IEnumerable](#) <[Tag](#)>

## UnArchiveTag(Guid)

```
public Result UnArchiveTag(Guid id)
```

Parameters

[id](#) [Guid](#)

Returns

[Result](#)

## UpdateTag(Guid, string, string, string)

the service will check if there different tag with same name.

```
public Result UpdateTag(Guid id, string name, string backgroundColorCode,  
string textColorCode)
```

## Parameters

**id** [Guid](#)

**name** [string](#)

the new Name

**backgroundColorCode** [string](#)

the color in hex code

**textColorCode** [string](#)

## Returns

[Result](#)