**Feed manager application.**

We have 2 trade feeds, each feed has common sets of fields:

StagingId, SourceTradeRef, CounterpartyId, PrincipalId, ValuationDate, CurrentPrice, SourceAccountId.

But Delta1Feed, and EmFeed have sublte differences:

1. DeltaOneFeed has two additional fields: Isin, MaturityDate
2. EmFeed also has two additional fields to the common set: Sedol, AssetValue

The list of feeds is imported to us from different unverified sources: text files, excel files, other databases. We need first to validate incoming feeds and then, if they are valid, save to the database. Also, if a feed already exists in our db, we should not insert it.

**Task1.**

The first task will be to write two validators: DeltaOneFeedValidator, EmFeedValidator, which will implement validation rules:

Common validation rules for both feeds:

* StagingId, CounterpartyId, PrincipalId, SourceAccountId must be valid non-negative identifiers >= 1.
* CurrentPrice should be a valid price: >= 0 and have only 2 digits after delimiter: 12.34 is valid, 45.6788 is not valid.

There are additional validation rules for Delta1Feed, and EmFeed feeds:

1. DeltaOneFeed.

* Isin should be valid ISIN. Rules are simplified: US9311421039 - It should start with 2 capitals letters and then 10 digits.
* MaturityDate should be bigger than ValuationDate.

1. EmFeed.

* Sedol should be valid decimal bigger than 0 and less than 100.
* AssetValue should be valid decimal bigger than 0 and less than Sedol.

Validators should implement common interface:

public interface IFeedValidator<T>

where T: TradeFeed

{

ValidateResult Validate(T feed);

}

Use ErrorCode class for error messages.

**Task2.**

If feed is validated correctly, we need to match feed with existing feeds stored in the database. We do not use a real database here, The database is substituted with in-memory repository, which will have a method to load all feeds by type and save a feed.

If feed is not valid,

public interface IDatabaseRepository

{

List<T> LoadFeeds<T>();

void SaveFeed<T>(T feed);

void SaveErrors(int feedStagingId, List<String> errors);

}

DeltaOneFeed and EmFeed are matched differently:

1. EmFeed we match by SourceAccountId, if is null in any feed, then we match by StagingId.
2. DeltaOneFeed we match by CouterpartyId+PrincipalId.

Implement two classes DeltaOneFeedMatcher and EmFeedMatcher which implement interface:

public interface IFeedMatcher<T>

where T : TradeFeed

{

bool Match(T current, T other);

}

And, finally, implement import process for each feed:

public class Delta1FeedImporter

{

public void Import(IEnumerable<Delta1Feed> feeds)

{

// TODO: Implement import logic

}

}

public class EmFeedImporter

{

public void Import(IEnumerable<EmFeed> feeds)

{

// TODO: Implement import logic

}

}

The task is to implement import logic for Delta1Feed and EmFeedImporter without semantic copy-paste. Use Factory method pattern.