LINQ Extension Methods

Create a class which implements the following extension methods:

- MySelect method which projects the elements into a new form.
  IEnumerable<TResult> MySelect <TSource, TResult>(this IEnumarable<TSource> source, Func<TSource, TResult> selector)

- MyWhere method filters elements from a collection based on a predicate.
  IEnumerable<TSource> MyWhere<TSource>(this IEnumerable<TSource> source, Func<TSource, bool> predicate)

- MyGroupBy method groups the elements of a sequence according to a specified key selector function.
  IEnumerable<IGrouping<TKey, TSource>> MyGroupBy<TSource, TKey>(this IEnumerable<TSource> source, Func<TSource, TKey> keySelector)

- MyToList which returns a collection as a List.
  List<TSource> MyToList<TSource>(this IEnumerable<TSource> source)

- MyOrderBy sorts the elements of a collection in a particular direction (ascending, descending) according to a key. This method has 2 overloads.
  IOrderedEnumerable<TSource> MyOrderBy<TSource, TKey>(this IEnumerable<TSource> source, Func<TSource, TKey> keySelector)

- MyToDictionary method converts a collection into a dictionary according to a specified key selector function.
  Dictionary<TKey, TSource> MyToDictionary<TSource, TKey>(this IEnumerable<TSource> source, Func<TSource, TKey> keySelector)

Deadline: December 19, 2019, 23:59.