1 Kursusgang 7 - Delegates

1. Define a delegate type named StringJoin with two string parameters and return-type string.
2. Define a method JoinThree parameterized with three strings and a StringJoin, which joins the strings from left to right.
3. Define a method JoinAllStrings, parameterized by a list of strings and a StringJoin, which joins all strings in the list from left to right.
4. Generic Join
   - Define a generic delegate type named *Join* with a single type parameter T, with two parameters and return-type of type T.
   - Similar to exercise 3, but generic, make a *JoinAll* method which can join any list of type T, using the delgate type *Join*
5. Write a generic method Exists(Predicate<T> f, T[] a) that takes a type parameter T and two arguments: a unary lambda expression f and an array a of type T. The method should return true if the array contains an element for which the predicate evaluates to true. Otherwise, it should return false.
6. Write a generic method twice(DELEGATETYPE f, T v) with type parameter T and two arguments: a DELEGATETYPE from the standard library f and a value v of type T. The method should return the result of applying f twice to the argument.