



LINQ part 3

Future Of DotNet

Practice

- `from a in numbersA from b in numbersB where a < b
select (a, b);`
 - 다중 from문의 실행 횟수를 확인
- `from c in customers from o in c.Orders
where o.Total < 500.00M select
(c.CustomerID, o.OrderID, o.Total);`
 - 미국 이외 나라의 1997년에 주문을 했던
거래처와 가장 큰 금액의 주문번호, 날짜, 금액



Partitions

- Take
 - `numbers.Take(3)`
- Skip
 - `numbers.Skip(4)`
- TakeWhile
 - `numbers.TakeWhile(n => n < 6)`
- SkipWhile
 - `numbers.SkipWhile(n => n % 3 != 0)`

Orderings

- orderby
 - orderby word
- Custom Comparer
 - `words.OrderBy(a => a, new CaseInsensitiveComparer())`
- orderby ~ descending
 - orderby d descending
- thenby
 - `orderby digit.Length, digit`
 - `words .OrderBy(a => a.Length) .ThenBy(a => a, new CaseInsensitiveComparer())`
- Reverse
 - `.Reverse()`

Practice

- ```
from c in customers from o in
c.Orders where o.Total < 500.00M
select (c.CustomerID, o.OrderID,
o.Total);
```

  - 미국 이외 나라의 1997년에 주문을 했던  
거래처와 가장 큰 금액의 주문번호, 날짜,  
금액을 출력하는데, 상위 10개만 출력

