开门见山

读这篇文章之前，我先说下，每一种搜索结果集，我都以三种方式变现出来，为啦更好的理解，希望不要嫌我啰嗦。

1. 简单的linq语法

```
//1 var ss = from r in db.Am_recProScheme select r; //2 var ss1 = db.Am_recProScheme; //3 string
sssql = "select * from Am_recProScheme";
```

2. 带where的查询

```
//1 var ss = from r in db.Am_recProScheme where r.rpId > 10 select r; //2 var ss1 =
db.Am_recProScheme.Where(p => p.rpId > 10); //3 string sssql = "select * from Am_recProScheme where
rpid>10";
```

3. 简单的函数计算（count，min，max，sum）

```
//1 ////获取最大的rpId //var ss = (from r in db.Am_recProScheme // select r).Max(p => p.rpId); ////获
取最小的rpId //var ss = (from r in db.Am_recProScheme // select r).Min(p => p.rpId); //获取结果集的总数
//var ss = (from r in db.Am_recProScheme // select r).Count(); //获取rpId的和 var ss = (from r in
db.Am_recProScheme select r).Sum(p => p.rpId); //2 //var ss1 = db.Am_recProScheme.Max(p=>p.rpId);
//var ss1 = db.Am_recProScheme.Min(p => p.rpId); //var ss1 = db.Am_recProScheme.Count() ; var ss1 =
db.Am_recProScheme.Sum(p => p.rpId); Response.Write(ss); //3 string sssql = "select max(rpId) from
Am_recProScheme"; sssql = "select min(rpId) from Am_recProScheme"; sssql = "select count(1) from
Am_recProScheme"; sssql = "select sum(rpId) from Am_recProScheme";
```

4. 排序order by desc/asc

```
var ss = from r in db.Am_recProScheme where r.rpId > 10 orderby r.rpId descending //倒序 // orderby
r.rpId ascending //正序 select r; //正序 var ss1 = db.Am_recProScheme.OrderBy(p => p.rpId).Where(p =>
p.rpId > 10).ToList(); //倒序 var ss2 = db.Am_recProScheme.OrderByDescending(p => p.rpId).Where(p =>
p.rpId > 10).ToList(); string sssql = "select * from Am_recProScheme where rpid>10 order by rpId
[desc|asc]";
```

5. top(1)

```
//如果取最后一个可以按倒叙排列再取值 var ss = (from r in db.Am_recProScheme select r).FirstOrDefault();
// ()linq to ef 好像不支持 Last() var ss1 = db.Am_recProScheme.FirstOrDefault(); //var ss1 =
db.Am_recProScheme.First(); string sssql = "select top(1) * from Am_recProScheme";
```

6. 跳过前面多少条数据取余下的数据

```
//1 var ss = (from r in db.Am_recProScheme orderby r.rpId descending select r).Skip(10); //跳过前10条
数据，取10条之后的所有数据 //2 var ss1 = db.Am_recProScheme.OrderByDescending(p =>
p.rpId).Skip(10).ToList(); //3 string sssql = "select * from (select ROW_NUMBER()over(order by rpId
desc) as rowNum, * from [Am_recProScheme]) as t where rowNum>10";
```

7. 分页数据查询

```
//1 var ss = (from r in db.Am_recProScheme where r.rpId > 10 orderby r.rpId descending select
r).Skip(10).Take(10); //取第11条到第20条数据 //2 Take(10): 数据从开始获取，获取指定数量（10）的连续数据 var
ss1 = db.Am_recProScheme.OrderByDescending(p => p.rpId).Where(p => p.rpId >
10).Skip(10).Take(10).ToList(); //3 string sssql = "select * from (select ROW_NUMBER()over(order by
rpId desc) as rowNum, * from [Am_recProScheme]) as t where rowNum>10 and rowNum<=20";
```

8. 包含，类似like ’%%’

```
//1 var ss = from r in db.Am_recProScheme where r.SortsText.Contains("张") select r; //2 var ss1 =
db.Am_recProScheme.Where(p => p.SortsText.Contains("张")).ToList(); //3 string sssql = "select *
from Am_recProScheme where SortsText like '%张%'";
```

9. 分组group by

```
//1 var ss = from r in db.Am_recProScheme orderby r.rpId descending group r by r.recType into n
select new { n.Key, //这个Key是recType rpId = n.Sum(r => r.rpId), //组内rpId之和 MaxRpId = n.Max(r =>
r.rpId),//组内最大rpId MinRpId = n.Min(r => r.rpId), //组内最小rpId }; foreach (var t in ss) {
Response.Write(t.Key + "--" + t.rpId + "--" + t.MaxRpId + "--" + t.MinRpId); } //2 var ss1 = from r
in db.Am_recProScheme orderby r.rpId descending group r by r.recType into n select n; foreach (var
t in ss1) { Response.Write(t.Key + "--" + t.Min(p => p.rpId)); } //3 var ss2 =
db.Am_recProScheme.GroupBy(p => p.recType); foreach (var t in ss2) { Response.Write(t.Key + "--" +
t.Min(p => p.rpId)); } //4 string sssql = "select recType,min(rpId),max(rpId),sum(rpId) from
Am_recProScheme group by recType";
```

```
//多字段
var result = (from item in data
group item by new { item.Name, item.Type } into items
select new
{
      items.Key.Name,
      items.Key.Type,
      Cnt = items.Count()
}).ToList();

var s = data.GroupBy(p => new { p.Type, p.Name }).Select(p=>new {
```

```
                p.Key.Type,
                p.Key.Name,
                cnt=p.Count()
}).ToList();
```

10.连接查询

```
//1 var ss = from r in db.Am_recProScheme join w in db.Am_Test_Result on r.rpId equals w.rsId
orderby r.rpId descending select r; //2 var ss1 = db.Am_recProScheme.Join(db.Am_Test_Result, p =>
p.rpId, r => r.rsId, (p, r) => p).OrderByDescending(p => p.rpId).ToList(); //3 string sssql =
"select r.* from [Am_recProScheme] as r inner join [dbo].[Am_Test_Result] as t on r.[rpId] = t.
[rsId] order by r.[rpId] desc";
```

11.sql中的In

```
//1 var ss = from p in db.Am_recProScheme where (new int?[] { 24, 25,26 }).Contains(p.rpId) select
p; foreach (var p in ss) { Response.Write(p.Sorts); } //2 string st = "select * from
Am_recProScheme where rpId in(24,25,26)";
```