

<%

```
int totRecords =30; // 총 레코드 수
int pageSize =5; //페이지 사이즈 (데이터 건수 , 한 페이지 보이는 레코드 수 )
int grpSize=5; // 한 그룹에 5개씩 보겠다

int totalPage=0; // 총 페이지수: 전체 페이지수
int totalGrp=0; // 총그룹의 수

int currentGrp =0; // 현재그룹
int currentPage =1;

int grpStartPage = 0; // 현재그룹의 현재시작
int grpEndPage =0; // 재그룹의 마지막

//총 페이지수 구하기
int reamin = totRecords % pageSize ;

if( reamin ==0 )
    totalPage = totRecords / pageSize;

else
    totalPage = totRecords / pageSize +1;

// 현재그룹 구하기
// 현재그룹의 시작번호
// 현재그룹의 끝번호
int remain2 = currentPage % grpSize; // 7 => 그룹2 , 시작 6 , 끝 10

if( remain2 ==0 )
    currentGrp = currentPage / grpSize ; // 10 /5 =>2 , 2번째 그룹

else
    currentGrp = currentPage / grpSize +1; // 4 /5 + 1 => 1번째 그룹 , 7 /5 => 1 +1 => 2그룹

grpStartPage = ( currentGrp -1 ) * grpSize +1 ; // 그룹의 시작번호 현재그룹 1 => 1 , 현재그룹 2 -> 6
grpEndPage = currentGrp * grpSize; // 그룹의 끝번호 1-> 5 , 2-> 10

if( grpEndPage > totalPage){
    grpEndPage = totalPage; // 그룹의 끝번호 10 -> 9로 변경
}

int index = grpStartPage; //현재그룹의 시작번호부터 마지막번호 까지 출력
if( currentGrp >1){
    out.println( "[ 이전 ]" );
}

while( index <= grpEndPage){
    %>
    [ <%=index %> ]
    index ++;
}
if( grpEndPage < totalPage){
    out.println( "[다음]");
}
%>
```

<%

</body>
</html>