**F24**

|  |  |
| --- | --- |
|  | 이번 시간에는 Core Database View CDS에 대해 알아보겠습니다.  CDS는 HANA Database로 전환되면서 데이터 연산 처리과정을 DB계층에 위임하여 정제된 데이터를 가져오는 Code-to-Data 방식입니다.  SAP GUI에서는 생성할 수 없고,  ABAP Development Tool, ADT를 통해서만 생성 할 수가 있습니다. |
|  | CDS 생성방법은 간단합니다.  로컬 혹은 일반 패키지에서 생성할 수 있습니다.  좌측 상단 NEW 클릭 후 Other 클릭한 후 Data definition 입력합니다. |
|  | **행동** |
|  | 지금 뜬 NEW Data definition 창을 통해 CDS DATA SERVICE를 생성할 수 있습니다. |
|  | Data definition은 CDS 정의하기 위한 곳으로 SAP 오브젝트 중에 하나이기 때문에  네이밍 룰은 ‘Z’나 ‘Y’로 동일합니다.  함께 생성해보겠습니다  로컬 패키지에 생성할 것이기 때문에 패키지에 딸라티엠피($TMP)를 적고  네이밍 룰에 따라 네임은 ZConnection\_001\_TMP를 입력합니다.  내역은 flight Connection 이라 적겠습니다. |
|  | **행동** |
|  | NEXT를 누르면 REQUEST가 나오는데 로컬이기에 상관이 없어 다시 한번 눌러줍니다. |
|  | **행동** |
|  | 템플릿이 나옵니다.  이 중 템플릿을 골라 선택하면 편하게 샘플 코드 소스가 생깁니다.  DEFINE VIEW ENTITY를 선택하고  FINISH 버튼을 누르면 이로써 데이터를 추출할 수 있는 CDS View를 만들었습니다 |
|  | **행동** |
|  | Define view entity 옆에 방금 적어준 ZConnection\_001\_TMP 가 적혀 있습니다.  **(드래그)** |
|  | FROM 옆에 ABAP Dictionary Table SPFLI를 적고 Spfli의 필드를 몇 개 적겠습니다.  빠른 진행을 위해 준비해둔 소스를 붙여놓겠습니다. |
|  | **SPFLI**  **// 1 - 1** |
|  | 상단의 액티브를 누르거나 **(가리키기)**  단축키 컨트롤 에프3을 누르면 액티브가 됩니다. 액티브하겠습니다. **(실행)** |
|  | 액티브를 하는 그 순간에 두 가지가 만들어지는데, Data definition(데이터 데피네이션)과 CDS View가 만들어집니다.  CDS View라고 불러도 되고, CDS Entity(엔터티)라고 불러도 됩니다. |
|  | 실행을 하기 위해선 오른쪽 버튼 누르셔서 Open with – Data preview(데이터 프리뷰)에서 데이터를 조회를 할 수가 있습니다. |
|  | **행동 Open with – Data preview** |
|  | 혹은 단축키 F8을 누르셔도 가능합니다. 단축키를 눌러 실행하겠습니다 **(F8)** |
|  | 해당 CDS를 통해서 이렇게 데이터가 조회되는 것을 확인 할 수 있습니다. |
|  | 필터기능에 대해 알아보겠습니다.  만약에 수많은 결과 중에서 내가 아메리칸 에어라인에 대한 데이터만 보고 싶으면  오른쪽 상단에 Add filter(애드필터)에 눌러서 carrid를 선택하시고 |
|  | **행동** |
|  | AA 입력 후 엔터를 치면 원하는 데이터만 필터링을 해서 볼 수가 있습니다. |
|  | 콘솔 기능에 대해 알아보겠습니다. **(필터끄고 가리켜)**  SQL Console(콘솔) 기능을 통해 해당 데이터를 취득하기 위해서 데이터베이스에 어떻게 쿼리가 짜여져있는지 확인해 볼 수 있습니다.  필터 전의 코드는 다음과 같습니다. |
|  | **(원을 그리며)** |
|  | 다시 필터에 조건 값을 넣고 콘솔기능을 실행하면  아까 와는 달리 조건절이 자동으로 들어가 있음을 확인할 수 있습니다.. |
|  | **(지우고 필터 - 콘솔 WHERE 드래그 )** |
|  | 지금까지 CDS VIEW 생성에 대한 기본 개념과 기능에 대해 알아보았습니다. |
|  | 만약 OPEN SQL을 통해 데이터 출력하는 것과 CDS VIEW을 활용한 데이터 출력이 어떠한 차이가 있는지 궁금하실 수 있습니다. 또한 굳이 CDS VIEW을 만들어 사용해야 하는지 의문이 드실 수 있습니다.  가장 큰 차이는 방식의 차이 입니다.  OPEN SQL에는 DATA-TO-CODE로 모든 데이터를 어플리케이션 서버로 가져와 연산처리 과정을 갖지만, CDS VIEW는 데이터베이스 테이블에 직접 접근하여 처리하기 때문에 최종 결과만 가져오는 CODE-TO-DATA 방식입니다.  CDS는 물리적인 세계에 존재하는 데이터베이스와 소프트웨어 세계에 존재하는 프로그램 사이에서 교량 역할을 해준다고 생각하시면 이해하시기 편합니다. |
|  | CDS VIEW를 만드는 두 번째 방법을 알아보겠습니다.  왼쪽을 보면 TREE에 이미 Data definition(데이터 데피네이션) 폴더가 보이실 겁니다. |
|  | **행동** |
|  | 해당 폴더는 첫 번째 방법을 통해 자동으로 생성되었습니다.  해당 폴더에 커서를 대고 마우스 우 클릭하여 New data definition 누릅니다.  빠르게 진행하겠습니다. 이번에는 ZCONNECTION\_002\_TMP로 하겠습니다. |
|  | **행동 ( 다 치고 레퍼런스 오브젝트를 가리키며 )** |
|  | 하단의 Reference Object(레퍼런스 오브젝트)는 CDS를 통해서 데이터를 추출할 TABLE을 입력하면  기본 템플릿을 제공해주기 때문에 조금 더 쉽게 CDS를 구현할 수가 있습니다. |
|  | 이 기능은 옵션으로 꼭 적을 필요 없으나 확인 차 적어보겠습니다. |
|  | **행동 SPLFI 적기** |
|  | 조금 전과 달리 이번에는 Define view라는 템플릿을 사용하도록 하겠습니다. |
|  | **생성 - SQLVIEW 드래그** |
|  | Define view라는 템플릿을 사용할 때에는 Entitiy와 달리 SqlViewName을 반드시 만들어 줘야 됩니다  SqlViewName을 작성할 때는 16자리 크기의 제약이 있으며 이 점을 유의하여 입력합니다.  우리는 Zcon\_002라고 하겠습니다. |
|  | **입력 Zcon\_002** |
|  | **컬럼 드래그 하며** |
|  | 아까와는 달리 레퍼런스 오브젝트를 적용하여 컬럼이 자동으로 불러왔습니다. |
|  | 이제 컨트롤 에프3을 눌러 액티브를 하겠습니다. **(행동)** |
|  | 액티브를 한 순간 DEFINE VIEW는 Data Definition, CDS View, SQL VIEW 총 3가지가 만들어집니다.  Define view 실행 결과는 Entity와 동일합니다. |
|  | **F8** |
|  | **(뷰와 엔티티를 한번에 보여주며)** |
|  | Define View와 Define View Entity의 차이는 다음과 같습니다.  Define View에서는 반드시 SQL View를 만들어줘야 하고  Define View entity에서는 SQL VIEW를 만들지 않아도 됩니다. |
|  | 다음은 SQLVIEW를 조회해보도록 하겠습니다. |
|  | 상단에 오픈 에스에이피 구이 버튼을 누르거나 (가리키기)  단축키 컨트롤 숫자 6을 누르겠습니다. |
|  | **컨트롤 6 엔터 엔터** |
|  | 티코드 SE11을 입력하고  데이터베이스 테이블에 에스큐엘네임을 적어 디스플레이합니다. |
|  | **행동** |
|  | CDS 뷰 내용을 확인하실 수 있습니다. 잘 보시면 MANDT가 포함되어 있습니다. |
|  | 그 다음에 Data Definition 즉, CDS View를 더블클릭해서 이동을 해보면  여기에는 mandt가 포함되어져 있지 않습니다. |
|  | **행동** |
|  | 즉, SqlView과 CDS View가 구조적인 차이를 갖는데, SQLVIEW은 MANDT를 포함하고 CDS VIEW는 포함하지 않습니다. |
|  | 그러면 우리가 프로그램을 통해서 SqlView와 CDS View의 구조적 차이를 확인해보겠습니다.  레포트 프로그램을 만들어 Sqlview와 cdsview를 이용해 인터널 테이블을 선언해보겠습니다. |
|  | 좌측 상단 NEW 클릭 후 ABAP Program을 검색합니다. |
|  | **행동 !!!!!!!!!!!!!!!!!!! 다 적고!!!!!!!!!!!!** |
|  | DATA 를 선언해 보도록 하겠습니다.  시간관계상 미리 준비된 소스를 보겠습니다. |
|  | **// 1 - 2** |
|  | 두가지 큰 차이점은 gt\_conn01은 MANDT 포함, gt\_conn02는 포함하고 있지 않습니다. |
|  | 실행하여 확인해보겠습니다. |
|  | **행동** |
|  | 다음은 Classic Open Sql과 New open sql에 대해 알아보겠습니다. |
|  | **// 1 - 3** |
|  | 클래식 오픈 에스큐엘은 두 가지로 나누었습니다.  첫번째 에스큐엘은 sql view 를 From절에 적었고  두번쨰 에스큐엘은 cds view 를 적었습니다. |
|  | **에러 가리키고** |
|  | 화면을 보시면 2번째는 에러가 일어남을 확인하실 수 있습니다.  그 이유는  Classic open sql from절 뒤에는 CDS View를 사용할 수가 없고 오직 from절에는 SQL View만 사용할 수 있기 때문입니다.  ADT는  버전에 따라 상이하지만 높은 버전을 쓰고 있다면 자동으로 신택스 체크하여 맞는 문법을 제안해줍니다.  지금 이 에러는  Classic open sql from 절에 cds view를 적을 때 New open sql 변경하라고 제안을 하고 있습니다.  따라서 변수 앞에 @을 붙여 문법오류를 해결합니다. |
|  | **@** |
|  | 이번에는 뉴 오픈 에스큐엘 사용법을 알아보겠습니다. |
|  | **// 1 - 4** |
|  | 반면에 new open sql을 사용할 때는 from절의 CDS View를 사용할 수가 있습니다.  물론 new open sql통해서 SQL View도 접근을 할 수는 있지만 바람직하지 않은 방법입니다. |
|  | 정리하자면, new open sql 사용을 권장합니다.  또한 new open sql에서 SQL View 보다 가급적 CDS View를 사용함이 바람직합니다. |
|  | 지금까지 CDS에 대한 기본 개념과 기능, 클래식 오픈 에스큐엘과 뉴 오픈 에스큐엘에서의 차이점에 대해 알아보았습니다.  감사합니다. |