

# 日々是Oracle APEX

Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2022年6月14日 火曜日

## DBAによるAPEXアプリの作成

SQLやリレーショナル・データベースの知識がある程度あると、効率良くAPEXでアプリケーションを作成することができます。

以下のCSV（またはExcel）から、データベースの知識を活用した上で、APEXでアプリケーションを作ってみます。

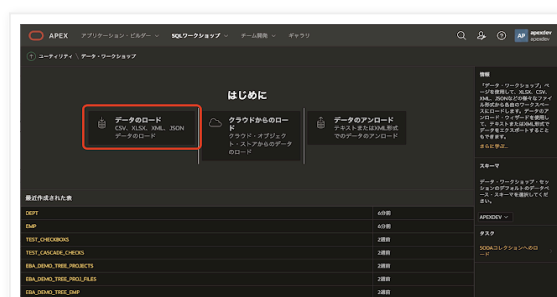
```
"EMPNO","ENAME","JOB","MGR","HIREDATE","SAL","COMM","DEPTNO","DNAME","LOC"
"7839","中島 亜希子","社長","","11/17/1981","5000","","10","会計","東京"
"7782","坂本 明","マネージャー","中島 亜希子","06/09/1981","2450","","10","会計","東京"
"7934","石原 裕美","店員","坂本 明","01/23/1982","1300","","10","会計","東京"
"7566","関口 晃","マネージャー","中島 亜希子","04/02/1981","2975","","20","研究開発","京都"
"7788","新井 敦子","アナリスト","関口 晃","12/09/1982","3000","","20","研究開発","京都"
"7902","石橋 敦","アナリスト","関口 晃","12/03/1981","3000","","20","研究開発","京都"
"7369","村田 淳","店員","石橋 敦","12/17/1980","800","","20","研究開発","京都"
"7876","増田 秀樹","店員","新井 敦子","01/12/1983","1100","","20","研究開発","京都"
"7698","伊藤 明子","マネージャー","中島 亜希子","05/01/1981","2850","","30","セールス","大阪"
"7499","村上 綾子","セールス","伊藤 明子","02/20/1981","1600","300","30","セールス","大阪"
"7521","斉藤 大介","セールス","伊藤 明子","02/22/1981","1250","500","30","セールス","大阪"
"7654","高橋 大輔","セールス","伊藤 明子","09/28/1981","1250","1400","30","セールス","大阪"
"7844","金子 恵美","セールス","伊藤 明子","09/08/1981","1500","0","30","セールス","大阪"
"7900","佐野 英樹","店員","伊藤 明子","12/03/1981","950","","30","セールス","大阪"
```

## データベースにデータをロードする

とりあえず、ファイルのデータをデータベースにロードします。

SQLワークショップのユーティリティのデータ・ワークショップを開きます。

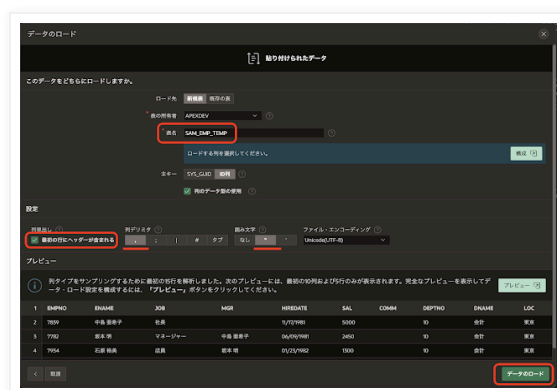
データのロードを実行します。



ファイルを選択するかデータをペーストした後に、データのロード画面が開きます。

データをアップロードする表をSAM\_EMP\_TEMPとします。この表はとりあえずデータを保持するための表です。アプリケーション作成には使用しません。

設定の最初の行にヘッダーが含まれる、列デリミタ、囲み文字やファイル・エンコーディングなどを設定し、データのロードを実行します。



データがロードできたら、データ・ワークショップを使った作業は完了です。

## スキーマを定義する

単純にCSVやExcelをアップロードして作成した表ではなく、データを見てスキーマを定義します。

クイックSQLによるモデルを、以下のように記述しました。

```
# prefix: sam
# ondelete: restrict
employees
  empno num /pk
  ename vc80 /nn
  job_id /fk job
  mgr num /fk employees
  hiredate date
  sal num
  comm num
  deptno /fk dept
```

```
job
  id /pk
  job vc40
```

```
dept
  deptno num /pk
  dname vc80 /nn
  loc vc80
```

以下のDDLが生成されます。クイックSQLを使わず、直接表を作成しても良いでしょう。

```
-- create tables
create table sam_employees (
  empno          number generated by default on null as identity
                constraint sam_employees_empno_pk primary key,
  ename          varchar2(80 char) not null,
  job_id         number
```

```

constraint sam_employees_job_id_fk
references sam_job,
        number
mgr
        constraint sam_employees_mgr_fk
references sam_employees,
        date,
hiredate
        number,
sal
        number,
comm
        number,
deptno
        number
        constraint sam_employees_deptno_fk
references sam_dept
)
;

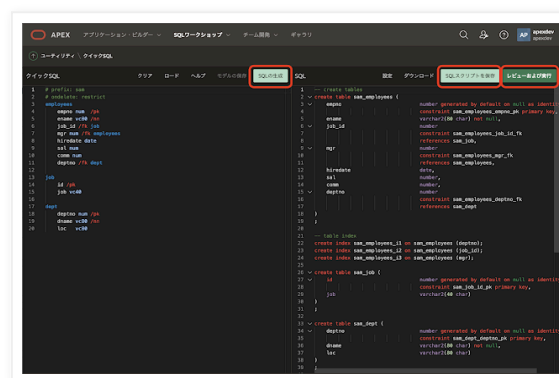
-- table index
create index sam_employees_i1 on sam_employees (deptno);
create index sam_employees_i2 on sam_employees (job_id);
create index sam_employees_i3 on sam_employees (mgr);

create table sam_job (
        id                number generated by default on null as identity
        constraint sam_job_id_pk primary key,
        job                varchar2(40 char)
)
;

create table sam_dept (
        deptno            number generated by default on null as identity
        constraint sam_dept_deptno_pk primary key,
        dname              varchar2(80 char) not null,
        loc                varchar2(80 char)
)
;

```

SQLの生成、SQLスクリプトの保存、レビューおよび実行を順次実施し、最終的上記のDDLを実行して、表SAM\_EMPLOYEES、SAM\_JOB、SAM\_DEPTを作成します。



APEXではアプリケーションを作成した後にスキーマを変更すると、作成済みのアプリケーションに変更を加えるのは手間がかかります。表に列を加えたり減らしたり、という程度であれば容易ですが、表を分割したり、ひとつの表にまとめたりした場合は、作成済みのページを削除して、ウィザードで再作成した方が早い場合もあります。

適切なスキーマを定義するのは、今後の作業にかかる手間が大きく変わるため、とても重要な作業になります。

(上記のクイックSQLから生成されたDDLを実行すると、表SAM\_EMPLOYEESの外部キーを参照している表SAM\_JOB、SAM\_DEPTより先にSAM\_EMPLOYEESを作成しようとするため、エラーが発生します。そのため、生成されたDDLを2回実行する必要があります。)

この時点でもアプリケーションは作成しません。

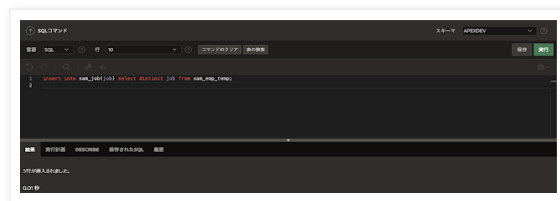
## 作成した表ヘデータを投入する

データをアップロードした表SAM\_EMP\_TEMPより、正規化された表SAM\_EMPLOYEES、SAM\_JOB、SAM\_DEPTへデータを入れ替えます。

SQLワークショップのSQLコマンドから、INSERT文を実行します。

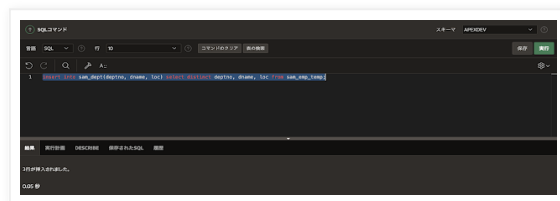
表SAM\_JOBにデータを投入します。

insert into sam\_job(job) select distinct job from sam\_emp\_temp;



表SAM\_DEPTにデータを投入します。

insert into sam\_dept(deptno, dname, loc) select distinct deptno, dname, loc from sam\_emp\_temp;



表SAM\_EMPLOYEESにデータを投入します。

```
insert into sam_employees(empno, ename, job_id, mgr, hiredate, sal, comm, deptno)
select
  t.empno, t.ename, j.id,
  (select m.empno from sam_emp_temp m where m.ename = t.mgr) mgr,
  t.hiredate, t.sal, t.comm, t.deptno
from sam_emp_temp t
join sam_job j on t.job = j.job;
```



以上で、正規化した表にデータの入れ替えができました。

## アプリケーションを作成する

アプリケーションを作成します。

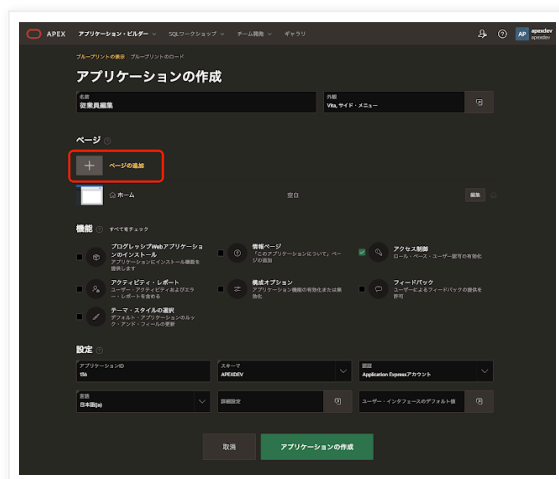
アプリケーション・ビルダーから作成を実行します。



新規アプリケーションを選択します。



名前を従業員編集とします。ページの追加をクリックし、アプリケーションにページを追加します。

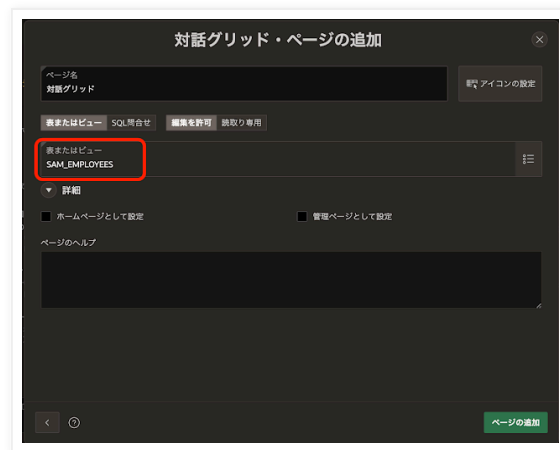


データ編集を目的としているアプリケーションであれば、エンド・ユーザーの利用に向いているページ・タイプは、編集可能対話グリッド、フォーム付き対話モード・レポート、フォーム付きファセット検索、フォーム付きスマート・フィルタの4種類になります。

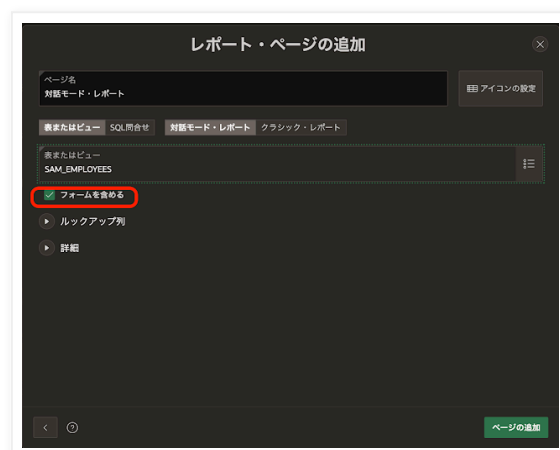


上記の4つのページを全て追加してみます。

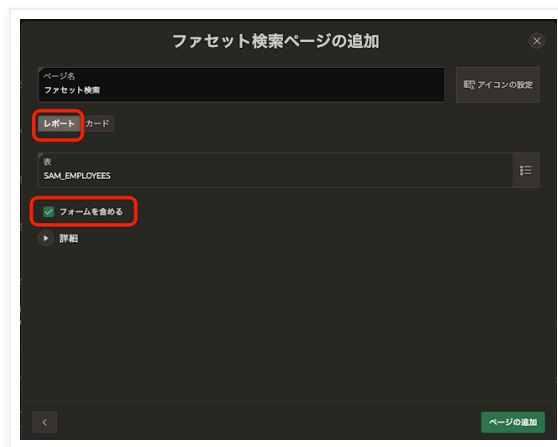
対話グリッドの追加は、以下のようになります。表またはビューとしてSAM\_EMPLOYEESを選択します。



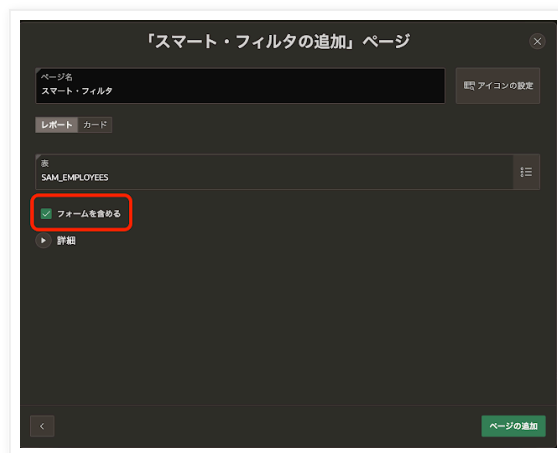
対話モード・レポートでは、フォームを含めるにチェックを入れて、データの編集ページも生成します。



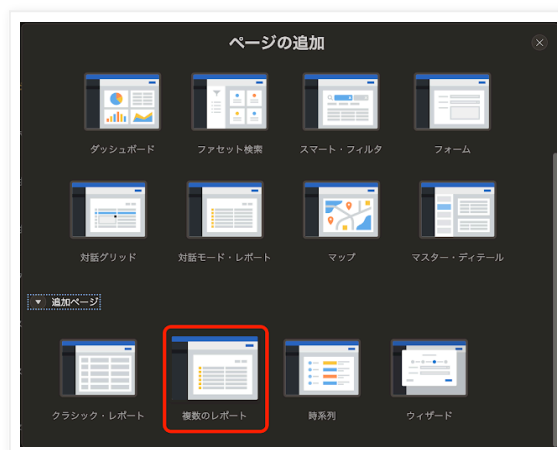
ファセット検索でも、フォームを含めるにチェックを入れます。また、表示形式はカードではなくレポートを選択します。



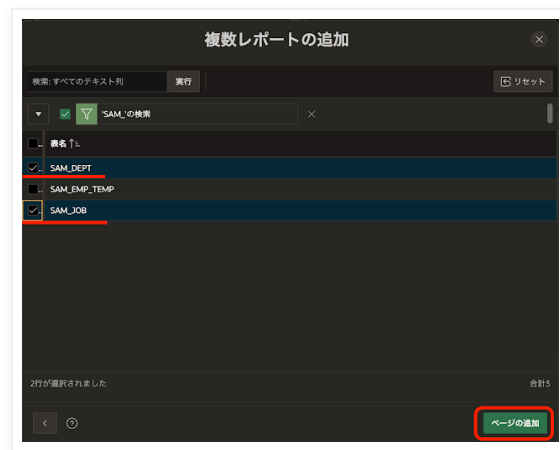
スマート・フィルタはファセット検索とユーザー・インターフェースは異なりますが、実装としては同じであるため、同じ設定にします。



最後に**追加ページ**を開き、マスター・メンテナンスの画面を作るために、**複数のレポート**を追加します。



表名が**SAM\_**で始まる表を検索し、**SAM\_DEPT**と**SAM\_JOB**に**チェック**を入れて**ページの追加**を実行します。



フォーム付きの対話モード・レポートとしてDept、Jobのページが追加されます。

DeptとJobのページはマスター・メンテナンス用なので、編集を開いて管理ページに変更します。



詳細を開いて管理ページとして設定にチェックを入れます。

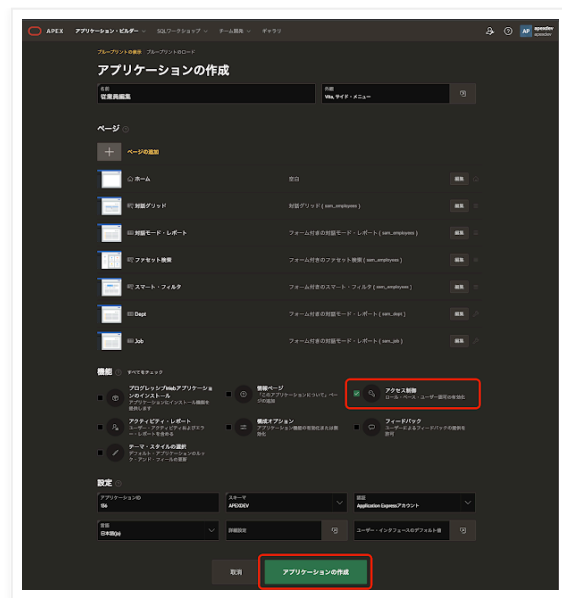
変更を保存します。DeptとJobの両方に実施します。



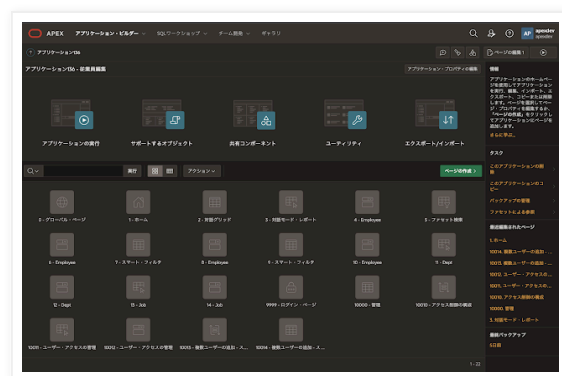
マスター・メンテナンスは管理ページにあります。アクセス制御にチェックを入れて、管理者権限を持っている人だけが、管理ページにアクセスできるようにします。

アプリケーションの作成を実行します。





アプリケーションが出来上がります。



## アプリケーションを実行する

作成されたアプリケーションを実行します。エンドユーザー向けに対話グリッド、対話モード・レポート、ファセット検索、スマート・フィルタのページが作成されています。



対話グリッドを開いてみます。

外部キー制約が付いている列は自動的に選択リストになっています。また、日付（日付ピッカーが開きます）、数値（右寄せになっています）など、データ型も考慮されています。

対話グリッド以外のページも、スキーマ定義が反映されています。

管理を開くと、アプリケーション管理としてマスター・メンテナンスや、アクセス制御の画面へのリンクにアクセスできます。このページにアクセスできるのは、管理者ロールを持ったユーザーに限定されています。

今回は設定しませんでした。が、ユーザー・インターフェース・デフォルトを設定することで、列名のラベルを日本語で設定し直す手間を省くことができます。

ユーザー・インターフェース・デフォルトについては、[こちらの記事](#)で説明しています。ただし、ユーザー・インターフェース・デフォルトは条件によっては適用されないこともあります。

このように、SQLの知識を活用することで、Oracle APEXのアプリケーション作成の効率を上げることができます。

Oracle APEXのアプリケーション作成の参考になれば幸いです。

完

Yuji N. 時刻: 23:14

共有

ウェブ バージョンを表示

自己紹介

Yuji N.

日本オラクル株式会社に勤務していて、Oracle APEXのGroundbreaker Advocateを拝命しました。こちらの記事につきましては、免責事項の参照をお願いいたします。

詳細プロフィールを表示