

Tugas Besar Cara Membuat Aplikasi "Data Hasil Pertanian"

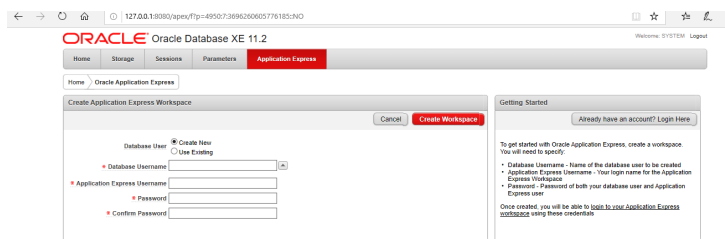
Murnia Lestari D4 Teknik Informatika 2C

18 December 2019

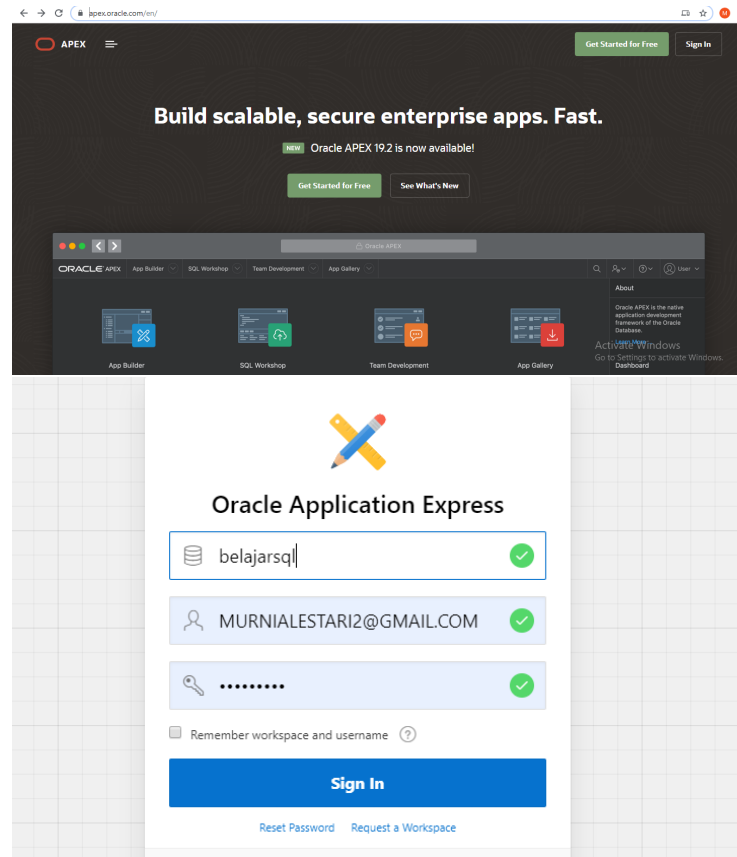
1 Langkah-langkah Membuat Aplikasi

1. Install oracle apex atau kamu juga dapat membuka oracle apex di browser.

Jika kamu menggunakan oracle apex yang online ,maka terlebih dahulu kamu harus menginstall oracle apex lalu ikuti langkah-langkah penginstalan dengan benar. Sehingga aplikasi dapat digunakan dengan baik .Setelah itu kamu akan diminta untuk mengisi login pendaftaran jika kamu belum memiliki akun oracle apex offline. Setelah melalui proses pendaftaran kamu akan masuk ke akun yang telah dibuat dengan memasukkan username dan password yang telah kamu buat. Setelah itu maka buatlah workspace baru dengan cara pilih application express lalu pada database user pilih create new,lalu isi database username,application express username,password,confirm password. Lalu pilih create workspace. Tetapi jika kalian sudah mempunyai workspace maka pilih already have account?login here pada sebelah kanan. Setelah itu maka kalian sudah bisa menggunakan oracle apex untuk membuat query,aplikasi dan lain lain pada oracle apex.



Tetapi jika kamu menggunakan apex online maka search <https://apex.oracle.com/en/>. Pilih sign in jika kamu sudah memiliki workspace .Masukkan nama database dibuat,nama username dan password tetapi jika kamu belum memiliki akun maka pilih request workspace.Lalu ikuti langkah-langkah pada form pendaftaran .Ketika sudah selesai masukkan database,username dan password yang telah dibuat.Jika sudah kamu bisa menggunakan aplikasi tersebut.



2. Jika kamu sudah menginstall aplikasi oracle apex atau sudah mendaftar di oracle apex online dan setelah itu kalian akan masuk ke dalam aplikasi oracle apex.
3. Pilih sql workshop dan pilih sql command. Sql command digunakan untuk mengetikkan query yang akan digunakan.
4. Langkah selanjutnya adalah pembuatan database yang dimulai dengan membuat table. Disini saya membuat aplikasi hasil pangan yang terdapat enam tabel yaitu kecamatan, kelurahan, komoditas, tanaman, lahan, panen. Query yang diinputkan di sql command seperti berikut :

(a) Table kecamatan

Table kecamatan mempunyai 2 kolom yaitu kode kecamatan dan nama kecamatan. Pada table ini yang menjadi primary keynya adalah kode kecamatan.

```
1 CREATE TABLE KECAMATAN
2 (KODE_KECAMATAN INT DEFAULT 0 CONSTRAINT KODE_KECAMATAN_PK PRIMARY KEY,
3 NAMA_KECAMATAN VARCHAR(40) NOT NULL);
```

(b) Table kelurahan

Table kelurahan mempunyai 3 kolom yaitu kode kecamatan, kode kelurahan, dan nama kelurahan. Pada table ini kode kecamatan foreign key dari table kecamatan, dan kode kelurahan merupakan primary key dari table kelurahan.

```
5 CREATE TABLE KELURAHAN
6 (KODE_KECAMATAN INT DEFAULT 0 CONSTRAINT KODE_KECAMATAN_FK REFERENCES KECAMATAN(KODE_KECAMATAN),
7 KODE_KELURAHAN INT DEFAULT 0 CONSTRAINT,
8 NAMA_KELURAHAN VARCHAR(40) NOT NULL);
```

(c) Table komoditas

Table komoditas mempunyai 2 kolom yaitu kode komoditas dan jenis komoditas. Pada table ini kode komoditas dijadikan sebagai primary key.

```
10 CREATE TABLE KOMODITAS
11 (KODE_KOMODITAS INT DEFAULT 0 CONSTRAINT KODE_KOMODITAS_PK PRIMARY KEY,
12 JENIS_KOMODITAS VARCHAR(40) NOT NULL);
```

(d) Table tanaman

Table tanaman mempunyai empat kolom yaitu kode komoditas, kode tanaman, nama tanaman dan panen ke. Pada table ini kode komoditas dijadikan sebagai foreign key dan kode tanaman merupakan primary key dari table tanaman.

```
14 CREATE TABLE TANAMAN
15 (KODE_KOMODITAS INT DEFAULT 0 CONSTRAINT KODE_KOMODITAS_FK REFERENCES KOMODITAS(KODE_KOMODITAS),
16 KODE_TANAMAN INT DEFAULT 0 CONSTRAINT KODE_TANAMAN_PK PRIMARY KEY,
17 NAMA_TANAMAN VARCHAR(40) NOT NULL,
18 PANEN_KE INT DEFAULT 0 );
```

(e) Table lahan

Table lahan mempunyai sembilan kolom yaitu kode lahan, kode kecamatan, kode tanaman, tahun, bulan, luas lahan sawah, luas non sawah, nama pemilik, alamat pemilik. Pada table ini kode kecamatan dan kode tanaman merupakan foreign key dari table kecamatan dan tanaman. Kode lahan pada table ini dijadikan sebagai primary key.

```
20 CREATE TABLE LAHAN
21 (KODE_LAHAN INT DEFAULT 0 CONSTRAINT KODE_LAHAN_PK PRIMARY KEY,
22 KODE_KECAMATAN INT DEFAULT 0 CONSTRAINT KODE_KECAMATAN_FK REFERENCES KECAMATAN(KODE_KECAMATAN),
23 KODE_TANAMAN INT DEFAULT 0 CONSTRAINT KODE_TANAMAN_FK REFERENCES TANAMAN(KODE_TANAMAN),
24 TAHUN INT DEFAULT 0,
25 BULAN VARCHAR (10) NOT NULL,
26 LUAS_LAHAN_SAWAH INT DEFAULT 0,
27 LUAS_NON_SAWAH INT DEFAULT 0,
28 NAMA_PEMILIK VARCHAR(30) NOT NULL,
29 ALAMAT_PEMILIK VARCHAR (50) NOT NULL
30 );
```

(f) Table panen

Table panen mempunyai Sembilan kolom yaitu kode hasil panen, tahun, bulan, kode kelurahan, kode kecamatan, kode tanaman, kode komoditas, kode lahan dan hasil panen. Pada table ini kode kelurahan, kode kecamatan, kode tanaman, kode komoditas, dan kode lahan merupakan foreign key dari table kecamatan, kelurahan, tanaman, komoditas, dan lahan. Kode hasil panen pada table ini merupakan primary key.

```
34 CREATE TABLE PANEN
35 (KODE_HASIL_PANEN INT DEFAULT 0 CONSTRAINT KODE_HASIL_PANEN_PK PRIMARY KEY,
36 TAHUN INT DEFAULT 0,
37 BULAN VARCHAR (10) NOT NULL,
38 KODE_KELURAHAN INT DEFAULT 0 CONSTRAINT KODE_KELURAHAN_FK REFERENCES KELURAHAN(KODE_KELURAHAN),
39 KODE_KECAMATAN INT DEFAULT 0 CONSTRAINT KODE_KECAMATAN_FK REFERENCES KECAMATAN(KODE_KECAMATAN),
40 KODE_TANAMAN INT DEFAULT 0 CONSTRAINT KODE_TANAMAN_FK REFERENCES TANAMAN(KODE_TANAMAN),
41 KODE_KOMODITAS INT DEFAULT 0 CONSTRAINT KODE_KOMODITAS_FK REFERENCES KOMODITAS(KODE_KOMODITAS),
42 KODE_LAHAN INT DEFAULT 0 CONSTRAINT KODE_LAHAN_FK REFERENCES LAHAN(KODE_LAHAN),
43 HASIL_PANEN INT DEFAULT 0
44 );
```

5. Setelah membuat table maka selanjutnya adalah mengisi baris pada kolom table yang telah dibuat. Berikut adalah cara menambahkan baris pada kolom :

(a) Table kecamatan

Pada table kecamatan mempunyai dua kolom maka baris yang dimasukkan juga sebanyak dua baris sesuai dengan jumlah kolom.

```
55 INSERT INTO KECAMATAN VALUES('180310','KOTABUMI SELATAN')
```

(b) Table kelurahan

Pada table kelurahan mempunyai tiga kolom maka baris yang dimasukkan juga sebanyak tiga baris sesuai dengan jumlah kolom, akan tetapi pada kode kecamatan kita masukkan kode kecamatan yang sudah kita buat sebelumnya pada table kecamatan karena kode kecamatan merupakan foreign key yang diambil dari table kecamatan.

```
58 INSERT ALL
59 INTO KELURAHAN VALUES('180310','1803102005','BANDAR PUTIH')
60 INTO KELURAHAN VALUES('180310','1803102007','CURUP GURUH')
61 INTO KELURAHAN VALUES('180310','1803102013','ALAM JAYA')
62 SELECT * FROM DUAL;
```

(c) Table komoditas

Pada table komoditas mempunyai dua kolom maka baris yang dimasukkan juga dua baris sesuai dengan jumlah kolom yang dibuat.

```
55 INSERT INTO KECAMATAN VALUES('180310','KOTABUMI SELATAN')
```

(d) Table tanaman

Pada table tanaman mempunyai tiga kolom maka baris yang akan dimasukkan juga tiga baris sesuai dengan jumlah kolom yang dibuat, akan tetapi kode komoditas merupakan foreign key yang diambil dari table komoditas sehingga isikan baris sesuai yang dibuat pada table komoditas.

```
64 INSERT ALL
65 INTO TANAMAN VALUES('01','0101','PADI')
66 INTO TANAMAN VALUES('01','0102','JAGUNG')
67 SELECT * FROM DUAL
```

(e) Table lahan

Pada table lahan mempunyai sepuluh kolom maka baris yang akan dimasukkan juga sebanyak sepuluh baris sesuai dengan jumlah kolom, akan tetapi perhatikan foreign key yang diambil dari table kelurahan, kecamatan dan juga tanaman harus sesuai dengan yang sudah dimasukkan dalam table sebelumnya.

```
48 INSERT ALL
49 INTO PANEN VALUES('181','2017','JUNI','1803102005','180310','101','1','18031011','1000')
50 INTO PANEN VALUES('182','2017','JULI','1803102007','180310','102','1','18031012','2000')
51 SELECT * FROM DUAL
```

(f) Table panen

Pada table panen mempunyai mempunyai sembilan kolom maka baris yang dimasukkan juga sebanyak Sembilan baris sesuai dengan kolom pada table tersebut,akan tetapi jangan lupa pada table panen terdapat foreign key dari table kecamatan, kelurahan, tanaman, komoditas, serta lahan maka masukkan isi baris sesuai dengan isi baris yang telah dimasukkan pada table selanjutnya.

```
69 INSERT ALL
70 INTO LAHAN VALUES ('18031011','180310','1803102005','101','2017','FEBRUARY','10000','0','BUDIYONO','JALAN MONOGIRI NO 118')
71 INTO LAHAN VALUES ('18031012','180310','1803102007','102','2017','MARET','0','20000','SUJARNO','JALAN BERNIAH RAYA NO 118')
72 SELECT * FROM DUAL
```

6. Setelah column dan baris diisi maka selanjutnya adalah pembuatan fungsi trigger. Fungsi trigger adalah fungsi yang digunakan untuk menampilkan suatu data secara otomatis pada saat insert, delete dan update pada table. Berikut merupakan trigger yang dibuat :

```
71 CREATE OR REPLACE TRIGGER HASIL_PANEN
72 AFTER
73 INSERT OR DELETE OR UPDATE ON PANEN
74 FOR EACH ROW
75 BEGIN
76     IF inserting THEN
77         UPDATE TANAMAN
78         SET PANEN_KE = PANEN_KE+1
79         WHERE KODE_TANAMAN = :new.KODE_TANAMAN;
80     END IF;
81     IF deleting THEN
82         UPDATE TANAMAN
83         SET PANEN_KE = PANEN_KE-1
84         WHERE KODE_TANAMAN = :old.KODE_TANAMAN;
85     END IF;
86 END;
```

```
90 IF updating THEN
91     BEGIN
92         UPDATE TANAMAN
93         SET PANEN_KE = PANEN_KE+1
94         WHERE KODE_TANAMAN = :new.KODE_TANAMAN;
95     END;
96 END IF;
97 END;
```

Pada aplikasi ini saya membuat trigger yang bernama HASIL PANEN, yang mana ketika kita menginsert, mendelete dan mengupdate pada table PANEN maka yang akan mengalami perubahan secara otomatis pada table TANAMAN tepatnya di colum PANEN KE.

7. Setelah itu adalah tahapan membuat view pada aplikasi. View merupakan perintah untuk menggabungkan query join, innerjoin table dengan lebih sederhana.

```
1 CREATE VIEW HASIL_PANEN AS SELECT
2 b.NAMA_TANAMAN, c.HASIL_PANEN, d.JENIS_KOMODITAS
3 FROM TANAMAN b, PANEN c, KOMODITAS d
4 WHERE b.KODE_TANAMAN=c.KODE_TANAMAN
5 AND d.KODE_KOMODITAS=c.KODE_KOMODITAS;
```

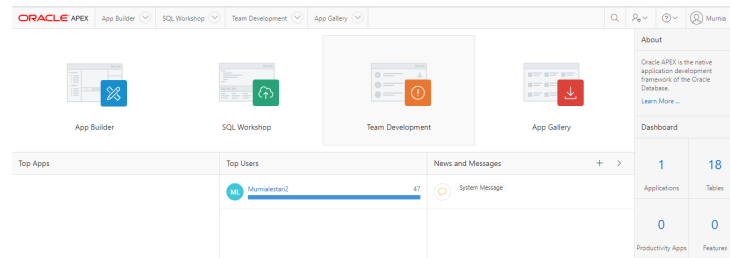
Jadi contoh pada view tersebut akan menampilkan nama tanaman, hasil panen, jenis komoditas dimana nama tanaman, hasil panen, dan jenis komoditas pada table yang berbeda. Jadi view lebih menyederhanakan query yang dimasukkan.

8. Selanjutnya adalah pembuatan sinonim. Sinonim digunakan untuk memanggil nama table dengan nama yang berbeda.

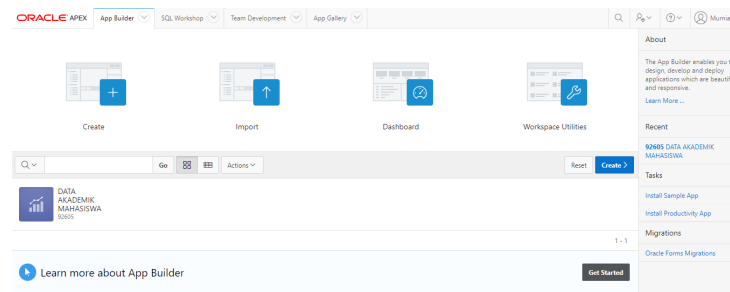
9 CREATE OR REPLACE SYNONYM PN FOR HASIL_PANEN;

Contohnya table HASIL PANEN dapat dipanggil dengan nama PN.

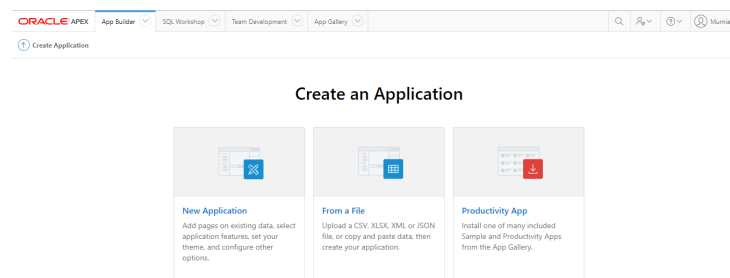
- Setelah semuanya selesai ,maka selanjutnya adalah pembuatan aplikasi dengan memilih app builder.



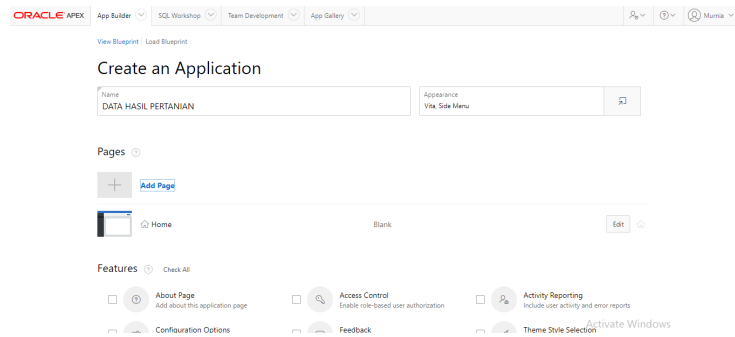
- setelah itu pilih create untuk membuat aplikasi yang baru dengan menggunakan data-data yang telah dibuat di oracle apex.



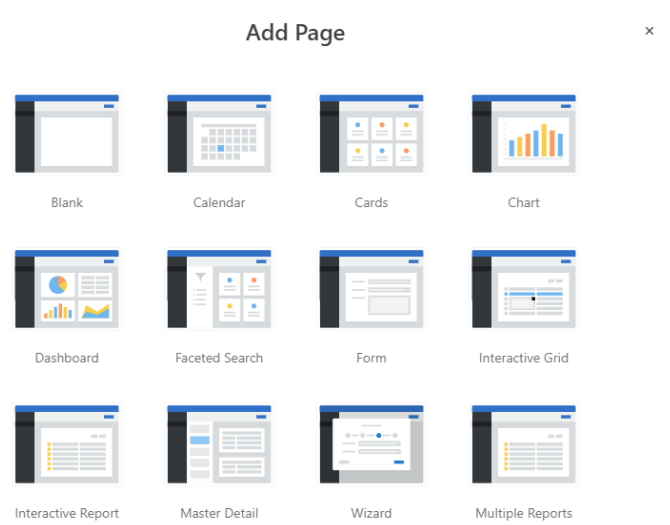
- Selanjutnya pilih new application yang digunakan untuk membuat aplikasi.



- Buatlah nama aplikasi yang akan dibuat,lalu pilih add page untuk memilih page yang akan digunakan untuk membuat aplikasi.



13. Pilih interactive report untuk page aplikasi yang dibuat.



14. Setelah itu isi page namanya dengan kecamatan yang merupakan table pertama pada aplikasi. Pilih table kecamatan pada table or view lalu add page.

Add Report Page ×

Page Name
 KECAMATAN

Set Icon

Table or View
SQL Query
Interactive Report
Classic Report

Table or View
 KECAMATAN

☒ Include Form

Advanced

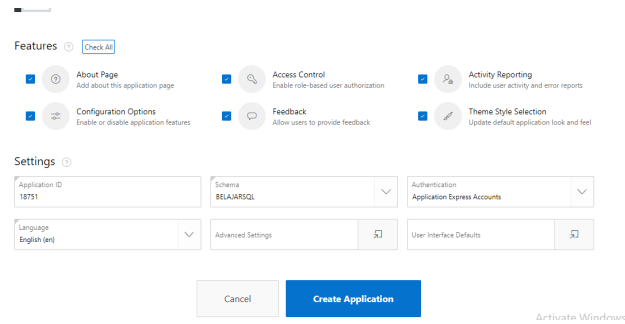
Add Page

15. Lakukan hal yang sama pilih add page dan pilih page intercative report lalu isi name page dengan kelurahan,komoditas,tanaman,lahan dan panen secara bergantian. Lalu pilih table atau view table yang yang telah dibuat yaitu kelurahan,komoditas,tanaman,lahan,dan panen juga secara bergantian. Jangan lupa jika ingin menambahkan view yang telah dibuat dengan nama hasil panen pada page name lalu pilih view yang telah dibuat yaitu hasil panen. Lalu add page.

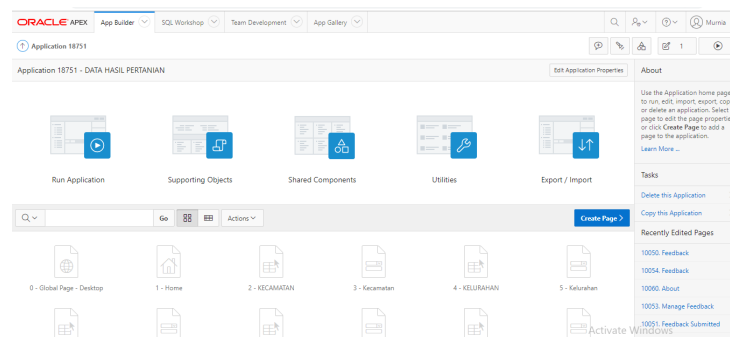
Icon	Page Name	Description	Edit
	KECAMATAN	Interactive Report with Form (kecamatan)	
	KEURAHAN	Interactive Report with Form (kelurahan)	
	KOMODITAS	Interactive Report with Form (komoditas)	
	TANAMAN	Interactive Report with Form (tanaman)	
	LAHAN	Interactive Report with Form (lahan)	
	PANEN	Interactive Report with Form (panen)	
	PANEN	Interactive Report with Form (panen)	
	HASIL PANEN	Interactive Report (hasil_panen)	

Activate Windows

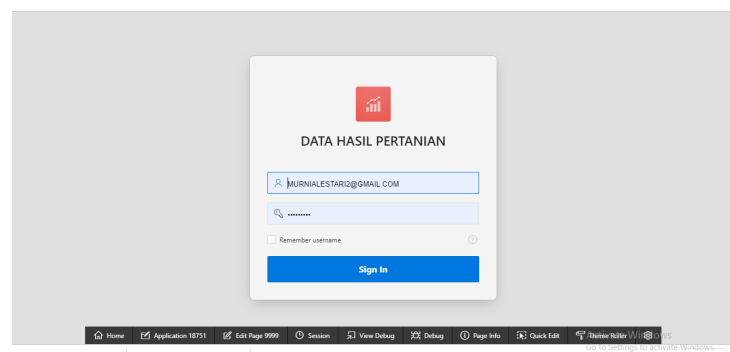
16. Setelah semuanya di add page maka pada features di cek all. Maka setelah itu create application. Tunggu hingga selesai.



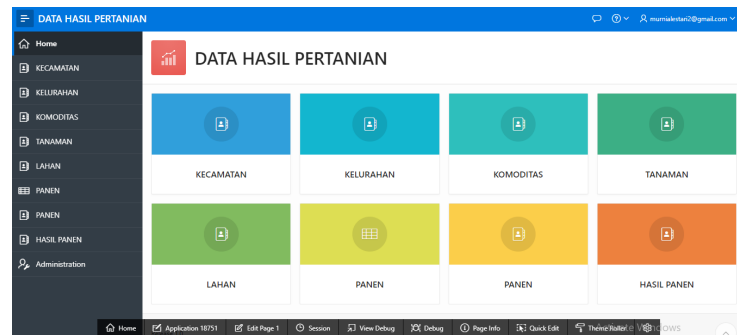
17. Setelah itu pilih run application untuk melihat aplikasi yang sudah dibuat.



18. Dan masukkan username dan password untuk melihat aplikasi yang telah dibuat. Lalu sign in.



19. Maka aplikasi data hasil pertanian sudah selesai dibuat.



listings https://apex.oracle.com/pls/apex/f?p=18751:LOGIN_DESKTOP:703118311964605:::::
User Name : MURNIALESTARI2@GMAIL.COM PASSWORD : murnia123