

Profile Mahasiswa

Heriyanto

December 2019

1 Membuat aplikasi melalui apex.oracle

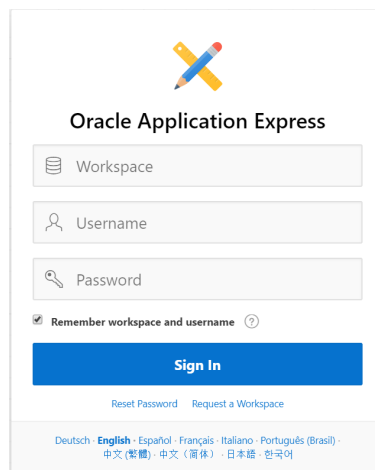
1.1 Apex Oracle

Application Express(Oracle Apex), dulu di sebut HTTP DB adalah sebuah Aplikasi database berbasis web yang digunakan dalam database Oracle. Oracle Apex ini mudah dipakai hanya menggunakan web browser saja dan proses pemrograman yang sederhana, serta mengembangkan ilmu dan kemampuan dengan Aplikasi ini secara aman dan cepat. Oracle Application Express memiliki kualitas database yang bagus, produktivitas dan standart luas yang dimiliki oleh perusahaan-perusahaan besar dan memiliki keamanan, stabilitas, ketersediaan dalam membangun suatu web. Application Express adalah suatu alat yang digunakan untuk membangun Aplikasi web-based. Selain itu Oracle Application Express tidak membutuhkan perangkat lainnya untuk mengembangkan, menyebarkan serta menjalankan Aplikasi, karena di dalam Oracle Application Express telah menyediakan tiga alat utama. Tiga alat utama ini memiliki Kegunaan yang penting di dalam Oracle Application Express:

- Application Builder : Membuat aplikasi, melihat aplikasi, mengimport aplikasi, mengatur service, mengatur user aplikasi dan memantau aktifitas yang di lakukan pengguna.
- SQL Workshop : Membuat tabel dan komponennya (menggunakan kode PL-SQL secara manual maupun otomatis), melihat struktur tabel dan komponennya, mengimpor dan mengekspor script.
- Utilitas : melihat report table dan komponennya dan history aplikasi.

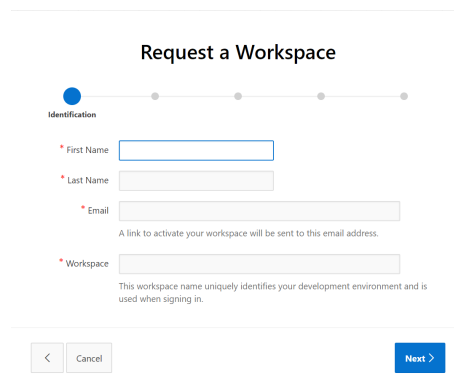
2 Langkah - Langkah Membuat Aplikasi di Apex Oracle

1. Pertama login ke apex.oracle.com, lalu klik Sign in
2. Akan dialihkan ke halaman login application express, silahkan masuk untuk pengguna yang telah memiliki workspace, jika belum memiliki workspace silahkan klik "Request a Workspace".



The screenshot shows the Oracle Application Express login interface. At the top is the Oracle logo (a stylized 'X' with orange and blue colors). Below it, the text "Oracle Application Express" is displayed. The login form includes a "Workspace" dropdown menu, a "Username" input field with a person icon, and a "Password" input field with a key icon. There is a checkbox labeled "Remember workspace and username" with a help icon. A prominent blue "Sign In" button is located below the password field. At the bottom of the form, there are links for "Reset Password" and "Request a Workspace". The footer contains a list of languages: Deutsch, English, Español, Français, Italiano, Português (Brasil), 中文 (繁體), 中文 (简体), 日本語, and 한국어.

3. Cara membuat workspace cukup mudah, silahkan isi form yang telah disediakan setelah meminta workspace, lalu ikuti arahan selanjutnya.



The screenshot displays the "Request a Workspace" form. At the top, the title "Request a Workspace" is centered. Below the title is a progress indicator with four dots; the first dot is blue, indicating the current step. The form is divided into sections by a horizontal line. The first section is labeled "Identification" and contains three required fields: "First Name", "Last Name", and "Email". Below the "Email" field, a note states: "A link to activate your workspace will be sent to this email address." The second section is labeled "Workspace" and contains a required text field. Below this field, a note states: "This workspace name uniquely identifies your development environment and is used when signing in." At the bottom of the form, there are three buttons: a back button with a left arrow, a "Cancel" button, and a "Next >" button.

4. Jika telah memiliki workspace silahkan login dengan menggunakan workspace tersebut dan username dan password yang anda buat.
5. Setelah login, di halaman utama klik "SQL Workshop", lalu pilih dan klik "SQL Commands".
6. Di SQL Commands ini kita akan membuat tabel terlebih dahulu. Tabel pertama adalah Tabel Mahasiswa. Ikuti sintaks dibawah:

```
1 CREATE TABLE "MAHASISWA"
2   (    "NIM" VARCHAR2(10),
3     "NAMA" VARCHAR2(30) NOT NULL ENABLE,
4     "ALAMAT" VARCHAR2(100) NOT NULL ENABLE,
5     "TGL_LAHIR" DATE NOT NULL ENABLE,
6     "ANGKATAN" NUMBER NOT NULL ENABLE,
7     "JURUSAN" VARCHAR2(20) NOT NULL ENABLE,
8     CONSTRAINT "PK_NIM" PRIMARY KEY ("NIM")
9   )
```

7. Tabel selanjutnya adalah Tabel Matkul.

```
1 CREATE TABLE "MATKUL"
2   (    "KODE_MATKUL" VARCHAR2(7) NOT NULL ENABLE,
3     "NAMA_MATKUL" VARCHAR2(20) NOT NULL ENABLE,
4     "SKS" NUMBER NOT NULL ENABLE,
5     CONSTRAINT "PK_KODEMATKUL" PRIMARY KEY ("KODE_MATKUL")
6   )
7
```

8. Tabel ketiga adalah tabel Nilaimhs.

```
1 CREATE TABLE "NILAI_MHS"
2   (    "NIM" VARCHAR2(10) NOT NULL ENABLE,
3     "KODE_MATKUL" VARCHAR2(7) NOT NULL ENABLE,
4     "NILAI" VARCHAR2(2) NOT NULL ENABLE,
5     "TGL" DATE
6   )
7
```

9. Terakhir adalah tabel Logmhs.

```

1 CREATE TABLE "LOG_MHS"
2 (
3     "NIM" VARCHAR2(10) NOT NULL ENABLE,
4     "NAMA" VARCHAR2(30) NOT NULL ENABLE,
5     "JURUSAN" VARCHAR2(30) NOT NULL ENABLE,
6     "ANGKATAN" NUMBER NOT NULL ENABLE,
7     "CHANGED_AT" TIMESTAMP (6),
8     "KETERANGAN" VARCHAR2(10) NOT NULL ENABLE
9 )

```

10. Setelah semua tabel telah dibuat, sekarang kita akan membuat trigger. Trigger adalah blok PL/SQL yang disimpan dalam database dan akan diaktivasi ketika kita melakukan statement-statement SQL (DELETE, UPDATE, dan INSERT) pada sebuah tabel. Aktivasi trigger didasarkan pada event yang terjadi di dalam tabel tersebut sehingga trigger dapat membantu dalam menjaga integritas dan konsistensi data. Implementasi trigger yang sering ditemui dalam dunia nyata adalah untuk meng-set dan mengubah nilai kolom dalam suatu tabel sehingga validasi nilai dari tabel tersebut akan terjaga. Adanya trigger dalam database akan meringankan kita dalam pembuatan aplikasi karena di dalam aplikasi yang kita buat, kita tidak perlu lagi untuk melakukan validasi data.
11. Disini kita akan membuat after insert dan before insert. Pertama mari kita buat trigger after delete, ikuti sintaks dibawah:

```

1 CREATE OR REPLACE TRIGGER "TR_DELETE_MHS"
2 AFTER DELETE ON mahasiswa
3 FOR EACH ROW
4 BEGIN
5     INSERT INTO log_mhs VALUES (
6         :OLD.NIM,
7         :OLD.NAMA,
8         :OLD.JURUSAN,
9         :OLD.ANGKATAN,
10        CURRENT_TIMESTAMP,
11        'DELETED'
12    );
13 END;

```

12. Sekarang kita akan memasukkan data ke tabel-tabel tersebut secara manual, kecuali tabel logmhs yang datanya akan masuk secara otoma-

tis karena trigger after insert yang telah kita buat tadi. Ketika ada perubahan data pada tabel mahasiswa akan langsung dimasukkan ke tabel logmhs. Sekarang kita coba menambah data di tabel mahasiswa.

```
1 insert all
2 into mahasiswa values ('1184023', 'Heriyanto', 'Jl.Jend.Sudirman', '04/26/2000', 2018, 'DIV TI')
3 into mahasiswa values ('1184098', 'Dyah Ayu Anandra', 'Jl. Sariasih', '12/04/2000', 2018, 'DIV TI')
4 into mahasiswa values ('1184085', 'Nuha Hanifatul', 'Sarimarah', '05/30/1999', 2018, 'DIV TI')
5 select * from dual;
```

13. Bisa dilihat dibawah bahwa data di logmhs telah ditambahkan setelah menambahkan data di tabel mahasiswa.

```
select * from log_mhs;
```

Results	Explain	Describe	Saved SQL	History		
1184023		Heriyanto	DIV TI	2018	18-DEC-19 04:28:30.461745 PM	ADDED
1184098		Dyah Ayu Anandra	DIV TI	2018	18-DEC-19 04:28:30.463625 PM	ADDED
1184085		Nuha Hanifatul	DIV TI	2018	18-DEC-19 04:28:30.463715 PM	ADDED

heriyanto@gmail.com | dslabz | en | Copyright © 1999, 2019, Oracle. All rights reserved. | Application Express 19.2.0.0.18

14. Tak lupa untuk memasukkan data di tabel matkul.

```
1 insert all
2 into matkul values ('PPI1234', 'General English III', '2')
3 into matkul values ('TI43203', 'Jaringan Komputer', '3')
4 into matkul values ('TI43193', 'Basis Data II', '4')
5 into matkul values ('TI41183', 'Pemrograman', '4')
6 select * from dual;
```

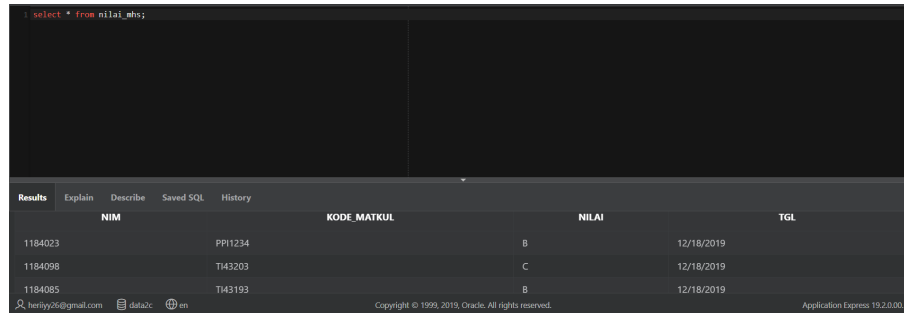
15. Selanjutnya adalah membuat trigger before insert pada tabel nilai_mhs.

```
1 CREATE OR REPLACE TRIGGER "TR_INSERT_NILAI_MHS"
2 BEFORE INSERT ON nilai_mhs
3 FOR EACH ROW
4 BEGIN
5     SELECT SYSDATE INTO :NEW.TGL FROM DUAL;
6 END;
```

16. Sekarang kita akan memasukkan data ke tabel nilaimhs tapi hanya untuk kolom nim, kodematkul, dan nilai saja. Kenapa tgl tidak, itu karena kita telah sebelumnya membuat trigger, kolom tgl akan terisi otomatis. Coba kita tambahkan data.

```
1 insert all
2 into nilai_mhs (nim, kode_matkul, nilai) values ('1184023', 'PPI1234', 'B')
3 into nilai_mhs (nim, kode_matkul, nilai) values ('1184098', 'TI43203', 'C')
4 into nilai_mhs (nim, kode_matkul, nilai) values ('1184085', 'TI43193', 'B')
5 select * from dual;
```

17. Kita lihat bahwa trigger-nya berhasil, kolom tgl terisi dengan hari bulan dan tahun kita menambahkan data.

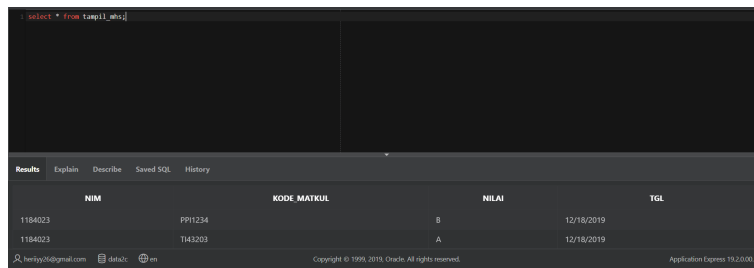


NIM	KODE_MATKUL	NILAI	TGL
1184023	PPI1234	B	12/18/2019
1184098	TI43203	C	12/18/2019
1184085	TI43193	B	12/18/2019

18. Sekarang kita akan membuat view untuk table nilaimhs dimana akan menampilkan nim = 1184023. View dapat didefinisikan sebagai ‘tabel virtual’. Tabel ini bisa berasal dari tabel lain, atau gabungan dari beberapa tabel. Tujuan dari pembuatan VIEW adalah untuk kenyamanan (mempermudah penulisan query), untuk keamanan (menyembunyikan beberapa kolom yang bersifat rahasia), atau dalam beberapa kasus bisa digunakan untuk mempercepat proses menampilkan data (terutama jika kita akan menjalankan query tersebut secara berulang).

```
1 CREATE OR REPLACE VIEW tampil_mhs AS SELECT * FROM nilai_mhs WHERE nim = 1184023;
2
3
```

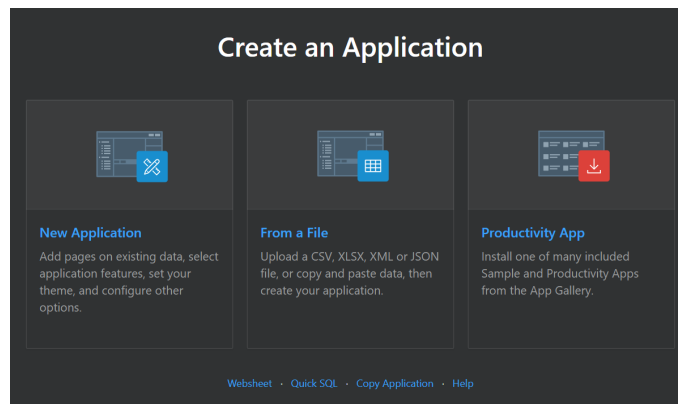
19. Untuk melihat hasil dari view dengan cara select * from nama view.



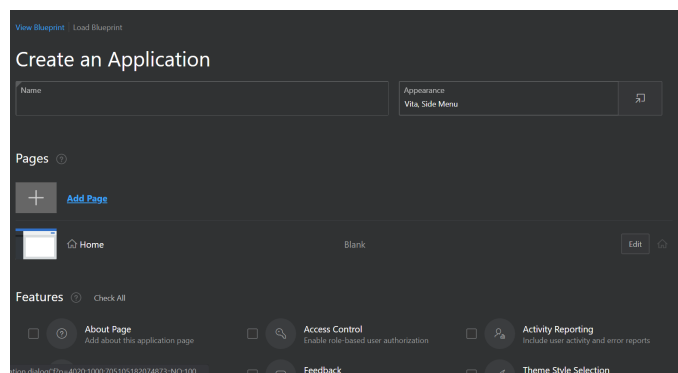
```
select * from sample11_ahs;
```

NIM	KODE_MATEKUL	NILAI	TGL
1184023	PP11234	B	12/18/2019
1184023	TI43203	A	12/18/2019

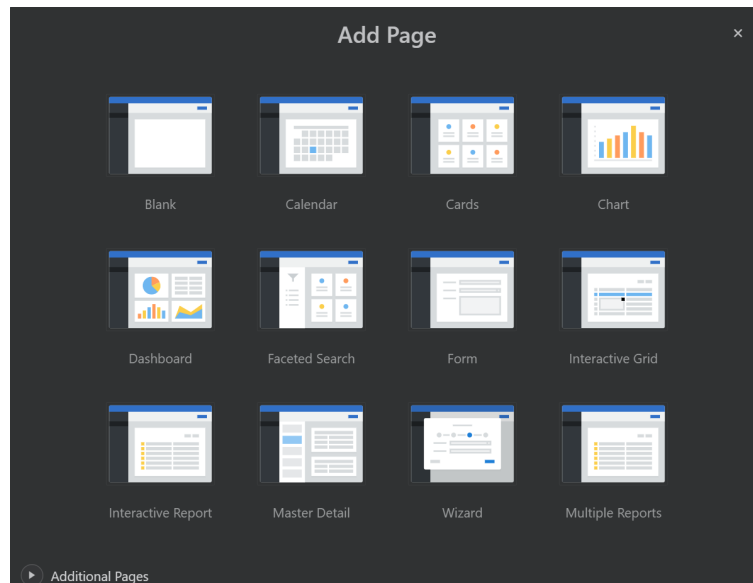
20. Selanjutnya pembuatan aplikasi Profile Mahasiswa. Caranya adalah klik "App Builder" lalu klik "Create".
21. Setelah itu pilih "New Application".



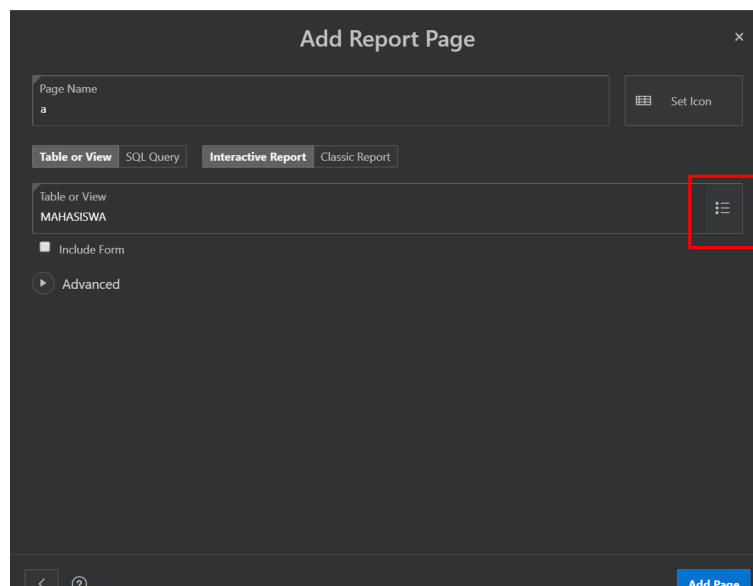
22. Lalu isi form nama aplikasi dan tambahkan menu pada aplikasi dengan klik "Add Page".



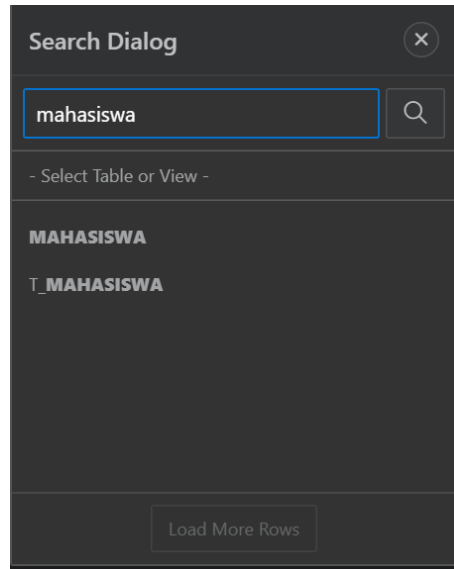
23. Pada add page pilih "Interactive Report".



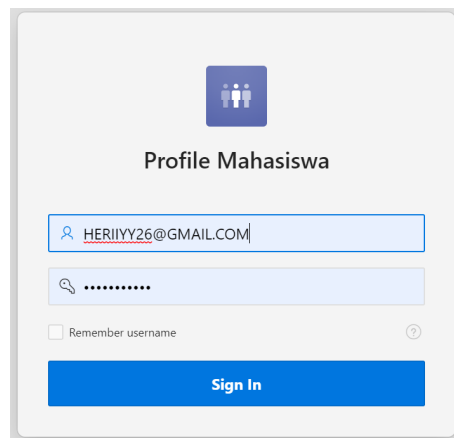
24. Lalu kita mengisi form menu dan menambahkan tabel, klik pada tanda kotak merah di gambar untuk menambahkan tabel.



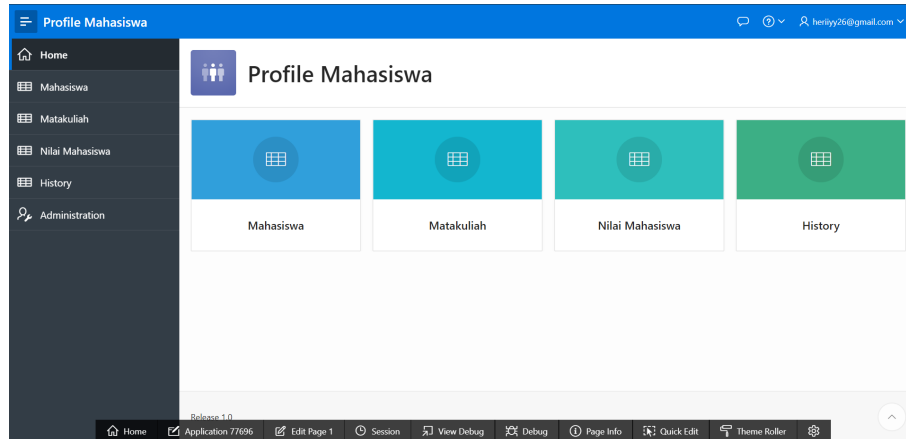
25. Silahkan cari nama tabel mahasiswa, dan lakukan hal yang sama untuk tabel matkul, tabel nilaimhs, dan tabel logmhs.



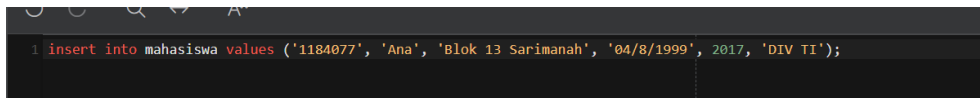
26. Jika menu sudah ditambahkan sekarang tinggal menekan "Create Application" dan tunggu beberapa saat.
27. Setelah aplikasi dibuat jalankan aplikasi tersebut dengan klik "Run Application". Dan masuk dengan username dan password yang sama dengan waktu membuat workspace.



28. Tampilan Dashboard Aplikasi.



29. Kita coba trigger after insert.



30. Data berhasil dimasukkan otomatis.

History

Nim ↑	Nama	Jurusan	Angkatan	Changed At	Keterangan
1184023	Heriyanto	DIV TI	2,018	12/18/2019	ADDED
1184023	Heriyanto	D4TI	2,018	12/18/2019	ADDED
1184023	Heriyanto	DIV TI	2,018	12/18/2019	ADDED
1184055	Ardhy	DIV TI	2,017	12/18/2019	ADDED
1184077	Ana	DIV TI	2,017	12/18/2019	ADDED
1184085	Nuha Hanifatul	DIV TI	2,018	12/18/2019	ADDED
1184098	Dyah Ayu Anandra	DIV TI	2,018	12/18/2019	ADDED

31. Kita coba juga before insert.

```
1 insert into nilai_mhs (nim, kode_matkul, nilai) values ('1184055', 'TI43193', 'B');
```

32. Kolom tgl berhasil terisi otomatis berdasarkan waktu sekrang.

Nilai Mahasiswa

Nim ↑	Kode Matkul	Nilai	Tgl
Ardhy	Basis Data II	B	12/18/2019
Dyah Ayu Anandra	Jaringan Komputer	C	12/18/2019
Heriyanto	General English III	B	12/18/2019
Heriyanto	Jaringan Komputer	A	12/18/2019
Nuha Hanifatul	Basis Data II	B	12/18/2019

1 - 5

3 Link

1. Link = <https://apex.oracle.com/pls/apex/f?p=77696:1:114506129892151:::>
2. ID = heriiyy26@gmail.com
3. PASS = makioshi194