

Ebook II:

Bot Telegram PHP dan DATABASE (SQL)



Studi Kasus:

Bot **"Catatan Si Bot"**

oleh:

bang Hasan (@hasanudinhs)

diskusi: @botphp

Daftar Isi

Cover	1
Daftar Isi	2
Pendahuluan	3
Mendapatkan File Materi	4
System Environment	5
Database	6
SQLite	7
Studi Kasus	8
Bot Diary, Catatan si Bot	9
Struktur Data Bot	11
SQLite Browser	12
Membuat Database	13
Membuat Table	14
Medoo, Database PHP Framework	15
Water Test, Persiapan Mulai	17
Struktur File dan Folder Bot	18
Hubungi Bapak Bot	19
Database Engine Bot	20
Engine Utama, diary.php	24
Clone Proyek	27
Diary Si Bot	28
Pengenalan Bot Diary	29
Pekerjaan Rumah Belum Tuntas	31
Penutup, Kado Bulan Besar	32
Tentang Penulis	33
Referensi	35
Disclaimer	37
Matur nuwun	38



Pendahuluan

Ini adalah penulisan buku ke-2, lanjutan dari Seri Ebook Bot Telegram, **Edisi Pengantar “Membuat Sendiri Bot Telegram dengan PHP”**.

Sebaiknya, sebelum membaca atau melanjutkan ebook ini sudah memahami dan sudah mempraktekkan ebook pertama tersebut.

Edisi yang sedang dibaca ini akan membahas materi tentang bot php lanjutan dengan memperkenalkan database sebagai pelengkap.

Usai pembahasan materi Ebook yang Pertama dahulu, ternyata banyak yang bertanya terlebih soal bot telegram yang dibuat dengan mengintegrasikan database.

Sebenarnya integrasi dengan database, adalah materi umum yang bisa didapat dengan belajar online tentang php dan database. Selayaknya php untuk web, tidak jauh beda. Begitu kira-kira.

Namun, beberapa pembaca tetap mendorong untuk mewujudkan tutorial atau contoh yang bisa langsung dipakai untuk bot telegram. Maka ebook ke-2 ini kiranya perlu untuk ditulis.

Spesial di ebook ini, pertama kalinya penulis mencoba menggunakan software Kingsoft Office, WPS Writer edisi Free Community dalam hal penulisan serius. Biasanya gak serius? *Ciyus* juga :-D

Hanya saja ternyata menulis pake WPS Writer untuk sebuah buku yang menggunakan beberapa *layout* sekaligus, butuh effort lebih dibandingkan office yang sudah biasa dipakai. Kesulitan terasa pertama kali adalah beberapa format masih kurang konsisten ketika ada perubahan. Namun, harus diwajarkan hati, hal baru memang sudah biasa membuat bingung..

Metode awam saja, tulis dulu idenya biar tidak melayang.. Baru nanti diotak-atik lagi, betul? Mari kita garuk-garuk aspal :D

Ok, let's go!

Mendapatkan File Materi

Tips & Trik: Gedebug Bot



Mendapatkan Materi Ebook

Private Message, atau Japri ke bot @gedebugbot :

1. Ebook I - Membuat Bot TELEGRAM dari PHP (ed 3).pdf

Ketikkan: !file BQADBQADewADksi6DLeM8q8Q_OL7Ag

2. Ebook II - Bot Telegram PHP dan Database SQL.pdf

Ketikkan: !File BQADBQADXbMBAAEIVs0FCaioMkwcQlAC

System Environment



Tidak ada batasan khusus untuk menggunakan sistem environment pada ebook ini, karena anggapan secara umum adalah sama seperti pada ebook yang pertama.

Namun kebutuhan wajib untuk dapat mengikuti ebook ini, adalah :

- ≈ PHP, minimal versi 5.4
- ≈ Modul tambahan atau ekstensi PHP nya, yakni: php cli, php curl, php ado, dan php sqlite

Untuk Operating System, dipersilakan menggunakan masing-masing. Namun, penulis tentu saja merekomendasikan seperti yang penulis pakai. Agar mudah dan tentu saja nyaman untuk developing.

Berikut yang dipakai penulis, mungkin ada yang berkenan menconteknya:

- ≈ Sistem Operasi: menggunakan Linux, tepatnya Distro **PeppermintOS**, turunan dari ubuntu. Lebih ringan, nyaman, dan tentu saja free :)
- ≈ PHP yang penulis gunakan adalah **PHP versi 7**, lengkap dengan modul ekstensi minimal di atas.
- ≈ Pengelolaan database secara GUI menggunakan SQLite Manager dan atau **DB Browser for SQLite**.

Developer tools dan pelengkap lainnya: git, mc, tmux, fping, dll.

Database



Pengertian database adalah sekumpulan data yang sudah disusun sedemikian rupa dengan ketentuan atau aturan tertentu yang saling berelasi sehingga memudahkan pengguna dalam mengelolanya juga memudahkan memperoleh informasi. Selain itu adapula yang mendefinisikan database sebagai kumpulan file, tabel, atau arsip yang saling terhubung yang disimpan dalam media elektronik.¹

Contoh-contoh software Database:

Microsoft SQL Server, Oracle, Sybase, Interbase, XBase, Firebird, MySQL, PostgreSQL, Microsoft Access, dBase III, Paradox, FoxPro, Visual FoxPro, Arago, Force, Recital, dbFast, dbXL, Quicksilver, Clipper, dlsb².

Itu di atas kutipan alias nyuplik, agar dalam ebook ini gak berasa loncat kehilangan bab penjelasan database hehe..

Dirasa hanya cukup segitu saja pengenalannya, selanjutnya dapat dipelajari dari internet sendiri tentang itu database.

Pada studi kasus di ebook ini, kita akan menggunakan salah satu database yang sudah tersedia secara umum aja dan sudah cukup terkenal di dunia *persilatan*. Sederhana, simple, dan cepat.

¹ <http://www.mandalamaya.com/pengertian-database/>

² <http://c-tekno.mywapblog.com/pengertian-database-dan-contohnya.xhtml>

SQLite



Baiklah, akhirnya database yang dijadikan contoh dalam penulisan ini adalah menggunakan SQLite. Karena untuk bot sifatnya lebih sesuai, yakni: ringan, dan dengan tipe database yang *embedded* (berdiri sendiri) lebih penulis sukai. Pada dasarnya bot telegram yang lebih fungsional sebagai asisten atau tools, tidak membutuhkan database besar dan komplek, yang sederhana pun sudah cukup. Kecuali jika diintegrasikan dengan web atau aplikasi, atau kebutuhan-kebutuhan khusus lainnya.

Meski di dalam tutorial ini menggunakan database SQLite, namun kalau jika mencoba menggunakan database lain, semisal mySQL atau mariaDB, PostgreSQL, Oracle, dlsb dipersilakan. Tidak dibatasi penggunaan *experienyanya* hanya untuk SQLite saja.

BATASAN: Di ebook ini tidak akan dibahas banyak cara pakai dan penggunaan database, karena bukan ebook bertema database, jadi hanya dibatasi dan berfokus sesuai topiknya.

Penggunaan database secara detail, dapat dicari di google ada begitu banyak contoh dan tutorialnya.

Studi Kasus

Ciyus deh, penulis aslinya *binun* ingin memberikan contoh program apa yang sederhana untuk dijadikan studi kasus pada ebook kedua ini. Wacana ini pernah di-“lempar”-kan ke grup bot php, gak banyak masukan. Mungkin dunia bot telegram memang belum terlalu menyentuh di kehidupan sehari-hari, sehingga mencari contohnya juga agak susah.

Di benak sempat terlintas beberapa ide dan kesulitannya:

- ≈ Sistem Absensi Siswa atau Karyawan (sepertinya butuh alat peraga tambahan biar sesuai real, kalau sekadar simulasi akan memunculkan banyak pertanyaan dan ketidakpercayaan hehe..)
- ≈ Sistem stock barang (agak komplek, sepertinya bakal ribet terutama jika buat yang baru mengenal apa itu database dan baru mulai. Otomatis nanti penulis dapat pertanyaan juga bingung, lebih sering banyak yang bingung cara menyampaikan pertanyaan daripada permasalahannya sendiri hehe..)
- ≈ Pembuatan jadwal sholat (ini sih terkait dengan sumber data, dan memang ke depan akan dibuat bersamaan dengan proyek Fathimah Open Source)
- ≈ Log trail, rasanya kok terlalu sederhana.. Tapi bisa jadi, alternatifnya.
- ≈ Pembuatan jadwal pelajaran sekaligus remindernya, hmm melibatkan skeduling cron, yang *windows based* akan kesulitan mengikutinya. Bisa juga *sih* yang cara yang mudah, tapi hanya untuk bot metode poll.
- ≈ Bot pencatatan buku perpustakaan, hmm.. Hmm, ini mirip dengan stock barang. Kecuali bot yang diperuntukkan pengecekan buku yang databasenya perpustakaan sudah ada.
- ≈ Dan lain sebagainya, yang sempat bikin garuk-garuk kepala sendiri :-D

Pada akhirnya, setelah merenung menimbang dan memperhatikan... Kiranya enak membuat bot seperti “diary”. Yakni, bot yang akan merekam atau mencatat pesan atau tulisan yang dibuat. Sehingga bisa ditampilkan kapan saja.

Diary sebenarnya lebih cocok buat kaum hawa, namun kaum Adam sering memakai diary juga sebenarnya buat menyimpan hal-hal penting lain meski beda konteks, yakni *non* curhatan perasaan dan hati hehe..

Dan.. Karena yang buat atau developernya seorang cowok tulen, diputuskanlah nama botnya *Catatan si Boy Bot* :-D

Bot Diary, “Catatan Si Bot”



Seperti dikisahkan dalam *episode* sebelumnya, studi kasus kali ini akan bercerita tentang bot diary. Yang pakai sih boleh cowok atau pun cewek, terserah saja.

Biar lebih jelas kita tuliskan ruang lingkup kerjanya bot, sebagai berikut :

- ~ Proses pencatatan bot akan dilakukan berdasarkan userID Telegram. Sehingga bisa multi user, siapapun sepanjang menggunakan telegram dan mengakses bot ini.
- ~ Detail info user tidak dicatat bot, biar gak semakin panjang tulisan ini. Kita sederhanakan saja. Penulis rasa cocok buat dijadikan bahan PR hehe
- ~ Bot akan mencatat atau merekam pesan yang masuk berdasarkan *trigger* tertentu ke database. Jadi tidak semua pesan ke bot akan disimpan. Hanya yang melalui command, misalnya `/tambah`
- ~ Pesan kita batasi dulu hanya sebatas teks dan emoticon (UTF Char) saja. Sementara untuk multimedia, mungkin ke depan saja. Atau tugas pembaca sendiri untuk studi kasus pribadi alias PR nya menyelesaikan atau menambahkan fitur yang ini hehe..
- ~ Bot memperlihatkan daftar catatan yang pernah dibuat.
- ~ Bot menampilkan ulang catatan yang pernah dibuat.
- ~ Bot bisa menghapus, tapi tidak bisa mengedit. Karena fitur edit terlalu kompleks untuk dibuat, maka kita tiadakan dulu.
- ~ Bot melakukan paging (membuat halaman baru) jika pesan yang dibuat sudah banyak. Karena kalau engga, layar bisa penuh (eh yang ini skip dulu, edisi berikutnya aja, mungkin)
- ~ Bot mendukung pencarian data diary.

- ~ Bot menampilkan statistik..
(eh yang ini gak usah, nanti makin lama kelarnya hehe)
- ~ Bot menampilkan report berdasarkan.. (ini juga kita skip aja).
- ~ Bot tidak ada adminnya, semua bersifat user. Sehingga satu user tidak akan melihat catatan user lainnya. Dan tidak ada sistem administrasi nya.

Bagaimana? Apakah ribet? Seharusnya belum, karena baru dimulai :-D

Baiklah mari kita lanjutkan.

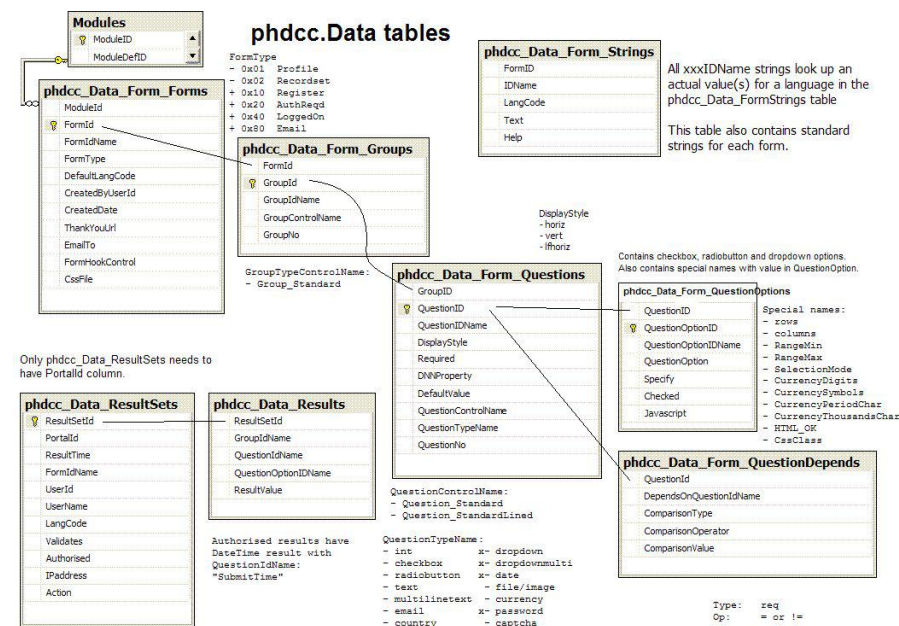
Tips :

Sebaiknya selalu buat ruang lingkup bot agar jelas tujuan dan apa yang akan dilakukan saat mengcoding.

Ketidakmampuan mendeskripsikan ruang lingkup bot, membuat tidak jelas bot yang akan dibuat akan bekerja seperti apa.



Struktur Data



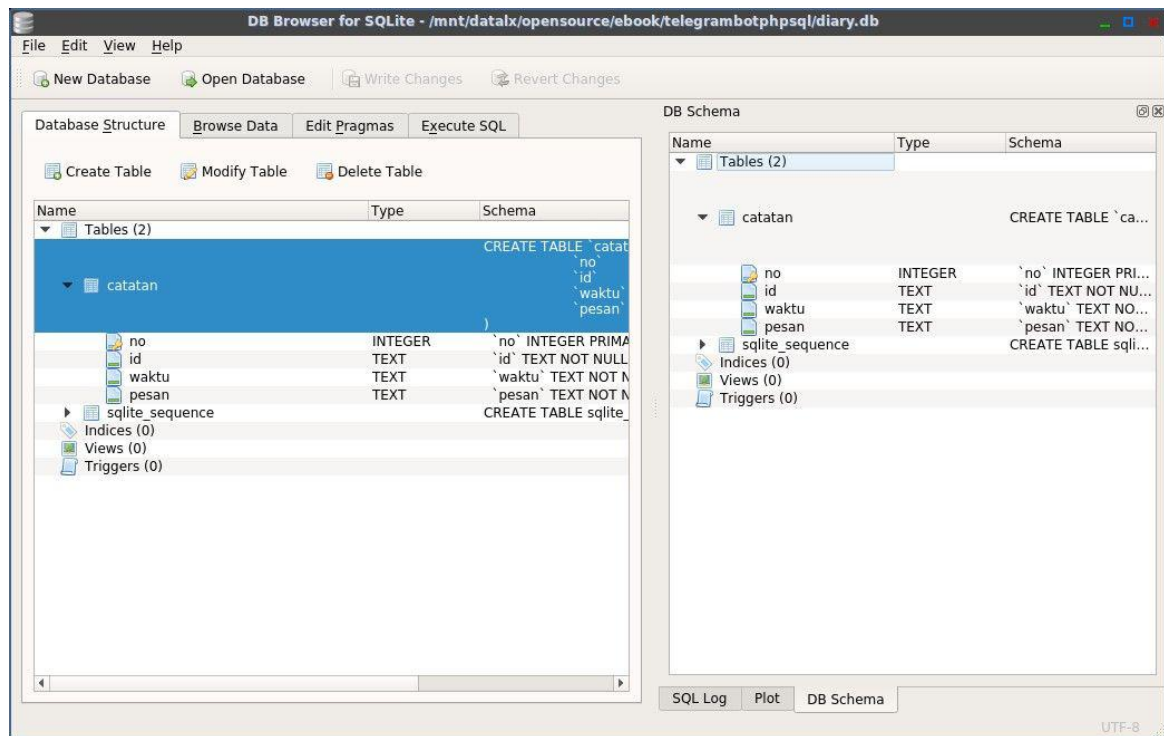
Kita enggak akan buat database serumit di atas, hehe.. sesederhana mungkin saja, dalam hal ini diusahakan 1 tabel saja dahulu. Karena baru mulai, baru belajar, jangan dibuat sulit, dibuat mudah saja yak.

Struktur table yang akan dibuat adalah sebagai berikut:

1. **no:** integer, primary key, auto increment
memuat nomor catatan yang dijadikan indeks semuanya.
2. **id:** text, not null
memuat id user telegram.
dibuat tipe nya text karena Integer kurang panjang.
3. **waktu:** text, not null
memuat data waktu, seharusnya tipe Date. Namun, SQLite tidak support format tersebut, maka kita isikan dalam tipe text.
4. **pesan:** text, not null
memuat pesan atau catatan diary nya.

Untuk membuat struktur tersebut, kita bisa menggunakan **DB Browser for SQLite** yang bisa bekerja baik dalam sistem operasi linux maupun windows.

DB Browser for SQLite



Dapat didownload di <http://sqlitebrowser.org/> baik Linux maupun Microsoft Windows dapat mempergunakannya.

Windows download di:

<https://github.com/sqlitebrowser/sqlitebrowser/releases/download/v3.9.0/DB.Browser.for.SQLite-3.9.0-win32.exe>

Untuk distro ubuntu / debian / peppermintOS atau sejenisnya:

```
sudo add-apt-repository -y ppa:linuxgndu/sqlitebrowser
sudo apt-get update
sudo apt-get install sqlitebrowser
```

Sedangkan untuk Fedora:

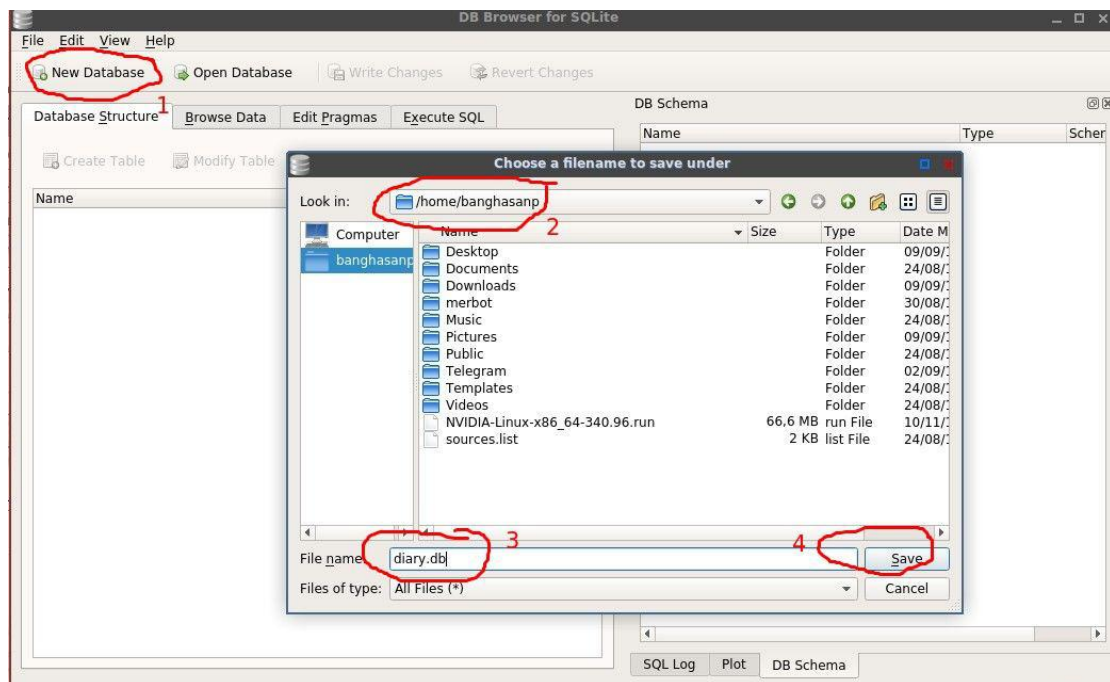
```
sudo dnf install sqlitebrowser
```

Untuk Sistem Operasi lainnya bisa di cek di situs resminya sqlite browser yak.

Melalui SQLite browser ini kita bisa dengan leluasa membuat, menambah, memodif, dan menghapus table atau data di dalamnya.

Membuat DataBase “diary.db”

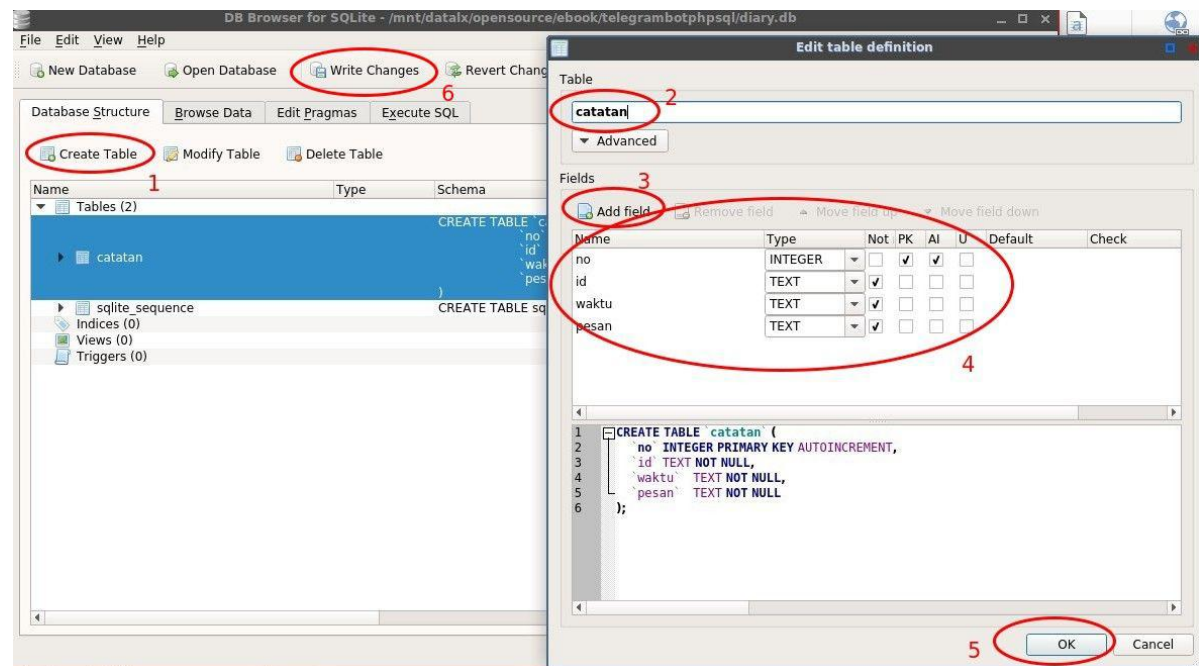
Buka SQLite Browser kemudian :



1. Klik **New Database**
2. Pilih Folder dimana Database akan disimpan. Usahakan 1 folder dengan proyek script
3. Beri nama databasenya: “diary.db”
4. Klik tombol **Save**.

Sampai di sini, mestinya sangat mudah yak.

Membuat Table “catatan”



1. Klik **Create Table**
2. Isikan nama table: “catatan”
3. Klik Add field
4. Tambahkan field-field seperti struktur data table di bab sebelumnya.
5. Sesudah ditambahkan semua, klik OK
6. Terakhir klik **Write Changes** untuk menyimpan semua perubahan tersebut.

Sampai di sini operasi pembuatan database dan table yang dibutuhkan sudah selesai.

Tips:

Semua perubahan dalam SQLite Browser tidak atau belum tersimpan sebelum kita klik tombol Write Changes. Berhati-hatilah sesudah menambah field kemudian langsung menutupnya.

Medoo



PHP Database Framework

Dalam hal pengurusan database di PHP, penulis termasuk "males" untuk membuat sendiri atau pakai dari bawaan PHP secara langsung.

Mengingat database framework untuk database sebenarnya sudah sangat banyak tersedia di internet. Namun, karena butuhnya hanya untuk akses database, tidak diperlukan framework yang besar seperti Code Igniter, Symfoni, Laravel, atau lainnya. Semakin besar frameworknya nanti malah semakin melebar dikhawatirkan bahasanya, tidak ke inti.

Atas pertimbangan tersebut, dan berpikir praktis (baca: males) dipilihlah medoo sebagai buah hati pengelolaan database kali ini.

Medoo telah ditest dan berkali-kali dilibatkan dalam pembuatan bot telegram, hasilnya relatif memuaskan.

Dalam beberapa kasus, fitur medoo oleh penulis telah di modifikasi untuk keperluannya.

Pembaca boleh mendownload langsung dari situs medoo.in ataupun menggunakan versi modifikasi yang telah disesuaikan. Perbedaan signifikan hanya ada pada fitur match search boolean dan tambahan fitur-fitur kecil. Semua tetap open source, jadi dapat diperiksa tidak akan ada yang 'nyelip' disisipkan dalam source code.

Requirement untuk menjalankan medoo:

- ~ PHP 5.1+, greatly recommend for PHP 5.4+ with PDO support.
Penulis sendiri menggunakan PHP versi 7, jadi selalu aman dan nyaman hehe..
- ~ Installed database like MySQL, MSSQL, SQLite or more.
- ~ Pastikan ekstensi php_pdo_xxx sudah diinstall dan di enable kan.
- ~ Sedikit pengetahuan tentang SQL.

Daftar Php_pdo Extension yang dapat dihandle oleh medoo:

- ~ MySQL, MariaDB -> php_pdo_mysql
- ~ MSSQL (Windows) -> php_pdo_sqlsrv
- ~ MSSQL (Linux/UNIX) -> php_pdo_dblib
- ~ Oracle -> php_pdo_oci
- ~ SQLite -> php_pdo_sqlite
- ~ PostgreSQL -> php_pdo_pgsql
- ~ Sybase -> php_pdo_dblib

Dengan adanya daftar pdi ekstensi yang dihandle diatas, pada prinsipnya database yang dipergunakan bisa dengan mudah dipindah atau diganti asal memiliki struktur data dan table yang sama, serta menggunakan query (sql) yang umum (sama).

Misalnya pada studi kasus ini, benar-benar SQL nya dibuat identik. Sehingga pembaca bisa dengan mudah melakukan portasi pindah data dari SQLite ke mySQL/mariaDB atau yang lainnya.

Catatan dan Peringatan:

Semua operasi pada pembuatan bot di sini menggunakan *command line* alias CLI. Tidak dijalankan pada browser atau GUI (kecuali DB Browser SQLite saja).

Sehingga direkomendasikan sekali pembaca juga mulai membiasakan diri menggunakan CLI. Agar mudah dalam developing dan debugging selanjutnya.

Water Test - Persiapan Memulai Cross Check PHP dan Ekstensi

Untuk memastikan semuanya oke.. Mari kita periksa dahulu alat perang kita.

1. Versi PHP

Ketikkan `php -v`

Pastikan versi php di atas 5.4

```
banghasanp@banghasanp-HP-EliteBook-8540p /mnt/data/x/opensource/ebook/telegrambotphpsql $ php -v
PHP 7.0.8-0ubuntu0.16.04.2 (cli) ( NTS )
Copyright (c) 1997-2016 The PHP Group
Zend Engine v3.0.0, Copyright (c) 1998-2016 Zend Technologies
with Zend OPcache v7.0.8-0ubuntu0.16.04.2, Copyright (c) 1999-2016, by Zend Technologies
```

2. Cek ekstensi PHP

Ketikkan `php --ini`

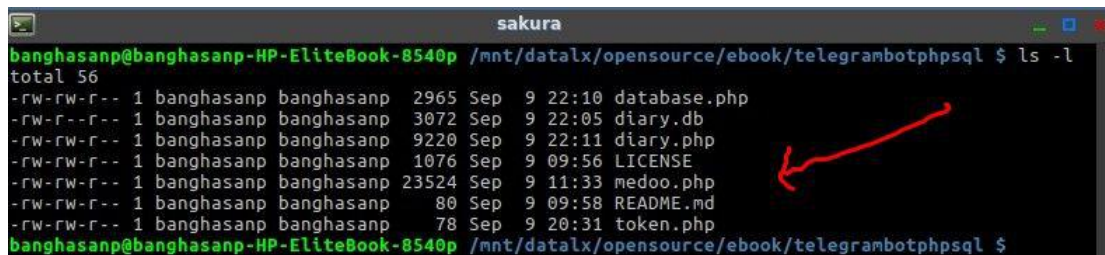
Pastikan ada pdo, curl, sqlite atau mysql.

```
sakura
--ri <name>      Show configuration for extension <name>.

banghasanp@banghasanp-HP-EliteBook-8540p /mnt/data/x/opensource/ebook/telegrambotphpsql $ php --ini
Configuration File (php.ini) Path: /etc/php/7.0/cli
Loaded Configuration File:      /etc/php/7.0/cli/php.ini
Scan for additional .ini files in: /etc/php/7.0/cli/conf.d
Additional .ini files parsed:    /etc/php/7.0/cli/conf.d/10-opcache.ini,
                                  /etc/php/7.0/cli/conf.d/10-pdo.ini,
                                  /etc/php/7.0/cli/conf.d/15-xml.ini,
                                  /etc/php/7.0/cli/conf.d/20-calendar.ini,
                                  /etc/php/7.0/cli/conf.d/20-ctype.ini,
                                  /etc/php/7.0/cli/conf.d/20-curl.ini,
                                  /etc/php/7.0/cli/conf.d/20-dom.ini,
                                  /etc/php/7.0/cli/conf.d/20-exif.ini,
                                  /etc/php/7.0/cli/conf.d/20-fileinfo.ini,
                                  /etc/php/7.0/cli/conf.d/20-ftp.ini,
                                  /etc/php/7.0/cli/conf.d/20-gd.ini,
                                  /etc/php/7.0/cli/conf.d/20-gettext.ini,
                                  /etc/php/7.0/cli/conf.d/20-iconv.ini,
                                  /etc/php/7.0/cli/conf.d/20-imagick.ini,
                                  /etc/php/7.0/cli/conf.d/20-imap.ini,
                                  /etc/php/7.0/cli/conf.d/20-intl.ini,
                                  /etc/php/7.0/cli/conf.d/20-json.ini,
                                  /etc/php/7.0/cli/conf.d/20-mbstring.ini,
                                  /etc/php/7.0/cli/conf.d/20-mcrypt.ini,
                                  /etc/php/7.0/cli/conf.d/20-pdo_sqlite.ini,
                                  /etc/php/7.0/cli/conf.d/20-phar.ini,
                                  /etc/php/7.0/cli/conf.d/20-posix.ini,
                                  /etc/php/7.0/cli/conf.d/20-readline.ini,
                                  /etc/php/7.0/cli/conf.d/20-shmop.ini,
                                  /etc/php/7.0/cli/conf.d/20-simplexml.ini,
                                  /etc/php/7.0/cli/conf.d/20-sockets.ini,
                                  /etc/php/7.0/cli/conf.d/20-sqlite3.ini,
                                  /etc/php/7.0/cli/conf.d/20-sysvmsg.ini,
                                  /etc/php/7.0/cli/conf.d/20-sysvsem.ini,
                                  /etc/php/7.0/cli/conf.d/20-sysvshm.ini,
                                  /etc/php/7.0/cli/conf.d/20-tokenizer.ini,
                                  /etc/php/7.0/cli/conf.d/20-wddx.ini,
                                  /etc/php/7.0/cli/conf.d/20-xmlreader.ini,
```

Cara lain, cek melalui `phpinfo()` ;

Struktur File dan Folder Proyek Diary “Catatan si Bot”



```
sakura
banghasanp@banghasanp-HP-EliteBook-8540p /mnt/data/tx/opensource/ebook/telegrambotphpsql $ ls -l
total 56
-rw-rw-r-- 1 banghasanp banghasanp 2965 Sep  9 22:10 database.php
-rw-rw-r-- 1 banghasanp banghasanp 3072 Sep  9 22:05 diary.db
-rw-rw-r-- 1 banghasanp banghasanp 9220 Sep  9 22:11 diary.php
-rw-rw-r-- 1 banghasanp banghasanp 1076 Sep  9 09:56 LICENSE
-rw-rw-r-- 1 banghasanp banghasanp 23524 Sep  9 11:33 medoo.php
-rw-rw-r-- 1 banghasanp banghasanp 80 Sep  9 09:58 README.md
-rw-rw-r-- 1 banghasanp banghasanp 78 Sep  9 20:31 token.php
banghasanp@banghasanp-HP-EliteBook-8540p /mnt/data/tx/opensource/ebook/telegrambotphpsql $
```

Dengan tujuan agar pembaca tidak bingung, ada 5 file minimal yang harus ada pada catatan si bot ini, yang harus diletakkan satu folder. Yakni, terdiri atas:

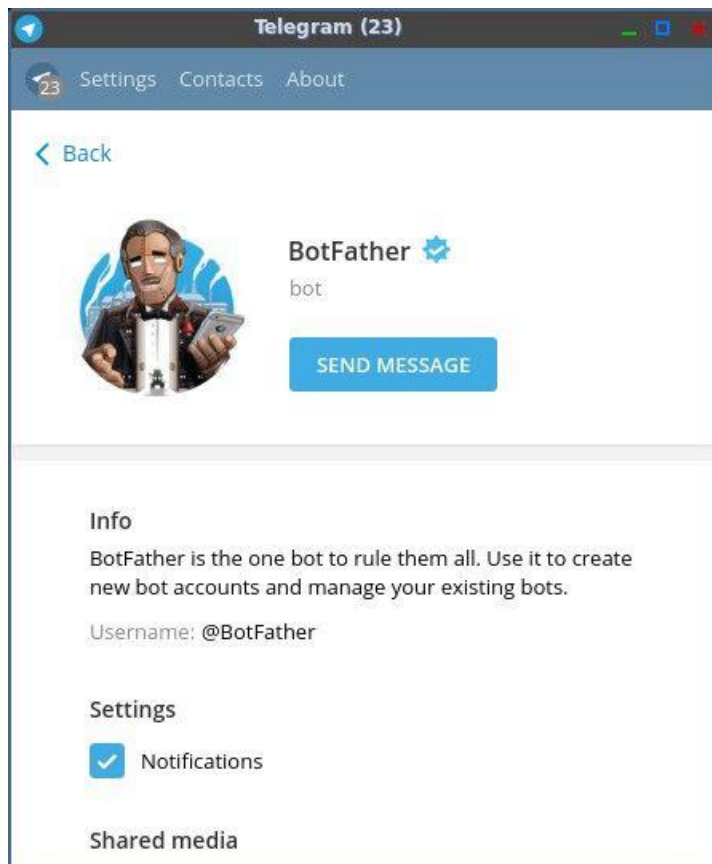
1. File `token.php`
2. File `diary.php` (file engine utama bot)
3. File `medoo.php` (php database framework)
4. File `diary.db` (file database)
5. File `database.php` (semua fungsi bot yang akses ke database)

Untuk file `medoo.php` dapat di download pada situs resminya medoo.in atau di dapat dari clone proyek ebook ke-2.

Sedangkan `diary.db` sudah dibahas sebelumnya.

Pembuatan Token Bot

Hubungi Bapak Bot!



Untuk membuat bot dengan bapak bot, bisa dibaca dan diikuti pada ebook pertama ya. Ikuti kembali sampai mendapatkan token.

Kemudian buat file `token.php` yang berisi isi token bot tersebut :

```
<?php
$token = "isidengantokenbotmu";
```

Simpan dan letakkan dalam satu folder proyek diary bot.

Catatan

Di ebook ke-2 ini, file **token.php** ini kamu harus membuatnya sendiri. Tidak dimasukkan dalam file proyek bawaan ebook maupun di proyek github.

Hal ini untuk antisipasi keamanan dan sekaligus keseriusanmu mengikuti tutorial ebook ini.

Database Engine Bot

```
1 <?php
2
3 /*
4  * Ebook II: Telegram Bot PHP dan Database SQL
5  * oleh bang Hasan ( @hasanudinhs )
6  *
7  * Fungsi Database untuk Diary Bot Telegram
8  *
9  */
10
11
12 // masukkan database framework nya
13 require_once 'medoo.php';
14
15 // koneksikan ke database
16
17 // ini contoh menggunakan SQLite
18 $database = new medoo([
19     'database_type' => 'sqlite',
20     'database_file' => 'diary.db',
21 ]);
22
```

Mari kita buat file “database.php”, sekaligus kita bahas satu-satu.

Masukkan PHP Database Framework medoo ke dalam proyek.

```
require_once 'medoo.php';
```

Kemudian diinisiasi koneksi ke database SQLite yang telah kita buat dan persiapkan di episode sebelumnya.

```
// koneksikan ke database
// ini contoh menggunakan SQLite
$database = new medoo([
    'database_type' => 'sqlite',
    'database_file' => 'diary.db'
]);
```

Jika ingin mengubah ke dalam database mySQL atau mariaDB, bisa menggunakan ini :

```
$database = new medoo([
    'database_type' => 'mysql',
    'database_name' => 'diary',
    'server' => 'localhost',
    'username' => 'root',
    'password' => 'rahasiakusendiri',
    'charset' => 'utf8'
]);
```


Sampai di sini, asalkan hati-hati dan teliti membaca semestinya tidak akan ada kesulitan yang berarti.

Kita lanjutkan fungsi insert atau menambahkan data pada Database :

```
36 // fungsi untuk menambah diary
37 function diarytambah($iduser, $pesan)
38 {
39     global $database;
40     $last_id = $database->insert('catatan', [
41         'id' => $iduser,
42         'waktu' => date('Y-m-d H:i:s'). ' WIB',
43         'pesan' => $pesan,
44     ]);
45
46     return $last_id;
47 }
```

Pada fungsi di atas, dipergunakan untuk menambah pesan atau catatan diary ke bot, berdasarkan id user. Dicatat pula waktunya.

Fungsi untuk menghapus diary.

Proteksi harus mengecek id user, tidak hanya nomor id pesan saja.

```
49 // fungsi menghapus diary
50 function diaryhapus($iduser, $idpesan)
51 {
52     global $database;
53     $database->delete('catatan', [
54         'AND' => [
55             'id' => $iduser,
56             'no' => $idpesan,
57         ],
58     ]);
59
60     return '🗑️ telah dilaksanakan..';
61 }
```

Sangat sederhana sekali script source codenya, bukan?

```
// fungsi melihat daftar diary user
function diarylist($iduser, $page = 0)
{
    global $database;
    $hasil = '@ Maaf ya, tidak ada catatan diary di hatiku..';
    $datas = $database->select('catatan', [
        'no',
        'id',
        'waktu',
        'pesan',
    ], [
        'id' => $iduser,
    ]);
    $jml = count($datas);
    if ($jml > 0) {
        $hasil = "\n[01F][3F0] *$jml Catatan Diary-mu Kusimpan Rapi Dihati:*\\n";
        $n = 0;
        foreach ($datas as $data) {
            $n++;
            $hasil .= "\n$n. ".substr($data['pesan'], 0, 10)."...\n[01F][3F0] ` $data[waktu]`\\n";
            $hasil .= "\n[01F][440] /view\_ $data[no]\\n";
        }
    }
    return $hasil;
}
```

Sedangkan diatas adalah fungsi untuk melihat semua pesan diary yang sudah dicatat ke dalam bot. Akan ditampilkan sekaligus list view berdasarkan nomor id pesan.

```
90 // fungsi melihat isi pesan diary
91 function diaryview($iduser, $idpesan)
92 {
93     global $database;
94     $hasil = '@ Maaf ya, diarymu yang itu tidak ditemukan dihatiku.\nMungkin saja bukan buatmu..';
95     $datas = $database->select('catatan', [
96         'no',
97         'id',
98         'waktu',
99         'pesan',
100     ], [
101         'AND' => [
102             'id' => $iduser,
103             'no' => $idpesan,
104         ],
105     ]);
106     $jml = count($datas);
107     if ($jml > 0) {
108         $data = $datas[0];
109         $hasil = "\n[01F][1F0] Diary nomor $data[no] yang tersimpan dihatiku berisi:\\n~~~~~\\n";
110         $hasil .= "\n[01F][1F0] $data[pesan]\\n[01F][3F0] ` $data[waktu]`";
111         $hasil .= "\n[01F][200] Hapus? /hapus\_ $data[no]";
112     }
113     return $hasil;
114 }
115 }
```

Fungsi tersebut untuk melihat isi catatan diary yang sudah tersimpan. Metodenya juga sederhana, cocokkan id dan nomor pesan yang diminta maka akan ditampilkan hasilnya.

```

117 // fungsi mencari pesan di diary
118 function diarycari($iduser, $pesan)
119 {
120     global $database;
121     $hasil = '☹ Maaf ya, apa yang kau cari selama ini tidak ditemukan..';
122     $datas = $database->select('catatan', [
123         'no',
124         'id',
125         'waktu',
126         'pesan',
127     ], [
128         'pesan[~]' => $pesan,
129     ]);
130     $jml = count($datas);
131     if ($jml > 0) {
132         $hasil = "📄 *$jml Catatan Diary-mu yang kau cari selalu kusimpan di hatiku*\n";
133         $n = 0;
134         foreach ($datas as $data) {
135             $n++;
136             $hasil .= "\n$n. ".substr($data['pesan'], 0, 10)."...\n🕒 `{$data[waktu]}`\n";
137             $hasil .= "\n👁 /view\_{$data[no]}\n";
138         }
139     }
140 }
141 return $hasil;
142 }

```

Dan terakhir fungsi database adalah melakukan pencarian berdasarkan kata kunci.

Source Code Lengkap Database.php

Dapat di unduh, dibuka atau dilihat di :

<https://github.com/banghasan/telegrambotphpsql/blob/master/database.php>

Engine Utama “diary.php”

Engine utama bot diary ini diambil dan dimodifikasi dari engine ebook pertama, yang menggunakan curl. Yakni `PertamaCurlBot.php` (bisa dilihat di : <https://gist.github.com/banghasan/67f365e6df3e3b8dc1cfa8208b52a305>)

Hasilnya file `diary.php` dapat dilihat, diunduh:

<https://github.com/banghasan/telegrambotphpsql/blob/master/diary.php>

Tidak banyak modifikasi, beberapa yang bisa dijelaskan diantaranya:

```
29  /* buatlah file token.php isinya :
30
31  <?php
32
33  $token = "isiTokenBotmu";
34
35  */
36  require_once 'token.php';
```

Perubahan yang dibuat adalah, buatlah file `token.php`, seperti yang sudah dijelaskan sebelumnya.

```
49  // ambil databasenya
50  require_once 'database.php';
51
52  // aktifkan ini jika ingin menampilkan debugging poll
53  $debug = false;
```

Hubungkan file `database.php` yang kita buat diatas ke engine utama bot.

```
174  // ----- pantengin mulai ini
175  function sendMessage($idpesan, $idchat, $pesan)
176  {
177      $data = [
178          'chat_id'      => $idchat,
179          'text'         => $pesan,
180          'parse_mode'  => 'Markdown',
181          'reply_to_message_id' => $idpesan,
182      ];
```

Kali ini, semua pesan akan di parse mode dalam format markdown. Hati-hati untuk pesan yang dimasukkan dengan tag terbuka, maka akan error.

Biasanya penulis berikan escape markdown, dalam tutorial ini di skip dulu. Mungkin diberikan dalam bentuk bonus atau materi tambahan saja.


```

239 case '/tambah':
240     if (isset($pecah[1])) {
241         $pesanproses = $pecah[1];
242         $r = diarytambah($iduser, $pesanproses);
243         $text = '😊 Goresan catatan indahmu telah berhasil kusematkan di dalam hatiku!';
244     } else {
245         $text = '❌ *ERROR:* Pesan yang ditambahkan tidak boleh kosong!_';
246         $text .= "\n\nContoh: `/pesan besok aku sahur mau puasa sunnah`";
247     }
248     break;
249
250 case '/view':
251     if (isset($pecah[1])) {
252         $pesanproses = $pecah[1];
253         $text = diaryview($iduser, $pesanproses);
254     } else {
255         $text = '❌ *ERROR:* `nomor pesan tidak boleh kosong.`';
256     }
257     break;
258
259 case '/hapus':
260     if (isset($pecah[1])) {
261         $pesanproses = $pecah[1];
262         $text = diaryhapus($iduser, $pesanproses);
263     } else {
264         $text = '❌ *ERROR:* `nomor pesan tidak boleh kosong.`';
265     }
266     break;
267
268 case '/list':
269     $text = diarylist($iduser);
270     if ($GLOBALS['debug']) {
271         print_r($text);
272     }
273     break;

```

Proses trigger command: /tambah /view /hapus dan /list

Semua mengakses pada file database.php yang telah dibuat sebelumnya. Sedikit pengecekan jika pada perintah /tambah tidak terdapat isi pesannya, maka akan menampilkan error. Demikian juga untuk lainnya.

Dalam hal ini pesan akan di pecah oleh fungsi preg match, untuk memastikan isi pesan kedua dalam format (/view_id dan /hapus_id) adalah tipe berupa angka :

```

205 if (preg_match("/^\s/view_(\d+)\s$/i", $pesan, $cocok)) {
206     $pesan = "/view $cocok[1]";
207 }
208
209 if (preg_match("/^\s/hapus_(\d+)\s$/i", $pesan, $cocok)) {
210     $pesan = "/hapus $cocok[1]";
211 }

```

Pada fungsi terakhir, yakni pencarian juga menggunakan preg match (regex).

Hal ini penulis rasa sangat urgent, karena untuk menghindari SQL *injection* yang sering menyerang. Sedikit trik proteksi yang diberikan di sini adalah hanya menerima pesan pencarian yang berupa huruf atau angka saja.

Selain itu akan diabaikan, dianggap error.

```

275     case '/cari':
276         // saya gunakan pregmatch ini salah satunya untuk mencegah SQL injection
277         // hanya huruf dan angka saja yang akan diproses
278         if (preg_match("/^\cari ((\w| )+)$/i", $pesan, $cocok)) {
279             $pesanproses = $cocok[1];
280             $text = diarycari($idUser, $pesanproses);
281         } else {
282             $text = "❌ *ERROR: `kata kunci harus berupa kata (huruf dan angka) saja.`";
283         }
284         break;

```

Buat yang bingung tentang regex, sangat wajar sekali kok.

Banyak cara sebenarnya untuk memverifikasi pesan, namun regex lah yang penulis anggap paling efektif.

Misalnya menggunakan array untuk mengecekkan seperti ini:

```

$allowedChars =
array("a","b","c","d","e","f","g","h","i","j","k","l","m","n","o","p","q","r","s",
"t","u","v","w","x","y","z","
","A","B","C","D","E","F","G","H","I","J","K","L","M","N","O","P","Q","R","S","
T","U","V","W","X","Y","Z"," " ,"0","1","2","3","4","5","6","7","8","9","
","@",".", "-", "_", "+", " ");
$input = "Test";
$input = str_split($input);
$message = "Sigh, not found...";
foreach($input as $letter) {
    if (in_array($letter, $allowedChars)) {
        $message = "Yep, found.";
        break;
    }
}
echo $message;

```

Namun hasilnya sangat panjang, tidak efektif. Sedangkan jika menggunakan regex hanya satu atau dua baris saja.

Clone Proyek

The screenshot shows the GitHub repository page for `banghasan/telegrambotphpsql`. The repository has 1 star, 0 forks, and 0 issues. The commit history table is as follows:

File	Commit Message	Time Ago
<code>.gitignore</code>	inisiasi	4 hours ago
<code>LICENSE</code>	Initial commit	14 hours ago
<code>README.md</code>	Update README.md	an hour ago
<code>database.php</code>	Applied fixes from StyleCI	an hour ago
<code>diary.db</code>	first release	an hour ago
<code>diary.php</code>	final update first version 0.1	30 minutes ago
<code>medoo.php</code>	Applied fixes from StyleCI	an hour ago

Untuk mendapatkan file-file dapat tutorial ebook ke-2 ini, tanpa sulit-sulit menulis ulang adalah dengan cara mengkloning proyek yang penulis buat di github.

Alamat url github: <https://github.com/banghasan/telegrambotphpsql>

Download langsung :

<https://github.com/banghasan/telegrambotphpsql/archive/master.zip>

Atau bisa menggunakan senjata andalan, git :

```
git clone git@github.com:banghasan/telegrambotphpsql.git
```

```
Cloning into 'telegrambotphpsql'...
```

```
remote: Counting objects: 29, done.
```

```
remote: Compressing objects: 100% (22/22), done.
```

```
remote: Total 29 (delta 8), reused 17 (delta 5), pack-reused 0
```

```
Receiving objects: 100% (29/29), 20.72 KiB | 0 bytes/s, done.
```

```
Resolving deltas: 100% (8/8), done.
```

```
Checking connectivity... done.
```

Maka akan terbentuk folder **telegrambotphpsql** disana file-file pada ebook ini dapat ditemukan.

Diary si Bot

Kiranya seperti ini hasil pertama ketika meminta bantuan si bot...



Untuk menjalankan diary si bot:

`php diary.php`

Kira-kira hasilnya adalah seperti ini :

```
php diary.php
Ver. 0.1 OK Start!
2016-09-10 00:10:01
----
```

Mengenal Bot Diary

Perintah-perintah yang dapat di jalankan pada bot diary :

~ **/tambah [pesan]**

Contoh:

```
/tambah Dear Diary, Apa kabarmu sekarang?
```

Maksimal 4000 karakter, boleh menggunakan char khusus (enter, emoticon, dan lain-lainnya). Hati-hati dengan char markdown (*, `, _).

~ **/list**

Contoh:

```
/list
```

Melihat daftar keseluruhan catatan diary yang sudah dimasukkan ke bot.

~ **/view_id**

melihat catatan diary berdasarkan nomor indeks pesan.

Contoh:

```
/view 100
```

```
/view_123
```

Hanya nomor pesan dan iduser yang cocok akan ditampilkan.

~ **/hapus_id**

menghapus catatan diary berdasarkan nomor indeks pesan.

Contoh:

```
/view 100
```

```
/view_123
```

Hanya nomor pesan dan iduser yang cocok akan benar-benar dihapus.

~ **/cari [kata kunci]**

mencari kata kunci tertentu.

Contoh:

```
/cari cinta tidak memilih
```

Dibatasi yang diproses hanya kata kunci berupa teks (huruf dan angka) saja.

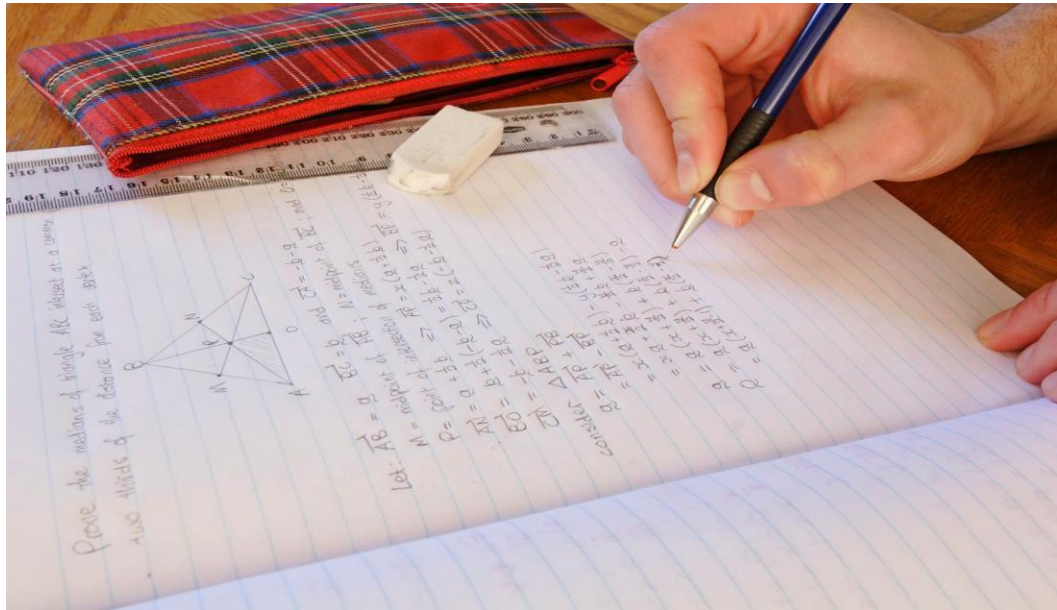
~ **/time**

melihat waktu sekarang pada bot (bisa berguna untuk pengecekan selisih waktu UTC).

- ~ **/start**
sapaan pertama bot, meski bisa diulang-ulang.
- ~ **/help**
menampilkan bantuan perintah-perintah bot.



Pekerjaan Rumah Belum Tuntas



Penulis rasa sudah sangat cukup sebagai contoh bot telegram yang beroperasi dengan bahasa pemrograman PHP dan konektivitas database.

Contoh yang dipaparkan telah mencakup banyak hal hampir keseluruhan dari CRUD meski tanpa U (Create, Read, Delete tanpa Update).

Jika bot diary ini lebih dikembangkan beberapa hal yang harus ditambahkan seperti pada paparan sebelumnya. Diantaranya :

- ~ Menambahkan Detail Info ke tabel user tersendiri
- ~ Menambahkan Paging (melihat list per halaman) misalnya dibatasi per view tampil tampil sebanyak 5 daftar catatan diary.
- ~ Menambahkan fitur-fitur lain, tambah penyimpanan foto, sticker, dan multimedia lainnya
- ~ Menambahkan asesoris, keyboard inline agar lebih cantik

Dan kesemuanya itu bisa pembaca coba praktekan sendiri untuk mengasah kemampuan ya :-D

Penutup



Kado untuk Bulan Besar

Alhamdulillah selesai sudah buku ke-2 tentang pembuatan bot telegram. Ebook yang pertama selesai pada bulan Ramadhan, dan kali ini di bulan Haji.

Semoga dapat bermanfaat, utamanya buat para developer atau orang umum namun memiliki semangat juang belajar yang tinggi. Tidak dikhususkan untuk kalangan tertentu.

Dan tentu saja bermanfaat bagi diri penulis sendiri, yang notebene juga masih sambil belajar. Dengan mencatatkan pada tulisan, harapannya tentu menjadi tahan lama dan lebih panjang sebagai pengingat. Apalagi jika ebook ini dapat dibaca dan dipraktekkan oleh orang lain, tentu menjadi memiliki nilai lebih manfaatnya.

Akhiru kata, mohon maaf jika dalam penyajian kesemua ebook ada yang kurang-kurang berkenan atau hal yang kurang tepat.

Sepekan yang syahdu mendung dan dingin
Depok - Sabtu, 10 September 2016
Bertepatan 8 Dzulhijjah 1437 H

Tentang Penulis



Memiliki nama *beken*, Hasanudin H. Syafaat, dilahirkan 23 Juni 1981 di kota tahu, Kediri – Jawa Timur. Maka jangan heran jika nanti menemukan kata “kediri” sering dijadikan contoh dalam banyak hal atau kasus. Karena, baginya “kediri” adalah *magic word* :-D

Sejak SMP sudah mengenal dan mempelajari bahasa pemrograman Bahasa Basics, Turbo Pascal, dan Assembly. Kemudian dilanjutkan Delphi, yang berlanjut dengan bahasa pemrograman lainnya, sejak kuliah PHP hingga sekarang. Saat dibangku sekolah, pernah mengeyam menjadi peserta TOKI (Tim Olimpiade Komputer Indonesia), mesti tidak pernah menjadi juara hehe..

Memiliki hobi di depan laptop dengan segala aktivitasnya, fotografi, jalan-jalan maupun naik gunung.

Pernah bekerja pada perusahaan aplikasi core banking selama lebih dari 5 tahun, sebagai programmer. Masih fokus terhadap aplikasi basis keuangan dan kecenderungannya akan bisnis syariah dan aplikasi penunjangnya. Saat ini bekerja sebagai programmer freelance. Belum menetap, posisi di kota Depok, yang berdekatan dengan Universitas Indonesia.

Tidak fanatis terhadap bahasa pemrograman tertentu, yang mana saja dipakai: C, python, nodejs, lua, dll.. yang penting aplikasi berjalan, bisa dikerjakan, dan client puas :-D

Bot telegram menjadi salah satu yang menarik perhatiannya. Dan saat ini telah membuat beberapa bot seperti :

- ~ **Fathimah**, bot religi yang multi talent. Berbagai fitur disematkan. Diantaranya: Quran Search Engine, alQuran per kata, Hadits Search Engine, Checking Sertifikat Halal Mui, Jadwal sholat, Berita Online, Info dari BMKG, Prakiraan Cuaca, Kamus Besar Bahasa Indonesia, dan beberapa AI dlsb.
- ~ **Cek Resi** dan **JNEbot**, pengecekan resi berbagai ekspedisi serta ongkos kirimnya.
- ~ **Cek Tiket KA**, bot pengecekan jadwal kereta api dan harga tiketnya.
- ~ **Cek Harga**, bot untuk pengecekan harga dari berbagai marketplace, khususnya untuk tokopedia lengkap detail sampai tracking resinya.
- ~ **Cek PLN**, bot info pengecekan PLN, besar tagihan dan kapan terbayar.
- ~ **BPJS**, bot info kepesertaan anggota BPJS. Namun hanya berumur kurang

dari 1 hari, langsung dimatikan terkait perizinan.

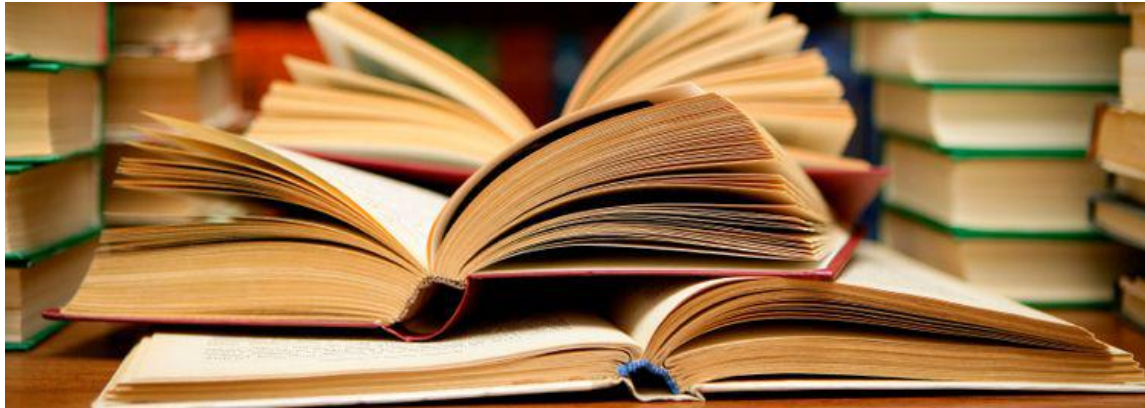
- ~ **iBanking**, bot untuk melakukan checking saldo dan mutasi akun bank. Dibatasi pengguna nya.
- ~ **Pulsa**, bot untuk transaksi pulsa pra dan pasca bayar, serta pembayaran tagihan PPOB.
- ~ **Simulasi** bot, dipergukan untuk integrasi komunikasi antar platform bot messenger, baik facebook messenger, line, telegram, dan lainnya.
- ~ **Quote**, bot untuk memberikan pesan-pesan hikmah sekaligus broadcast ke semua anggotanya
- ~ **SMS Gratis**, bot untuk kirim sms secara gratis.
- ~ **Covata**, bot proto, menggunakan tg-cli dengan berbagai fiturnya sendiri.
- ~ **myQuiz**, bot untuk hiburan bermain quiz.
- ~ Serta bot lainnya yang belum diperkenalkan secara umum, biasanya untuk tools saja, termasuk bot display toko, bot cek domain, kalkulator, dlsb.

Semoga, diantara bot tersebut Anda sudah mengenalnya :D

Jangan sungkan menyapa, berdialog, berdiskusi dengannya. Apalagi jika ada proyek yang menarik buat kebaikan bersama hehe..

Namun, sebaiknya jika hal umum ngobrol di grup Deveoper Bot Telegram - PHP @botphp <https://telegram.me/botphp> saja, agar saling bersosial dan membentuk komunitas bersama.

Referensi



Ditulis secara tidak beraturan, tidak diurutkan berdasar materi, juga tidak diurutkan berdasarkan abjad. Sebagian besar adalah yang dipergunakan oleh penulis dalam menyelesaikan ebook II ini.

1. Catatanku tentang OS Peppermint di **medium**, <https://t.co/WdckRQ0yGA>
2. **PeppermintOS** <https://peppermintos.com/guide/downloading/>
3. **Sakura**, Terminal Emulator <http://www.pleyades.net/david/projects/sakura>
4. **PHP** Docs, <http://php.net/docs.php>
5. **PHP Live Regex**, <http://www.phpliveregex.com/>
6. **SQLite**, <https://www.sqlite.org/> dan <https://id.wikipedia.org/wiki/SQLite>
7. **SQLite Browser**, <http://sqlitebrowser.org/>
8. **SQLite Manager**, <https://sourceforge.net/projects/sqlitemanager/>
9. **Sublime-Text**, editor terbaikku (belilah lisensinya jika suka),
<http://www.sublimetext.com/>
10. **Notepad++**, editor alternatif <https://notepad-plus-plus.org/>
11. **Meedo**, Database Framework untuk PHP, <http://medoo.in/>
12. **Github**, hosting proyek opensource, <https://github.com/>
13. **Telegram bot API**, <https://core.telegram.org/bots/api>
14. Tutorial **Ebook I: Edisi Pengantar Pembuatan Bot Telegram Menggunakan PHP**, <http://tutorial.fathimah.ga/>
15. **Kingsoft Office WPS**, untuk menulis tutorial ebook ini,
<http://www.kingsoftstore.com/> dan <http://wps.com/>
16. Download **Source Code Materi Ebook II** ini:
<https://github.com/banghasan/telegrambotphpsql>
17. **Diskusi Bot PHP** di Telegram: <https://telegram.me/botphp>
disarankan join di sini untuk diskusi, tanya jawab, dan mengembangkan komunitas bot php di telegram.
18. Resume channel **koleksi bot**: <https://telegram.me/botkoleksi>
19. Proyek **Fathimah**, <http://www.fathimah.ga/>
20. **Fathimah Channel**, base keseluruhan proyek dan semangat juang
<https://telegram.me/fathimahchan>

21. Selasar myQuran, Komunitas Muslim Indonesia - <https://telegram.me/myqers>
22. Hostus VPS murah meriah stabil dan mantap, yang saya pakai untuk kebutuhan akses online perbotan dan “jembatan” bot, gunakan link ini untuk mendapatkan spesial rate: <https://goo.gl/pVpHTW>
23. Themes Ebook ini diambil dari WPS Template.

Disclaimer

Penulis tidak bertanggung jawab, jika dalam praktek membuat bot dalam ebook ini menyebabkan efek samping apapun, misalnya kepala jadi pusing, mumet, ketagihan, lemas karena begadang ngetes, atau bahkan kalau sampai menimbulkan kerusakan hardware.

Semua tulisan adalah orisinal oleh penulis, termasuk source code-nya. Kecuali source code yang berasal dari pihak lain yang telah disebutkan.

Beberapa gambar *nyomot* dari internet, google image, yang bisa jadi memiliki hak cipta, penulis tidak memeriksanya dulu hanya dilihat sekilas saja tanpa jelas adanya larangannya mengambilnya. Namun, gambar yang diambil hanyalah sebagai penghias atau dekorasi sepenuhnya di sini, tidak termasuk inti pada bahasan dan dibuat dalam rangka pendidikan, bukan untuk komersial. Jika ada pihak yang complain terhadap gambar tersebut, akan dihapus serta merta dalam ebook ini. Dan semoga gak ada yang *complain* :-D

Semua bentuk imbalan yang diterima dianggap sebagai Donasi. Meski hanya sebatas senyuman, adalah donasi kebaikan :)

Matur nuwun!

