Petunjuk:

1. Ujian bersifat open book dan dapat dibantu dengan komputer. Ujian dikerjakan sendiri dan tipe soal harus sesuai dengan nomor urut presensi, jika terdapat **kesamaan lebih dari 90% maka akan didiskualifikasi** bagi yang memberi jawaban dan yang meminta jawaban.
2. Soal dan jawaban harus dicetak

Soal:

1. Buatlah aplikasi Android tentang **Biografi Musik** menggunakan kelas **ContentProvider** yang mengakses database **SQLite** dengan relasi antar tabel sebagai berikut:

|  |  |
| --- | --- |
| **Tabel Lagu** | |
| **Field** | **Type** |
| Judul\_lagu | Text (100) |
| Nama\_pengarang \* | Text (50) |
| Nama\_perusahaan\_rekaman \* | Text (50) |
| Tahun\_pembuatan | Text (50) |

|  |  |
| --- | --- |
| **Tabel Pengarang** | |
| **Field** | **Type** |
| Nama\_pengarang \* | Text (50) |
| Tanggal lahir | Text (10) |
| Keterangan | text ( 200 ) |

|  |  |
| --- | --- |
| **Tabel PerusahaanRekaman** | |
| **Field** | **Type** |
| Nama\_perusahaan\_rekaman \* | text ( 100 ) |
| Tahun\_berdiri | text ( 10 ) |

\*= Foreign Key

Catatan: Satu pengarang dapat mencipta lebih dari 1 lagu

1. Aplikasi dapat melakukan operasi CRUD (create read update delete) pada database. Untuk operasi read melalukan query pencarian pengarang, lagu, perusahaan rekaman.
2. Soal dapat didownload di <https://github.com/mpriyonots/Mobile-Computing/>
3. Buatlah **laporan aplikasi** yang berisi source code ,flow chart dan algoritma aplikasi dan **presentasikan** paling lambat **sebelum** tanggal **12 Januari 2015**.

Petunjuk:

1. Ujian bersifat open book dan dapat dibantu dengan komputer. Ujian dikerjakan sendiri dan tipe soal harus sesuai dengan nomor urut presensi, jika terdapat **kesamaan lebih dari 90% maka akan didiskualifikasi** bagi yang memberi jawaban dan yang meminta jawaban.
2. Soal dan jawaban harus dicetak

Soal:

1. Buatlah aplikasi Android tentang **Repositori Game** menggunakan kelas **ContentProvider** yang mengakses database **SQLite** dengan relasi antar tabel sebagai berikut:

|  |  |
| --- | --- |
| **Tabel Game** | |
| **Field** | **Type** |
| Nama\_game | Text (100) |
| Tahun\_pembuatan | text ( 50 ) |
| Nama\_perusahaan\* | text ( 100 ) |
| Nama\_pengembang\* | text ( 100 ) |
| Keterangan | Text(500) |

|  |  |
| --- | --- |
| **Tabel PerusahaanPengembang** | |
| **Field** | **Type** |
| Nama\_perusahaan\* | numerik (6 ) |
| Tahun berdiri | text ( 100 ) |
| Keterangan | text ( 200 ) |

|  |  |
| --- | --- |
| **Tabel Pengembang** | |
| **Field** | **Type** |
| Nama\_pengembang\* | text ( 100) |
| email | text ( 100 ) |
| Kota | text ( 50) |

\*= Foreign Key

Catatan: Sebuah Game dapat dibuat oleh banyak pengembang/programmer seperti di bagian credit

1. Aplikasi dapat melakukan operasi CRUD (create read update delete) pada database. Untuk operasi read melalukan query nama game, nama pengembang, nama perusahaan.
2. Soal dapat didownload di <https://github.com/mpriyonots/Mobile-Computing/>
3. Buatlah **laporan aplikasi** yang berisi source code ,flow chart dan algoritma aplikasi dan **presentasikan** paling lambat **sebelum** tanggal **12 Januari 2015**