



دانشگاه تهران
پروژه برنامه سازی پیشرفته دانشکده آمار، ریاضی و علوم کامپیوتر مهلت ارسال: ۲۱ مرداد ۹۹

توضیحات کلی پروژه

پروژه شامل دو بخش دیتابیس یا پایگاه داده (database) و اپلیکیشن (Application) می باشد که باید با هم در ارتباط کامل باشند.

دیتابیس:

دیتابیس در واقع نوعی برنامه است به منظور نگهداری اطلاعات.
برنامه دیتابیس مربوط به پروژه شامل یک بخشی به صورت یک Shell که سری دستورات زیر را قبول میکند.

۱. INSERT
۲. DELETE
۳. UPDATE
۴. SELECT

Schema

هر دیتابیس برای اینکه داده ها را نگه دارد به این نیاز دارد که نوع اطلاعات درونی آن از پیش تعریف شود.

برای این کار یک فایل در نظر گرفته میشود که به آن فایل Schema میگویند.
در فایل Schema نیاز است که نامی که برای آن لیست داده در نظر میگیریم را تعریف کنیم و بعد از آن باید نوع داده های داخل آن تعریف بشوند.

در پروژه باید یک فایل به نام Schemas.txt وجود داشته باشد که در آن به فرم زیر اطلاعات وارد شوند.

```
<table1_name>  
<field1_name> <is_unique> <field1_type>  
<field2_name> <is_unique> <field2_type>  
<field3_name> <is_unique> <field3_type>  
  
<table2_name>  
<field1_name> <is_unique> <field1_type>  
<field2_name> <is_unique> <field2_type>  
<field3_name> <is_unique> <field3_type>
```

در مثال بالا برای نام table در واقع یک نام دلخواه برای نام داده ای که می خواهیم ذخیره کنیم را اختصاص میدهیم.

از طرفی باید تمامی مقادیر داخلی آن نیز در عنوان های field ذخیره شوند. که در کنار آنها نیز با یک فاصله باید یکتا بودن و یا نوع داده نیز تعریف شوند. انواع داده ها شامل CHAR، INTEGER، BOOLEAN، TIMESTAMP می باشند.

برای نوع داده CHAR باید ماکسیمم تعداد کاراکتر نیز تعریف میشود.

یکتا بودن یک نوع داده یعنی در هیچ جای داده های ذخیره شده نباید این مقدار عینا تکرار شود. در مثال پایین یک نمونه Schema دیده میشود که متغیر username به صورت UNIQUE نگه داشته میشود که نشانگر این است که یعنی هیچ خطی در نتایج دیتابیس نباید username تکراری دیده شود.

تایپ TIMESTAMP یک رشته حروف نمایانگر یک تاریخ میباشد.

```
user
username UNIQUE CHAR(200)
password CHAR(64)
joined_at TIMESTAMP
```

دستور INSERT

این دستور به منظور اضافه کردن اطلاعات استفاده میشود و به فرم زیر نوشته میشود

```
~$ INSERT INTO <table_name> VALUES (<field1_value>,<field2_value>,<field3_value> );
```

دستور SELECT

این دستور به منظور مشاهده داده ها تحت یک سری شرایط میباشد و برای اینکار از دستور WHERE استفاده میشود

شرایط فقط با == و یا != بررسی میشوند و چند شرط مختلف با پرانتز بندی و AND و یا OR تعریف میشود

```
~$ SELECT FROM <table_name> WHERE <field_name>=<field_value> OR <field_name>=<value>;
```

```
~$ SELECT FROM users WHERE username="Ehsan" OR phone="09990000000";
```

دستور DELETE

این دستور به منظور حذف اطلاعات از داخل دیتابیس می باشد.

```
~$ DELETE FROM <table_name> WHERE <field_name>=<field_value> OR <field_name>=<value>;
```

```
~$ DELETE FROM users WHERE username="Ehsan" OR phone="09990000000";
```

دستور UPDATE

این دستور به منظور تغییر اطلاعات داخل دیتابیس استفاده میشود و مقادیر داخل دیتابیس را با مقدار داده شده تحت شرایط خاص خود تغییر میدهد.

```
~$ UPDATE <table_name> WHERE <field_name>=<field_value> OR <field_name>=<value> VALUES (<field1_value>,<field2_value>,<field3_value> );
```

توجه:

وجود داشتن ; در انتهای دستور الزامیست.

محدودیتی در شرایط وجود ندارند و میتوانند داخل پرانتز باشند یا زنجیره ای عمل کنند.

در صورتی که دستور داده شده به هر فرمی اشتباه وارد شود باید ارور مربوط حتما اتفاق بیفتد.

هر table باید به صورت پیش فرض دارای یک متغیر یکتا به نام id باشد که مقدار آن یا از ۱ شروع میشود و به صورت صعودی رشد میکند برای هر row دیتابیس یا یک رشته کاراکتر یکتا ساخته شود.

هر table باید یک فایل از نوع text مربوط به خود داشته باشد که داخل آن مقادیر ذخیره شده دیتابیس داخل آن میرود برای مثال اگر فرض کنیم که یک table با نام users داشته باشیم و متغیرهای آن username و password باشند داخل فایل users.txt مانند زیر خواهد بود:

```
username password
ehsan ehsan
ahmad ahmad
ali pass1234
abbas nozari
```

اپلیکیشن:

اپلیکیشن پروژه به صورت بخش خلاقیتی اما اجباری پروژه میباشد و باید دارای چند ویژگی باشد:

به طور کلی این اپلیکیشن شبیه اپ معروف توییتر میباشد

۱.Login/Register

یک بخش برای ورود کاربران به داخل اپلیکیشن و یا ثبت نام کردن کاربران.

۲.Tweet

حال با کاربری که وارد اپ شدیم باید بتوانیم به متن خیلی ساده را به عنوان توییت ارسال کنیم

۳. View

بتوانیم تمامی توییت های ارسال شده را مشاهده کنیم

۴. Like

بتوانیم یک توییت را لایک کنیم و یا لیست افرادی که یک توییت را لایک کرده بودند را مشاهده کنیم

۵.Retweet

بتوانیم توییت یک کاربر را دوباره از طرف کاربری که وارد شدیم ارسال کنیم

توجه:

پیاده سازی اپلیکیشن به عهده خود دانشجو است ولی باید دارای ویژگی های بالا باشند و بتوان آنها را از هم تفکیک کرد. به طور پیشفرض انتظار می رود که از ترمینال برای نشان دادن اپ استفاده میشود اما استفاده از لایبرری هایی مثل Qt یا SFML یا هر لایبرری دیگری بلامانع است و از طرفی نمره امتیازی ندارد.

برای ارتباط با دیتابیس باید پروژه به صورت کلاینت و سرور پیاده سازی شود. روشهای پیشنهادی شامل، gRPC، Protobuf، Request HTTP میباشد اما استفاده از روشهای دیگر بلامانع است

پیاده سازی بخشهای اضافی به معنای پشتیبانی از قابلیتهای بیشتر طبیعتا نمره امتیازی دارد مانند همراه کردن چند دیتابیس و یا اضافه کردن کامنت به توییت یا لایک کردن کامنت و غیره