

توضیحات کلی پروژه :

در این پروژه ما یک سیستم خدمات بانکی کوچک را پیاده سازی خواهیم کرد. این برنامه متشکل از ۲ قسمت دیتابیس (database) و اپلیکیشن (application) می باشد که با یک دیگر ارتباط دارند.

دیتابیس:

دیتابیس در اصل نوعی از برنامه است که اطلاعات در آن ذخیره می شود. در این پروژه برنامه دیتابیس ما شامل یک بخش به صورت یک shell است که اطلاعات دستورات زیر را قبول می کند.

1. INSERT
2. DELETE
3. UPDATE
4. SELECT

: schema

هر دیتابیس برای این که داده ها را نگه دارد نیاز دارد که نوع اطلاعات درونی آن از پیش تعریف شود. که برای این کار از یک فایل به نام schema استفاده می کنیم.

در این فایل نیاز است نامی که برای آن لیست داده در نظر میگیریم را تعریف کنیم و همچنین نوع داده های داخل آن تعریف شود. پس در این پروژه باید یک فایل به نام schema.txt داشته باشیم که در آن اطلاعات به فرم زیر وارد شوند.

```
<table_name1>
<field_name1> <is_unique> <field_type1>
<field_name2> <is_unique> <field_type2>
<field_name3> <is_unique> <field_type3>

<table_name2>
<field_name1> <is_unique> <field_type1>
<field_name2> <is_unique> <field_type2>
<field_name3> <is_unique> <field_type3>
```

برای نام table در واقع یک نام دلخواه برای نام داده هایی که می خواهیم ذخیره کنیم را اختصاص می دهیم . همچنین همه مقادیر داخلی آن در عنوان های فیلد ذخیره می شوند.

سپس در کنار آنها نیز یکتا بودن یا نوع داده تعریف شود(به این صورت که اگر نوع داده بلافاصله بیاید، پس یکتا نیست). انواع داده ها شامل CHAR ,INTEGER, BOOLEAN, TIMESTAMP می باشد (برای نوع داده CHAR باید ماکزیمم تعداد کاراکتر نیز تعریف شود). اگر داده ای یکتا باشد یعنی در هیچ جای داده های ذخیره شده نباید این مقدار عینا تکرار شود. تایپ TIMESTAMP یک رشته حروف نمایانگر یک تاریخ می باشد.

User
username UNIQUE CHAR(200)
password CHAR(64)
joined_at TIMESTAMP

INSERT

این دستور به منظور اضافه کردن اطلاعات استفاده می شود و به فرم زیر وارد می شود

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

SELECT

این دستور به منظور مشاهده داده ها تحت یک سری شرایط می باشد و برای این کار از دستور WHERE استفاده می شود. شرایط فقط با == و یا != بررسی می شوند و چند شرط مختلف با پرانتز بندی و AND و یا OR تعریف می شود.

```
$ SELECT FROM <table_name> WHERE <field_name>==<field_value> OR <field_name>==<field_value>;  
$ SELECT FROM users WHERE username=="eminem" OR phone=="09171656786";
```

DELETE

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

```
$ DELETE FROM <table_name> WHERE <field_name>==<field_value> OR <field_name>==<field_value>;  
$ DELETE FROM users WHERE username=="eminem" OR phone=="09171656786";
```

UPDATE

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

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

نکته:

وجود داشتن ; در انتهای دستور الزامی است.

محدودیتی در شرایط وجود ندارد و می تواند 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 در دیتابیس باید به صورت پیشفرض دارای یک متغیر یکتا به نام id باشد که از ۱ شروع می شود و به صورت صعودی برای هر ردیف رشد می کند و یا می تواند یک رشته کاراکتر یکتا باشد.

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

Username password
Setareh 1234
Saman 5678
Erfan 9000

اپلیکیشن:

سیستم شما باید خدمات ذیل را پشتیبانی کند:

1. **عضویت کاربر جدید:** کاربران جدید می توانند با وارد کردن نام ، کد ملی، رمز عبور، شماره تلفن و ایمیل در سیستم عضو شوند.
2. **ورود به برنامه:** کاربران از طریق کد ملی و رمز عبور می توانند وارد سیستم شده و پس از انجام کارهای خود، از آن خارج شوند.
3. **افتتاح حساب:** کاربر می تواند با مشخص کردن alias حساب و رمز عبور آن حسابی جدید افتتاح کند.
4. **مشاهده اطلاعات حساب ها:** این اطلاعات شامل نوع آن، موجودی و لیست تراکنش ها می شوند.
5. **مدیریت حساب:** کاربر می تواند حساب جدید باز کند (حساب جدید یک رمز نیز دارد که برای انتقال وجه از آن استفاده می شود) و اطلاعات مربوط به حساب فعلی خود را مشاهده کنند. همچنین می توانند برای حساب های خود alias تعیین کنند.
6. **تعریف حساب های پرکاربرد:** کاربر می تواند از شماره حسابهایی که زیاد استفاده می کند يك لیست به همراه يك alias برای آن شماره حساب ایجاد نماید.
7. **انتقال وجه:** کاربران با مشخص کردن مبلغ، حساب مقصد (کاربر می تواند مقصد را از لیست حساب های پرکاربرد انتخاب کند) و رمز حساب میتوانند به حساب خود و یا کاربران دیگر وجه منتقل کند.
8. **پرداخت قبض:** کاربر می تواند با وارد کردن شناسه قبض و شناسه پرداخت آن، مبلغ قبض را پرداخت کند.
9. **درخواست وام:** کاربر میتواند با انتخاب مبلغ و دوره پرداخت از سیستم وام بگیرد. مبلغ وام به حساب وی واریز شده و قسط های آن به صورت دوره ای از حسابش کسر خواهند شد.
10. **بستن حساب:** کاربر باید بتواند با وارد کردن رمز حساب خود را ببندد. در صورت خالی نبودن حساب، سیستم باید يك شماره حساب مقصد برای واریز موجودی فعلی از کاربر درخواست کند