/\*addPost({

id: '21',

description: 'Just something',

createdAt: new Date('2018-02-03T12:00:00'),

author: 'Lanny',

photoLink: 'https://pp.userapi.com/c841537/v841537399/78135/Z\_1P54xxqdc.jpg',

hashTags : ['#someHashTags'],

likes : ['Man', 'Girl']

});\*/

//removePost('1');

//editPost('2', {description : 'Gruyeres is really fantastic town!!!'});

//showPosts(0,20);

<section class="search-results">  
 <div class ="search-title">  
 <h2>Gallery</h2>  
 </div>  
 </section>  
</section>

<div class="photos">  
 <div class="result circle-block">  
 <div class="result-left">  
 <div class="result-img">  
 <img src="assets/img/image-after.jpg" alt="">  
 </div>  
 <div class="result-description">  
 <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. <p>  
 </div>  
 </div>  
 <div class="result-right">  
  
 <h4>Your photo</h4>  
 <ul>  
 <li>#list</li>  
 <li>#of</li>  
 <li>#hashtags</li>  
 <li>  
 <div class="post-description">  
 <div class="time">  
 <i class="fa fa-clock-o"></i>  
 <p>Time</p>  
 </div>  
 <div class="date">  
 <i class="fa fa-calendar"></i>  
 <p>Date</p>  
 </div>  
 </div>  
 </li>  
 </ul>  
  
 <div class="post-likes">  
 <p>N</p>  
 <i class="fa fa-heart"></i>  
 </div>  
 <div class="edit">  
 <i class="fa fa-edit"></i>  
 <i class="fa fa-trash-o"></i>  
 </div>  
  
  
 </div>  
 </div>  
 <div class="result circle-block">  
 <div class="result-left">  
 <div class="result-img">  
 <img src="assets/img/image-after.jpg" alt="">  
  
 </div>  
 <div class="result-description">  
 <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. <p>  
 </div>  
 </div>  
 <div class="result-right">  
  
 <h4>Your photo</h4>  
  
  
  
 <ul>  
 <li>#list</li>  
 <li>#of</li>  
 <li>#hashtags</li>  
 <li>  
 <div class="post-description">  
 <div class="time">  
 <i class="fa fa-clock-o"></i>  
 <p>Time</p>  
 </div>  
 <div class="date">  
 <i class="fa fa-calendar"></i>  
 <p>Date</p>  
 </div>  
 </div>  
 </li>  
 </ul>  
  
 <div class="post-likes">  
 <p>N</p>  
 <i class="fa fa-heart"></i>  
 </div>  
 <div class="edit">  
 <i class="fa fa-edit"></i>  
 <i class="fa fa-trash-o"></i>  
 </div>  
  
  
 </div>  
 </div>  
 <div class="result circle-block">  
 <div class="result-left">  
 <div class="result-img">  
 <img src="assets/img/image-after.jpg" alt="">  
  
 </div>  
 <div class="result-description">  
 <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. <p>  
 </div>  
 </div>  
 <div class="result-right">  
  
 <h4>Your photo</h4>  
  
  
  
 <ul>  
 <li>#list</li>  
 <li>#of</li>  
 <li>#hashtags</li>  
 <li>  
 <div class="post-description">  
 <div class="time">  
 <i class="fa fa-clock-o"></i>  
 <p>Time</p>  
 </div>  
 <div class="date">  
 <i class="fa fa-calendar"></i>  
 <p>Date</p>  
 </div>  
 </div>  
 </li>  
 </ul>  
  
 <div class="post-likes">  
 <p>N</p>  
 <i class="fa fa-heart"></i>  
 </div>  
 <div class="edit">  
 <i class="fa fa-edit"></i>  
 <i class="fa fa-trash-o"></i>  
 </div>  
  
  
 </div>  
 </div>  
  
 <div class="result circle-block">  
 <div class="result-left">  
 <div class="result-img">  
 <img src="assets/img/image-after.jpg" alt="">  
  
 </div>  
 <div class="result-description">  
 <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. <p>  
 </div>  
 </div>  
 <div class="result-right">  
  
 <h4>Your photo</h4>  
  
  
  
 <ul>  
 <li>#list</li>  
 <li>#of</li>  
 <li>#hashtags</li>  
 <li>  
 <div class="post-description">  
 <div class="time">  
 <i class="fa fa-clock-o"></i>  
 <p>Time</p>  
 </div>  
 <div class="date">  
 <i class="fa fa-calendar"></i>  
 <p>Date</p>  
 </div>  
 </div>  
 </li>  
 </ul>  
  
 <div class="post-likes">  
 <p>N</p>  
 <i class="fa fa-heart"></i>  
 </div>  
 <div class="edit">  
 <i class="fa fa-edit"></i>  
 <i class="fa fa-trash-o"></i>  
 </div>  
  
  
 </div>  
 </div>  
  
 <div class="result circle-block">  
 <div class="result-left">  
 <div class="result-img">  
 <img src="assets/img/image-after.jpg" alt="">  
  
 </div>  
 <div class="result-description">  
 <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. <p>  
 </div>  
 </div>  
 <div class="result-right">  
  
 <h4>Your photo</h4>  
  
  
  
 <ul>  
 <li>#list</li>  
 <li>#of</li>  
 <li>#hashtags</li>  
 <li>  
 <div class="post-description">  
 <div class="time">  
 <i class="fa fa-clock-o"></i>  
 <p>Time</p>  
 </div>  
 <div class="date">  
 <i class="fa fa-calendar"></i>  
 <p>Date</p>  
 </div>  
 </div>  
 </li>  
 </ul>  
  
 <div class="post-likes">  
 <p>N</p>  
 <i class="fa fa-heart"></i>  
 </div>  
 <div class="edit">  
 <i class="fa fa-edit"></i>  
 <i class="fa fa-trash-o"></i>  
 </div>  
  
  
 </div>  
 </div>  
 <div class="result circle-block">  
 <div class="result-left">  
 <div class="result-img">  
 <img src="assets/img/image-after.jpg" alt="">  
  
 </div>  
 <div class="result-description">  
 <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. <p>  
 </div>  
 </div>  
 <div class="result-right">  
  
 <h4>Your photo</h4>  
  
  
  
 <ul>  
 <li>#list</li>  
 <li>#of</li>  
 <li>#hashtags</li>  
 <li>  
 <div class="post-description">  
 <div class="time">  
 <i class="fa fa-clock-o"></i>  
 <p>Time</p>  
 </div>  
 <div class="date">  
 <i class="fa fa-calendar"></i>  
 <p>Date</p>  
 </div>  
 </div>  
 </li>  
 </ul>  
  
 <div class="post-likes">  
 <p>N</p>  
 <i class="fa fa-heart"></i>  
 </div>  
 <div class="edit">  
 <i class="fa fa-edit"></i>  
 <i class="fa fa-trash-o"></i>  
 </div>  
  
  
 </div>  
 </div>  
 </div>  
  
 <div class="button-load">  
 <button>Load more..</button>  
 </div>  
  
 </section>  
  
 </section>  
  
  
 <section class="footer circle-block">  
 <div class="author">  
 <p>© Yastrebova Veronika</p><p>veronikayastr@gmail.com</p><p>last change: 26.02.2018</p></p>  
 </div>  
  
 <div class="univer">  
 <p>2nd course <br>13th group</br></p>  
 </div>  
  
 </section>  
  
</div>

|  |
| --- |
| void AddMoneyQuery(sqlite3\* db) |
|  | { |
|  | int cardNumber = -1; |
|  | int addSum = -1; |
|  | int currentSum = -1; |
|  | char sql[1024]; |
|  | char buf[256]; |
|  | int rc = -1; |
|  | char\* zErrMsg; |
|  |  |
|  | while (!ACEnterCardNumber(&cardNumber)); |
|  | if (!ACCardNumberExists(db, cardNumber)) |
|  | { |
|  | printf\_s("ERROR: Card with such number was not found!\n"); |
|  | return; |
|  | } |
|  | while (!ACEnterAddSum(&addSum)); |
|  |  |
|  | strcpy\_s(sql, 1024, "SELECT balance FROM accounts where cards\_id=(SELECT id FROM cards WHERE number=\""); |
|  | \_itoa\_s(cardNumber, buf, 256, 10); |
|  | strcat\_s(sql, 1024, buf); |
|  | strcat\_s(sql, 1024, "\");"); |
|  |  |
|  | rc = sqlite3\_exec(db, sql, GetSumCallback, &currentSum, &zErrMsg); |
|  | if (rc != SQLITE\_OK) |
|  | { |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); |
|  | sqlite3\_free(zErrMsg); |
|  | } |
|  | if (currentSum == -1) |
|  | { |
|  | printf\_s("ERROR: This card is not connected to account!\n"); |
|  | return; |
|  | } |
|  |  |
|  | strcpy\_s(sql, 1024, ""); |
|  | strcpy\_s(sql, 1024, "UPDATE accounts SET balance = "); |
|  | \_itoa\_s(addSum + currentSum, buf, 256, 10); |
|  | strcat\_s(sql, 1024, buf); |
|  | strcat\_s(sql, 1024, " WHERE cards\_id=(SELECT id FROM cards WHERE number=\""); |
|  | \_itoa\_s(cardNumber, buf, 256, 10); |
|  | strcat\_s(sql, 1024, buf); |
|  | strcat\_s(sql, 1024, "\");"); |
|  |  |
|  | rc = sqlite3\_exec(db, sql, 0, 0, &zErrMsg); |
|  | if (rc != SQLITE\_OK) |
|  | { |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); |
|  | sqlite3\_free(zErrMsg); |
|  | } |

}

|  |
| --- |
| void DebitMoneyQuery(sqlite3\* db) |
|  | { | |
|  | int cardNumber = -1; | |
|  | int debitSum = -1; | |
|  | int currentSum = -1; | |
|  | char sql[1024]; | |
|  | char buf[256]; | |
|  | int rc = -1; | |
|  | char\* zErrMsg; | |
|  |  | |
|  | while (!ACEnterCardNumber(&cardNumber)); | |
|  | if (!ACCardNumberExists(db, cardNumber)) | |
|  | { | |
|  | printf\_s("ERROR: Card with such number was not found!\n"); | |
|  | return; | |
|  | } | |
|  | while (!ACEnterDebitSum(&debitSum)); | |
|  |  | |
|  | strcpy\_s(sql, 1024, "SELECT balance FROM accounts where cards\_id=(SELECT id FROM cards WHERE number=\""); | |
|  | \_itoa\_s(cardNumber, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, "\");"); | |
|  |  | |
|  | rc = sqlite3\_exec(db, sql, GetSumCallback, &currentSum, &zErrMsg); | |
|  | if (rc != SQLITE\_OK) | |
|  | { | |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); | |
|  | sqlite3\_free(zErrMsg); | |
|  | } | |
|  | if (currentSum == -1) | |
|  | { | |
|  | printf\_s("ERROR: This card is not connected to account!\n"); | |
|  | return; | |
|  | } | |
|  |  | |
|  | int over = 1; | |
|  | if (debitSum > currentSum) { | |
|  | over = Overdraft(db, debitSum, cardNumber, currentSum); | |
|  | } | |
|  |  | |
|  | if (over == 1) { | |
|  |  | |
|  | strcpy\_s(sql, 1024, "SELECT balance FROM accounts where cards\_id=(SELECT id FROM cards WHERE number=\""); | |
|  | \_itoa\_s(cardNumber, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, "\");"); | |
|  |  | |
|  | rc = sqlite3\_exec(db, sql, GetSumCallback, &currentSum, &zErrMsg); | |
|  | if (rc != SQLITE\_OK) | |
|  | { | |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); | |
|  | sqlite3\_free(zErrMsg); | |
|  | } | |
|  |  | |
|  | strcpy\_s(sql, 1024, ""); | |
|  | strcpy\_s(sql, 1024, "UPDATE accounts SET balance = "); | |
|  | \_itoa\_s(currentSum - debitSum, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, " WHERE cards\_id=(SELECT id FROM cards WHERE number=\""); | |
|  | \_itoa\_s(cardNumber, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, "\");"); | |
|  |  | |
|  | rc = sqlite3\_exec(db, sql, 0, 0, &zErrMsg); | |
|  | if (rc != SQLITE\_OK) | |
|  | { | |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); | |
|  | sqlite3\_free(zErrMsg); | |
|  | } | |
|  | } | |
|  | } | |
|  |  | |
|  | void TransferMoneyQuery(sqlite3\* db) { | |
|  | int cardNumber1 = -1; | |
|  | int cardNumber2 = -1; | |
|  | int debitSum = -1; | |
|  | int currentSum1 = -1; | |
|  | char sql[1024]; | |
|  | char buf[256]; | |
|  | int rc = -1; | |
|  | char\* zErrMsg; | |
|  |  | |
|  | while (!ACEnterCardNumber(&cardNumber1)); | |
|  | if (!ACCardNumberExists(db, cardNumber1)) | |
|  | { | |
|  | printf\_s("ERROR: Card with such number was not found!\n"); | |
|  | return; | |
|  | } | |
|  | while (!ACEnterDebitSum(&debitSum)); | |
|  |  | |
|  | strcpy\_s(sql, 1024, "SELECT balance FROM accounts where cards\_id=(SELECT id FROM cards WHERE number=\""); | |
|  | \_itoa\_s(cardNumber1, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, "\");"); | |
|  |  | |
|  | rc = sqlite3\_exec(db, sql, GetSumCallback, &currentSum1, &zErrMsg); | |
|  | if (rc != SQLITE\_OK) | |
|  | { | |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); | |
|  | sqlite3\_free(zErrMsg); | |
|  | } | |
|  | if (currentSum1 == -1) | |
|  | { | |
|  | printf\_s("ERROR: This card is not connected to account!\n"); | |
|  | return; | |
|  | } | |
|  |  | |
|  | int currentSum2 = -1; | |
|  | while (!ACEnterCardNumber(&cardNumber2)); | |
|  | if (!ACCardNumberExists(db, cardNumber2)) | |
|  | { | |
|  | printf\_s("ERROR: Card with such number was not found!\n"); | |
|  | return; | |
|  | } | |
|  |  | |
|  | strcpy\_s(sql, 1024, "SELECT balance FROM accounts where cards\_id=(SELECT id FROM cards WHERE number=\""); | |
|  | \_itoa\_s(cardNumber2, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, "\");"); | |
|  |  | |
|  | rc = sqlite3\_exec(db, sql, GetSumCallback, &currentSum2, &zErrMsg); | |
|  | if (rc != SQLITE\_OK) | |
|  | { | |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); | |
|  | sqlite3\_free(zErrMsg); | |
|  | } | |
|  | if (currentSum2 == -1) | |
|  | { | |
|  | printf\_s("ERROR: This card is not connected to account!\n"); | |
|  | return; | |
|  | } | |
|  |  | |
|  | int over = 1; | |
|  | if (debitSum > currentSum1) { | |
|  | over = Overdraft(db, debitSum, cardNumber1, currentSum1); | |
|  | } | |
|  |  | |
|  | if (over == 1) { | |
|  | strcpy\_s(sql, 1024, ""); | |
|  | strcpy\_s(sql, 1024, "UPDATE accounts SET balance = "); | |
|  | \_itoa\_s(currentSum1 - debitSum, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, " WHERE cards\_id=(SELECT id FROM cards WHERE number=\""); | |
|  | \_itoa\_s(cardNumber1, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, "\");"); | |
|  |  | |
|  | rc = sqlite3\_exec(db, sql, 0, 0, &zErrMsg); | |
|  | if (rc != SQLITE\_OK) | |
|  | { | |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); | |
|  | sqlite3\_free(zErrMsg); | |
|  | } | |
|  |  | |
|  |  | |
|  | strcpy\_s(sql, 1024, ""); | |
|  | strcpy\_s(sql, 1024, "UPDATE accounts SET balance = "); | |
|  | \_itoa\_s(debitSum + currentSum2, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, " WHERE cards\_id=(SELECT id FROM cards WHERE number=\""); | |
|  | \_itoa\_s(cardNumber2, buf, 256, 10); | |
|  | strcat\_s(sql, 1024, buf); | |
|  | strcat\_s(sql, 1024, "\");"); | |
|  |  | |
|  | rc = sqlite3\_exec(db, sql, 0, 0, &zErrMsg); | |
|  | if (rc != SQLITE\_OK) | |
|  | { | |
|  | printf\_s("SQL ERROR: %s\n", zErrMsg); | |
|  | sqlite3\_free(zErrMsg); | |
|  | } | |
|  | } | |
|  | } | |
| void TransferBetweenAccounts(sqlite3\* db) { | |
|  | | printf\_s("\tTRANSFER\n"); |
|  | | TransferMoneyQuery(db); |
|  | | } |