










Ссылка на задачу	 <a href="#">JUST_DO_IT_SPRINT</a>  Jira
Тестируемое веб-приложение	 <a href="#">Bank Saint-Petersburg</a> BSPB internet-bank for private customers demo.bspb.ru

## BUG-REPORT

Описание	Bug-report
При отправке письма банку происходит бесконечная загрузка	 JUS-375: При отправке письма банку происходит бесконечная загрузка  DONE
"Обрезалась" правая сторона страницы (проблема с версткой) на устройстве iPad Air	 JUS-379: "Обрезалась" правая сторона страницы (проблема с версткой) на устройстве iPad Air  DONE
В блоке "Сервисы для бизнеса" поле ИНН принимает НЕ только цифры	 JUS-396: В блоке "Сервисы для бизнеса" поле ИНН принимает НЕ только цифры  DONE

## TEST CASE

Описание	Test-case
Открытие долларового счета в банке "БСПБ"	<a href="https://app.qase.io/case/GFJDI-94">https://app.qase.io/case/GFJDI-94</a>
Проверка валидности гиперссылки у иконки "Вконтакте"	<a href="https://app.qase.io/case/GFJDI-80">https://app.qase.io/case/GFJDI-80</a>
Заказ новой банковской карты "Business"	<a href="https://app.qase.io/case/GFJDI-75">https://app.qase.io/case/GFJDI-75</a>

## ДОКУМЕНТАЦИЯ К ПРОДУКТУ


- Требования к форме аутентификации
  - Уникальность идентификатора
  - Маскирование пароля
  - Логин и пароль — на одной странице
  - Одноразовые пароли
  - Восстановление доступа
  - Легкость использования
  - Блокировка пользователя при трех неверных попытках входа


▼ Требования к паролю

1. Пароль должен содержать не менее 4 символов.
2. Ограничение на максимальную длину пароля должно быть не менее 12 символов.
3. Пароль должен состоять из букв латинского алфавита (A-z), арабских цифр (0-9) и специальных символов, приведенных в п. 5 данных требований.
4. Буквенная часть пароля должна содержать как строчные, так и прописные (заглавные) буквы.
5. Допускается наличие следующих символов: ~ ! ? @ # \$ % ^ & \* \_ - + ( ) [ ] { } > < / \ | " ' . , : ;
6. Блокировка повторяющихся символов
7. Запрет использования персональной информации
8. Передача пароля методом POST

▼ Требования к форме "Окно ввода кода подтверждения входа":

1. Поле для ввода кода подтверждения.
2. Код подтверждения должен состоять из 4 символов и содержать только цифры
3. Кнопка для отправки введенного кода.
4. Возможность запроса нового кода.
5. Отображение оставшегося времени действия кода.
6. Подсказка или инструкция по вводу кода.
7. Отображение сообщения об успешной или неуспешной аутентификации.
8. В случае неверного кода появится сообщение красного цвета "Неверный код"
9. Дизайн и расположение элементов для удобства использования.



 Неверный код

Отправили SMS код на ваш номер телефона

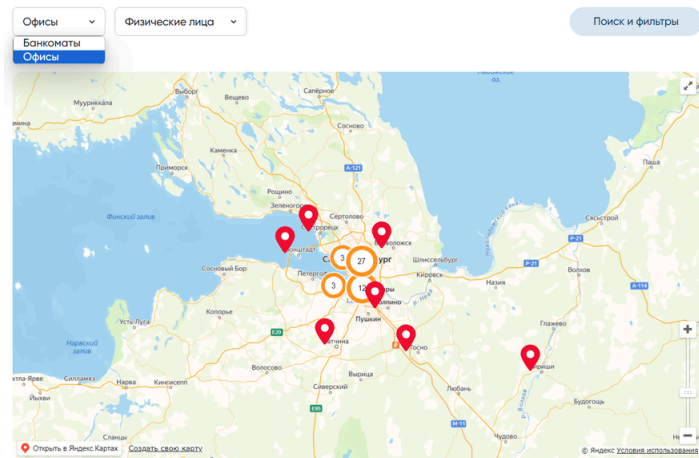
0000
 ↻ Снова через 110 сек

Подтвердить

[Вернуться к входу](#)

▼ Требования для страницы "Офисы и банкоматы" сайта банка БСПБ

1. Информация о расположении: на странице должна быть представлена подробная информация о местоположении всех офисов и банкоматов банка, включая адреса, контактные телефоны, часы работы и карту с маркерами.
2. Фильтрация и поиск: пользователи должны иметь возможность фильтровать офисы и банкоматы по различным критериям (например, по городу, району, часам работы) и использовать поиск для быстрого нахождения нужной информации.
3. Интерактивная карта: на странице должна быть представлена интерактивная карта с отмеченными на ней офисами и банкоматами для удобства пользователей.
4. Удобство использования на мобильных устройствах: страница должна быть адаптирована для просмотра на мобильных устройствах, чтобы пользователи могли быстро найти нужную информацию в любое время.
5. Обновление информации: информация о расположении офисов и банкоматов должна быть актуальной и регулярно обновляться, чтобы предоставлять пользователям достоверные данные.
6. Дополнительная информация: на странице можно также предоставить дополнительную информацию о услугах, доступных в офисах банка, инструкции по пользованию банкоматами и т.д.



▼ Требования к разделу FAQ

1. Актуальность и полнота информации для удовлетворения потребностей пользователей.
2. Структурированность и упорядоченность: Структурирование вопросов и ответов по категориям или темам для удобства поиска информации.
3. Ясность и понятность: Формулировка ответов на вопросы простым и понятным языком, чтобы пользователи могли легко понять предоставленную информацию.
4. Поиск и навигация: Обеспечение возможности быстрого поиска нужной информации в разделе FAQ с помощью функции поиска или удобной навигации.
5. Обновление и модерация: Регулярное обновление и проверка FAQ на наличие актуальной информации, а также модерация комментариев или вопросов пользователей.
6. Ответы на популярные вопросы: Включение ответов на наиболее часто задаваемые вопросы, чтобы предоставить пользователям быструю помощь без необходимости обращения в службу поддержки.
7. Интерактивность и обратная связь: Предоставление возможности пользователям задавать свои вопросы непосредственно в разделе FAQ или оставлять комментарии для получения дополнительной информации.

## Часто задаваемые вопросы

Физическим лицам Юридическим лицам (ИП) Котировальная доска

## 1. Где найти платёж, если я не вижу его в Истории?

Все истории ваших платежей, переводов и операций по карте хранятся в Выписке. Чтобы платёж не пропал из Истории, отметьте его галочкой **Сохранить**.

## 2. Как распечатать детали платежа, которого сейчас нет в Истории?

Сохраните нулевой платёж в качестве шаблона в старом интернет-банке, и в течение дня он появится в Истории в новой системе. Перенесённый шаблон не пропадет из Истории до тех пор, пока вы не снимите с него галочку **Сохранить**. Таким образом, можно будет просмотреть или распечатать детали этого платежа в любой момент, когда появится необходимость.

## 3. Какие операции подтверждаются SMS кодом?

Все операции, за исключением перевода между своими счетами и обмена валюты, где денежные средства перемещаются между счетами одного клиента. Пополнение вклада равносильно переводу средств между своими счетами и поэтому тоже не требует подтверждения SMS кодом.

## 4. Чем различаются значения в графах Остаток и Доступный остаток?

Остаток - это ваши средства на банковском счете. Доступный остаток - это средства, которые вы можете потратить, включая заемные средства.

## 5. Что такое заблокированная сумма?

В графе **Заблокировано** отображаются суммы, израсходованные (или внесенные) по карте, но еще не отраженные на вашем банковском счете. Таким образом, вы видите в интернет-банке эти денежные средства, но воспользоваться ими не можете.

## 6. Где посмотреть выписку по переводам до 2012 года?

Выписку по операциям совершенным ранее 1 января 2012г. можно получить в любом офисе банка.

## Требования к форме обратной связи

1. Простота использования: форма должна быть легко заполняемой и интуитивно понятной для пользователей.
2. Обязательные поля: необходимо определить, какие поля обязательны для заполнения, чтобы получить достаточно информации от пользователя.
3. Подтверждение отправки: после заполнения формы пользователь должен получить подтверждение о том, что его сообщение было успешно отправлено.
4. Возможность прикрепления файлов: если требуется отправка дополнительных материалов, форма должна предоставлять возможность прикрепить файлы.
5. Адаптивный дизайн: форма должна корректно отображаться на различных устройствах и экранах.

Оцените Интернет-банк

Нравится

Хорошо

Не нравится

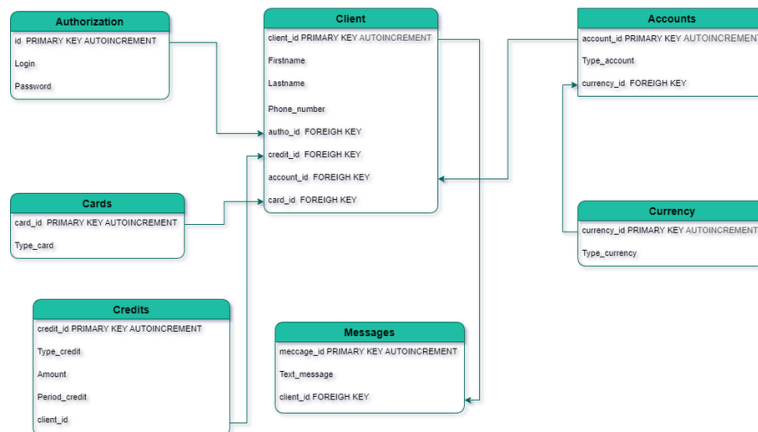
Комментарий (Дополнительно)

Отвечить позже

Отправить ответ

Спасибо за ваш отзыв!

## DATABASE BSPB



## MySQL tables

```
mysql> select *from authorization;
+----+-----+-----+
| id | Login   | Password |
+----+-----+-----+
| 1  | twix22  | tjv826pl |
| 2  | bounty20 | Pra71lNf |
| 3  | snikers30 | tv80Jvw |
| 4  | Mars30  | FT7160n |
+----+-----+-----+
4 rows in set (0.00 sec)

mysql> select *from client;
+----+-----+-----+-----+-----+-----+-----+
| client_id | First_name | Last_name | Phone_number | autho_id | credit_id | account_id | card_id |
+----+-----+-----+-----+-----+-----+-----+
| 1  | Джеймс   | Бонд     | 89822220125 | 1        | 1        | 2        | 2        |
| 2  | Индиана  | Джонс    | 89822230125 | 2        | 3        | 1        | 2        |
| 3  | Тарзан   | Иванов   | 89812330000 | 3        | 2        | 3        | 3        |
| 4  | Робин    | Гуд      | 89852550005 | 4        | 4        | 4        | 4        |
+----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select *from cards;
+----+-----+
| card_id | Type_card |
+----+-----+
| 1  | Travel    |
| 2  | Детская   |
| 3  | Премиальная карта BLACK |
| 4  | Единая карта петербуржца |
+----+-----+
4 rows in set (0.00 sec)

mysql> select *from credits;
+----+-----+-----+-----+-----+
| credit_id | Type_credit | Amount | Period_credit | client_id |
+----+-----+-----+-----+-----+
| 1  | Ипотечный  | 200000000 | 20 лет        | 2        |
| 2  | Автострахование | 2000000 | 5 лет         | 2        |
| 3  | Потребительский | 500000 | 5 лет         | 4        |
| 4  | Ипотечный  | 500000000 | 20 лет        | 1        |
+----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> select *from accounts;
+----+-----+-----+
| account_id | Type_account | currency_id |
+----+-----+-----+
| 1  | Зарплатный  | 2          |
| 2  | Накопительный | 1          |
| 3  | Брокерский  | 2          |
| 4  | Накопительный | 2          |
+----+-----+-----+
4 rows in set (0.00 sec)

mysql> select *from currency;
+----+-----+
| currency_id | Type_currency |
+----+-----+
| 1  | RUB           |
| 2  | USD           |
| 3  | EUR           |
| 4  | JPY           |
+----+-----+
4 rows in set (0.00 sec)

mysql> select *from messages;
+----+-----+-----+
| Message_id | Text_message | client_id |
+----+-----+-----+
| 1  | За что списали деньги? | 2        |
| 2  | Дайте больше кредит, я даю деньги бедным! | 4        |
| 3  | Хотел сказать, что я - Джеймс.. Джеймс Бонд.. | 1        |
| 4  | Да я не муж Королевой!.. | 3        |
+----+-----+-----+
4 rows in set (0.00 sec)
```

#### ▼ Create tables

```
1 CREATE TABLE Autorization (
2 id INT PRIMARY KEY AUTO_INCREMENT,
3 Login VARCHAR(10),
4 Password VARCHAR(10)
5 );
6
7 CREATE TABLE Client (
8 client_id INT PRIMARY KEY AUTO_INCREMENT,
9 First_name VARCHAR(100),
10 Last_name VARCHAR(100),
11 Phone_number VARCHAR(50),
12 autho_id INT REFERENCES Autorization(id),
13 credit_id INT REFERENCES Credits(credit_id),
14 account_id INT REFERENCES Accounts(account_id),
15 card_id INT REFERENCES Cards(card_id)
16 );
17
18 CREATE TABLE Credits(
19 credit_id INT PRIMARY KEY AUTO_INCREMENT,
20 Type_credit VARCHAR(50) NOT NULL,
21 Amount INT,
22 Period_credit VARCHAR(50),
23 client_id INT REFERENCES Client (client_id)
24 );
25
```

```

26 CREATE TABLE Cards (
27 card_id INT PRIMARY KEY AUTO_INCREMENT,
28 Type_card VARCHAR(100)
29 );
30
31 CREATE TABLE Accounts(
32 account_id INT PRIMARY KEY AUTO_INCREMENT,
33 Type_account VARCHAR(50) NOT NULL,
34 currency_id INT REFERENCES Currency (currency_id)
35 );
36
37 CREATE TABLE Currency (
38 currency_id INT PRIMARY KEY AUTO_INCREMENT,
39 Type_currency VARCHAR(50)
40 );
41
42 CREATE TABLE Messages (
43 Message_id INT PRIMARY KEY AUTO_INCREMENT,
44 Text_message VARCHAR(200),
45 client_id INT REFERENCES client (client_id)
46 );

```

▼ Insert into..

#### Autorization

```

mysql> INSERT INTO authorization (id, Login, Password) VALUES ( 1, 'twix22', 'tjV826pl');

mysql> INSERT INTO authorization (id, Login, Password) VALUES ( 2, 'bounty20', 'Pra71LNf');

mysql> INSERT INTO authorization (id, Login, Password) VALUES ( 3, 'snikers30', 'tv80Jvw');

mysql> INSERT INTO authorization (id, Login, Password) VALUES ( 4, 'Mars30', 'FT716On');

```

#### Client

```

mysql> INSERT INTO client (client_id, First_name, Last_name, Phone_number, autho_id, credit_id, account_id, card_id) VALUES (1, 'Джеймс', 'Бонд', '89822320125', 1, 1, 2, 2);
mysql> INSERT INTO client (client_id, First_name, Last_name, Phone_number, autho_id, credit_id, account_id, card_id) VALUES (2, 'Индиана', 'Джонс', '89832330125', 2, 3, 1, 2);
mysql> INSERT INTO client (client_id, First_name, Last_name, Phone_number, autho_id, credit_id, account_id, card_id) VALUES (3, 'Тарзан', 'Иванов', '89832330000', 3, 2, 3, 3);
mysql> INSERT INTO client (client_id, First_name, Last_name, Phone_number, autho_id, credit_id, account_id, card_id) VALUES (4, 'Робин', 'Гуд', '89852550005', 4, 4, 4, 4);

```

#### Credits

```

mysql> INSERT INTO credits (credit_id, Type_credit, Amount, Period_credit, client_id) VALUES (1, 'Ипотечный', 200000000, '20 лет', 2);

mysql> INSERT INTO credits (credit_id, Type_credit, Amount, Period_credit, client_id) VALUES (2, 'Австрахование', 2000000, '5 лет', 2);

mysql> INSERT INTO credits (credit_id, Type_credit, Amount, Period_credit, client_id) VALUES (3, 'Потребительский', 500000, '5 лет', 4);

mysql> INSERT INTO credits (credit_id, Type_credit, Amount, Period_credit, client_id) VALUES (4, 'Ипотечный', 50000000, '20 лет', 1);

```

#### Currency

```

mysql> INSERT INTO currency (currency_id, Type_currency) VALUES (1, 'RUB');

mysql> INSERT INTO currency (currency_id, Type_currency) VALUES (2, 'USD');

mysql> INSERT INTO currency (currency_id, Type_currency) VALUES (3, 'EUR');

mysql> INSERT INTO currency (currency_id, Type_currency) VALUES (4, 'JPY');

```

#### Accounts

```

mysql> INSERT INTO accounts (account_id, Type_account, currency_id) VALUES (1, 'Зарплатный', 2);

mysql> INSERT INTO accounts (account_id, Type_account, currency_id) VALUES (2, 'Накопительный', 1);

mysql> INSERT INTO accounts (account_id, Type_account, currency_id) VALUES (3, 'Брокерский', 1);

mysql> INSERT INTO accounts (account_id, Type_account, currency_id) VALUES (4, 'Накопительный', 2);

```

#### Cards

```

mysql> INSERT INTO cards (card_id, Type_card) VALUES (1, 'Travel');

mysql> INSERT INTO cards (card_id, Type_card) VALUES (2, 'Детская');

mysql> INSERT INTO cards (card_id, Type_card) VALUES (3, 'Премиальная карта BLACK');

mysql> INSERT INTO cards (card_id, Type_card) VALUES (4, 'Единая карта петербуржца');

```

#### Messages

```

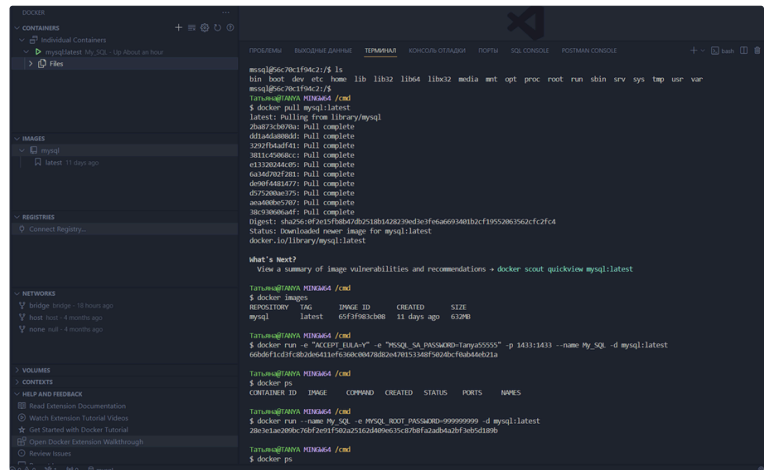
mysql> INSERT INTO messages (Message_id, Text_message, client_id) VALUES (1, 'За что списали деньги?', 2);

mysql> INSERT INTO messages (Message_id, Text_message, client_id) VALUES (2, 'Дайте больше кредит, я даю деньги бедным!', 4);

```

```
mysql> INSERT INTO messages (Message_id, Text_message, client_id) VALUES (4, 'Да я не муж Королевой!..', 3);
```

<b>docker pull</b> используется для получения образа Docker из Docker Registry	<pre>TutorialDemos:mysql /root\$ docker pull mysql latest TutorialDemos:mysql /root\$ docker run --name mysql -e MYSQL_ROOT_PASSWORD=root mysql mysql&gt; show databases; +-----+   Database   +-----+   information_schema     mysql          performance_schema     sys         +-----+ 4 rows in set (0.08 sec)</pre>
<b>docker scout</b> используется для анализа уязвимостей образов Docker и Docker Registry	<pre>TutorialDemos:mysql /root\$ docker scout quickview mysql:latest Vulnerability summary for mysql:latest</pre>
<b>docker images</b> используется для вывода списка всех образов Docker, которые находятся на локальном хосте	<pre>TutorialDemos:mysql /root\$ docker images REPOSITORY TAG IMAGE ID CREATED SIZE mysql latest e6f9f83c808 11 days ago 632kB</pre>
<b>docker run</b> используется для создания и запуска контейнера на основе нашего Docker образа.	<pre>TutorialDemos:mysql /root\$ docker run --name my_sql --rm --detach --env MYSQL_ROOT_PASSWORD=root mysql:latest mysql&gt; show databases;</pre>
<b>docker ps</b> используется для вывода списка активных контейнеров Docker на локальной машине.	<pre>TutorialDemos:mysql /root\$ docker ps CONTAINER ID        IMAGE               COMMAND                  Status    Up 5 seconds   0B memory     2MB/cpu       1m ago       my_sql_1</pre>
<b>docker exec -it My_SQL bash</b> используется для запуска интерактивной оболочки - bash, внутри контейнера Docker с именем "My_SQL". Команда <b>mysql -u root -p</b> используется для подключения к серверу MySQL с именем пользователя "root" и запросом пароля для входа.	<pre>TutorialDemos:mysql /root\$ docker exec -it my_sql bash bash-4.0# mysql -u root -p Enter password: Your MySQL connection id is 1 Server version: 4.3.36-mysq Community Server - GPL Copyright (c) 2000, 2002, Oracle and/or its affiliates. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.  Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  mysql&gt; show databases; +-----+   Database   +-----+   information_schema     mysql          performance_schema     sys         +-----+ 4 rows in set (0.08 sec)</pre>



Перейти в домашнюю директорию	<pre>Татьяна@TANYA MINGW64 ~ \$ cd</pre>
Создать новую папку	<pre>Татьяна@TANYA MINGW64 /c/test/project \$ mkdir final_bash  Татьяна@TANYA MINGW64 /c/test/project \$ ls final_bash/ final2_project/ general_project_2.0/</pre>
Перейти в новую папку	<pre>Татьяна@TANYA MINGW64 /c/test/project \$ cd final_bash/</pre>
Создать в этой новой папке новый файл	<pre>Татьяна@TANYA MINGW64 /c/test/project/final_bash \$ touch my_bash.txt  Татьяна@TANYA MINGW64 /c/test/project/final_bash \$ ls my_bash.txt</pre>
Перейти в текстовый редактор и написать случайный текст	<pre>Татьяна@TANYA MINGW64 /c/test/project/final_bash \$ vim my_bash.txt</pre> <p>Вот здесь, пожалуйста, укажите, какую именно строку вы хотите изменить. Если вы хотите изменить строку 1, то вы можете использовать команду:</p> <pre>1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100</pre>

Выйти и сохранить этот текст	<pre>~ ~ my_bash.txt[+] [unix] (11:39 29/03/2024) :wq</pre>
Посмотреть конкретное словосочетание в файле (  grep)	<pre>Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ grep "nordic" my_bash.txt школа nordic IT school - мы крутые!</pre>
Посмотреть конкретное словосочетание в файле и перенаправить поток в новый файл	<pre>Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ cat my_bash.txt   grep "nordic" &gt;&gt; my_bash2.txt  Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ ls folder1/ folder2/ my_bash.txt my_bash2.txt  Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ cat my_bash2.txt школа nordic IT school - мы крутые!</pre>
Посмотреть, что находится в вашей директории	<pre>Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ ls my_bash.txt my_bash2.txt</pre>
Как посмотреть, что находится в вашей директории, НО, включая скрытые файлы	<pre>Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ ls -a ./ ../ my_bash.txt my_bash2.txt</pre>

Посмотреть, где вы находитесь (в плане пути)	<pre>Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ pwd /c/test/project/Final_bash</pre>
Вы в домашней директории, надо: а. создать 2 папки б. перейти в первую любую папку с. создать там файл д. скопировать и перенести этот файл во вторую папку	<pre>Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ mkdir folder1 folder2  Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ ls folder1/ folder2/ my_bash.txt my_bash2.txt  Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ cd folder1  Татьяна@TANYA MINGW64 /c/test/project/Final_bash/folder1 \$ touch file.txt  Татьяна@TANYA MINGW64 /c/test/project/Final_bash/folder1 \$ cp file.txt /c/test/project/Final_bash/folder2  Татьяна@TANYA MINGW64 /c/test/project/Final_bash \$ cd folder2  Татьяна@TANYA MINGW64 /c/test/project/Final_bash/folder2 \$ ls file.txt</pre>

## BASH\_SCRIPT

### 1. contacts.sh

```
1 #!/bin/bash  
2  
3 echo "Введите имя:"  
4 read name  
5 echo "Введите номер телефона:"  
6 read phone  
7  
8 if [ ! -f contacts.csv ]; then  
9     echo "Имя,Номер телефона" > contacts.csv  
10 fi  
11  
12 echo "$name,$phone" >> contacts.csv  
13 echo "Данные успешно добавлены в таблицу."  
14  
15 sed 's/,/,/g' contacts.csv | column -t -s: | sed 's/,/,/g'find_dir_file.sh
```

### 2. find\_dir\_file.sh

```
1 #!/bin/bash  
2 # Поиск файлов и папок на диске  
3 files=$(find /c/users/Татьяна -type f | wc -l)  
4 directories=$(find /c/users/Татьяна -type d | wc -l)  
5 # Вывод результатов  
6 echo "Количество файлов: $files"  
7 echo "Количество папок: $directories"  
8 # Распределение по группам  
9 if [ $files -gt $directories ]; then  
10 echo "Больше файлов, чем папок"
```



```

11 elif [ $files -lt $directories ]; then
12     echo "Больше папок, чем файлов"
13 else
14     echo "Равное количество файлов и папок"
15 fi

```

### 3. search\_of\_file.sh

```

1  #!/bin/bash
2
3  echo "Введите имя файла, который вы ищете:"
4  read -p "Имя файла: " filename
5  directory="/c/test/Bash_script"
6  echo '===== ПОИСК... ====='
7  result=$(ls "$directory" | grep -i "$filename")
8  if [ -z "$result" ]; then
9      echo "Файл не найден"
10 else
11     file_path=$(find "$directory" -iname "$filename")
12     if [ -z "$file_path" ]; then
13         echo "Файл не найден"
14     else
15         echo "Файл найден: $file_path"
16     fi
17 fi
18

```



```

Petstore
- createUsersWithListInput
  POST https://petstore.swagger.io/v2/user/createWithList [200 OK, 370B, 768ms]
  ✓ Status code is 200
  ✓ Response time is less than 400ms
  ✓ Тест содержит Content-Type

- getUserByName
  GET https://petstore.swagger.io/v2/user/user1 [200 OK, 472B, 217ms]
  ✓ Status code is 200
  ✓ Response time is less than 400ms
  ✓ Тест содержит Content-Type

- Updated user
  PUT https://petstore.swagger.io/v2/user/user2 [200 OK, 387B, 135ms]
  ✓ Status code is 200
  ✓ Response time is less than 400ms
  ✓ Тест содержит Content-Type

- logs users into the system
  GET https://petstore.swagger.io/v2/user/login?username=user1&password=9999 [200 OK, 466B, 140ms]
  ✓ Status code is 200
  ✓ Response time is less than 400ms
  ✓ Тест содержит Content-Type

- create Users
  POST https://petstore.swagger.io/v2/user [200 OK, 369B, 381ms]
  ✓ Status code is 200
  ✓ Response time is less than 400ms
  ✓ Тест содержит Content-Type

```

	executed	failed
iterations	1	0
requests	5	0
test-scripts	5	0
prerequisite-scripts	0	0
assertions	15	1
total run duration: 2.1s		
total data received: 372B (approx)		
average response time: 328ms [min: 135ms, max: 768ms, s.d.: 237ms]		

```

# failure      detail
1. AssertionError Response time is less than 400ms
                  expected 768 to be below 400
                  at assertion:1 in test-script
                  inside "createUsersWithListInput"

```

### Petstore.postman\_collection.json

```

1  {
2      "info": {
3          "_postman_id": "0f90d27c-db10-40e4-b24f-91af670dd304",
4          "name": "Petstore",
5          "schema": "https://schema.getpostman.com/json/collection/v2.1.0/collection.json",
6          "_exporter_id": "31789902",
7          "_collection_link": "https://planetary-rocket-255884.postman.co/workspace/Nordic-school-9a1c0dd9-9514-4f5e-934c-efe9d91c5bb6/collection/31789902",
8      },
9      "item": [
10         {

```

```

11     "name": "createUsersWithListInput",
12     "event": [
13         {
14             "listen": "test",
15             "script": {
16                 "exec": [
17                     "pm.test(\"Status code is 200\", function () {\r",
18                         "    pm.response.to.have.status(200);\r",
19                     "});\r",
20                     "\r",
21                     "pm.test(\"Response time is less than 400ms\", function () {\r",
22                         "    pm.expect(pm.response.responseTime).to.be.below(400);\r",
23                     "});\r",
24                     "pm.test(\"Тест содержит Content-Type\", function () {\r",
25                         "    pm.response.to.have.header(\"Content-Type\");\r",
26                     "});\r",
27                     "\r",
28                     ""
29                 ],
30                 "type": "text/javascript",
31                 "packages": {}
32             }
33         }
34     ],
35     "request": {
36         "method": "POST",
37         "header": [],
38         "body": {
39             "mode": "raw",
40             "raw": "[\r\n    {\r\n        \"id\": 1,\r\n        \"username\": \"user1\",\r\n        \"firstName\": \"Tanya\",\r\n        \"lastName\": \"Vas\"
41             "options": {
42                 "raw": {
43                     "language": "json"
44                 }
45             }
46         },
47         "url": {
48             "raw": "{{URL}}/user/createWithList",
49             "host": [
50                 "{{URL}}"
51             ],
52             "path": [
53                 "user",
54                 "createWithList"
55             ]
56         }
57     },
58     "response": []
59 },
60 {
61     "name": "getUserByName",
62     "event": [
63         {
64             "listen": "test",
65             "script": {
66                 "exec": [
67                     "pm.test(\"Status code is 200\", function () {\r",
68                         "    pm.response.to.have.status(200);\r",
69                     "});\r",
70                     "\r",
71                     "pm.test(\"Response time is less than 400ms\", function () {\r",
72                         "    pm.expect(pm.response.responseTime).to.be.below(400);\r",
73                     "});\r",
74                     "\r",
75                     "\r",
76                     "pm.test(\"Тест содержит Content-Type\", function () {\r",
77                         "    pm.response.to.have.header(\"Content-Type\");\r",
78                     "});\r",
79                     ""
80                 ],
81                 "type": "text/javascript",
82                 "packages": {}
83             }
84         }
85     ],
86     "protocolProfileBehavior": {
87         "disableBodyPruning": true

```

```

88     },
89     "request": {
90         "method": "GET",
91         "header": [],
92         "body": {
93             "mode": "raw",
94             "raw": "",
95             "options": {
96                 "raw": {
97                     "language": "json"
98                 }
99             }
100         },
101         "url": {
102             "raw": "{{URL}}/user/user1",
103             "host": [
104                 "{{URL}}"
105             ],
106             "path": [
107                 "user",
108                 "user1"
109             ]
110         }
111     },
112     "response": []
113 },
114 {
115     "name": "Updated user",
116     "event": [
117         {
118             "listen": "test",
119             "script": {
120                 "exec": [
121                     "pm.test(\"Status code is 200\", function () {\r",
122                     "    pm.response.to.have.status(200);\r",
123                     "});\r",
124                     "\r",
125                     "pm.test(\"Response time is less than 400ms\", function () {\r",
126                     "    pm.expect(pm.response.responseTime).to.be.below(400);\r",
127                     "});\r",
128                     "\r",
129                     "pm.test(\"Тест содержит Content-Type\", function () {\r",
130                     "    pm.response.to.have.header(\"Content-Type\");\r",
131                     "});\r",
132                     ""
133                 ],
134                 "type": "text/javascript",
135                 "packages": {}
136             }
137         }
138     ],
139     "request": {
140         "method": "PUT",
141         "header": [],
142         "body": {
143             "mode": "raw",
144             "raw": "{\r\n  \"id\": 0,\r\n  \"username\": \"user2\",\r\n  \"firstName\": \"Klava\",\r\n  \"lastName\": \"Coca\",\r\n  \"email\": \"user2@example.com\"\r\n}",
145             "options": {
146                 "raw": {
147                     "language": "json"
148                 }
149             }
150         },
151         "url": {
152             "raw": "{{URL}}/user/user2",
153             "host": [
154                 "{{URL}}"
155             ],
156             "path": [
157                 "user",
158                 "user2"
159             ]
160         }
161     },
162     "response": []
163 },
164 {

```

```

165     "name": "logs users into the system",
166     "event": [
167         {
168             "listen": "test",
169             "script": {
170                 "exec": [
171                     "pm.test(\"Status code is 200\", function () {\r",
172                     "    pm.response.to.have.status(200);\r",
173                     "});\r",
174                     "\r",
175                     "pm.test(\"Response time is less than 400ms\", function () {\r",
176                     "    pm.expect(pm.response.responseTime).to.be.below(400);\r",
177                     "});\r",
178                     "\r",
179                     "\r",
180                     "pm.test(\"Тест содержит Content-Type\", function () {\r",
181                     "    pm.response.to.have.header(\"Content-Type\");\r",
182                     "});\r",
183                     ""
184                 ],
185                 "type": "text/javascript",
186                 "packages": {}
187             }
188         }
189     ],
190     "protocolProfileBehavior": {
191         "disableBodyPruning": true
192     },
193     "request": {
194         "method": "GET",
195         "header": [],
196         "body": {
197             "mode": "raw",
198             "raw": "[\r\n    {\r\n        \"id\": 1,\r\n        \"username\": \"Tanyaaaaa\",\r\n        \"firstName\": \"Tanya\",\r\n        \"lastName\": \"\r\n    }",
199             "options": {
200                 "raw": {
201                     "language": "json"
202                 }
203             }
204         },
205         "url": {
206             "raw": "{{URL}}/user/login?username=user1&password=9999",
207             "host": [
208                 "{{URL}}"
209             ],
210             "path": [
211                 "user",
212                 "login"
213             ],
214             "query": [
215                 {
216                     "key": "username",
217                     "value": "user1"
218                 },
219                 {
220                     "key": "password",
221                     "value": "9999"
222                 }
223             ]
224         }
225     },
226     "response": []
227 },
228 {
229     "name": "create Users",
230     "event": [
231         {
232             "listen": "test",
233             "script": {
234                 "exec": [
235                     "pm.test(\"Status code is 200\", function () {\r",
236                     "    pm.response.to.have.status(200);\r",
237                     "});\r",
238                     "\r",
239                     "pm.test(\"Response time is less than 400ms\", function () {\r",
240                     "    pm.expect(pm.response.responseTime).to.be.below(400);\r",
241                     "});\r",

```

```

242         "\r",
243         "\r",
244         "pm.test(\\"Тест содержит Content-Type\\", function () {\r",
245         "     pm.response.to.have.header(\\"Content-Type\\");\r",
246         "});\r",
247         "\r",
248         ""
249     ],
250     "type": "text/javascript",
251     "packages": {}
252 }
253 }
254 ],
255 "request": {
256     "method": "POST",
257     "header": [],
258     "body": {
259         "mode": "raw",
260         "raw": " {\r\n     \\"id\\": 2,\r\n     \\"username\\": \\"user\\",\r\n     \\"firstName\\": \\"Tanya\\",\r\n     \\"lastName\\": \\"Vas\\",\r\n
261         "options": {
262             "raw": {
263                 "language": "json"
264             }
265         }
266     },
267     "url": {
268         "raw": "{{URL}}/user",
269         "host": [
270             "{{URL}}"
271         ],
272         "path": [
273             "user"
274         ]
275     }
276 },
277 "response": []
278 }
279 ]
280 }

```

▼ petstore.postman\_environment.json

```

1  {
2      "id": "bc8edb12-71e6-45a1-b396-3ab4c90b170b",
3      "name": "petstore",
4      "values": [
5          {
6              "key": "URL",
7              "value": "https://petstore.swagger.io/v2",
8              "type": "default",
9              "enabled": true
10         }
11     ],
12     "_postman_variable_scope": "environment",
13     "_postman_exported_at": "2024-04-02T18:46:24.022Z",
14     "_postman_exported_using": "Postman/10.24.11"
15 }

```

▼ Petstore.postman\_test\_run.json

```

1  {
2      "id": "13a16140-6f1b-4fd1d-83d2-1ae1fa56108a",
3      "name": "Petstore",
4      "timestamp": "2024-04-03T13:31:56.654Z",
5      "collection_id": "31789902-0f90d27c-db10-40e4-b24f-91af670dd304",
6      "folder_id": 0,
7      "environment_id": "31789902-bc8edb12-71e6-45a1-b396-3ab4c90b170b",
8      "totalPass": 14,
9      "delay": 0,
10     "persist": true,
11     "status": "finished",
12     "startedAt": "2024-04-03T13:31:54.483Z",
13     "totalFail": 1,
14     "results": [
15         {

```

```

16     "id": "b097b49a-72aa-41dd-bd2d-32c4bbab9b24",
17     "name": "createUsersWithListInput",
18     "url": "https://petstore.swagger.io/v2/user/createWithList",
19     "time": 609,
20     "responseCode": {
21         "code": 200,
22         "name": "OK"
23     },
24     "tests": {
25         "Status code is 200": true,
26         "Response time is less than 400ms": false,
27         "Тест содержит Content-Type": true
28     },
29     "testPassFailCounts": {
30         "Status code is 200": {
31             "pass": 1,
32             "fail": 0
33         },
34         "Response time is less than 400ms": {
35             "pass": 0,
36             "fail": 1
37         },
38         "Тест содержит Content-Type": {
39             "pass": 1,
40             "fail": 0
41         }
42     },
43     "times": [
44         609
45     ],
46     "allTests": [
47         {
48             "Status code is 200": true,
49             "Response time is less than 400ms": false,
50             "Тест содержит Content-Type": true
51         }
52     ]
53 },
54 {
55     "id": "a0fdc73e-5248-4201-bb85-b5c7ef746404",
56     "name": "getUserByName",
57     "url": "https://petstore.swagger.io/v2/user/user1",
58     "time": 124,
59     "responseCode": {
60         "code": 200,
61         "name": "OK"
62     },
63     "tests": {
64         "Status code is 200": true,
65         "Response time is less than 400ms": true,
66         "Тест содержит Content-Type": true
67     },
68     "testPassFailCounts": {
69         "Status code is 200": {
70             "pass": 1,
71             "fail": 0
72         },
73         "Response time is less than 400ms": {
74             "pass": 1,
75             "fail": 0
76         },
77         "Тест содержит Content-Type": {
78             "pass": 1,
79             "fail": 0
80         }
81     },
82     "times": [
83         124
84     ],
85     "allTests": [
86         {
87             "Status code is 200": true,
88             "Response time is less than 400ms": true,
89             "Тест содержит Content-Type": true
90         }
91     ]
92 },

```

```

93 {
94     "id": "fa5e712c-aabc-467d-b5f3-12f6795820a6",
95     "name": "Updated user",
96     "url": "https://petstore.swagger.io/v2/user/user2",
97     "time": 126,
98     "responseCode": {
99         "code": 200,
100         "name": "OK"
101     },
102     "tests": {
103         "Status code is 200": true,
104         "Response time is less than 400ms": true,
105         "Тест содержит Content-Type": true
106     },
107     "testPassFailCounts": {
108         "Status code is 200": {
109             "pass": 1,
110             "fail": 0
111         },
112         "Response time is less than 400ms": {
113             "pass": 1,
114             "fail": 0
115         },
116         "Тест содержит Content-Type": {
117             "pass": 1,
118             "fail": 0
119         }
120     },
121     "times": [
122         126
123     ],
124     "allTests": [
125         {
126             "Status code is 200": true,
127             "Response time is less than 400ms": true,
128             "Тест содержит Content-Type": true
129         }
130     ],
131 },
132 {
133     "id": "d12f91cc-bfa4-4c05-8c07-ecd046c1f7aa",
134     "name": "logs users into the system",
135     "url": "https://petstore.swagger.io/v2/user/login?username=user1&password=9999",
136     "time": 128,
137     "responseCode": {
138         "code": 200,
139         "name": "OK"
140     },
141     "tests": {
142         "Status code is 200": true,
143         "Response time is less than 400ms": true,
144         "Тест содержит Content-Type": true
145     },
146     "testPassFailCounts": {
147         "Status code is 200": {
148             "pass": 1,
149             "fail": 0
150         },
151         "Response time is less than 400ms": {
152             "pass": 1,
153             "fail": 0
154         },
155         "Тест содержит Content-Type": {
156             "pass": 1,
157             "fail": 0
158         }
159     },
160     "times": [
161         128
162     ],
163     "allTests": [
164         {
165             "Status code is 200": true,
166             "Response time is less than 400ms": true,
167             "Тест содержит Content-Type": true
168         }
169     ]

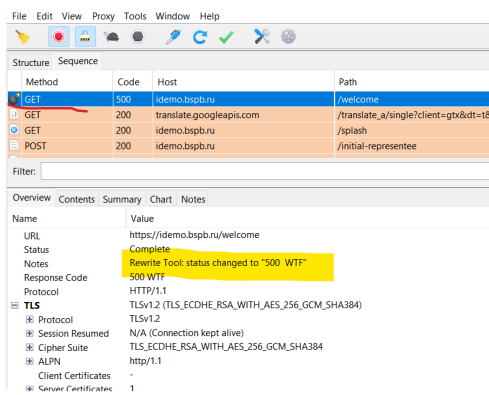
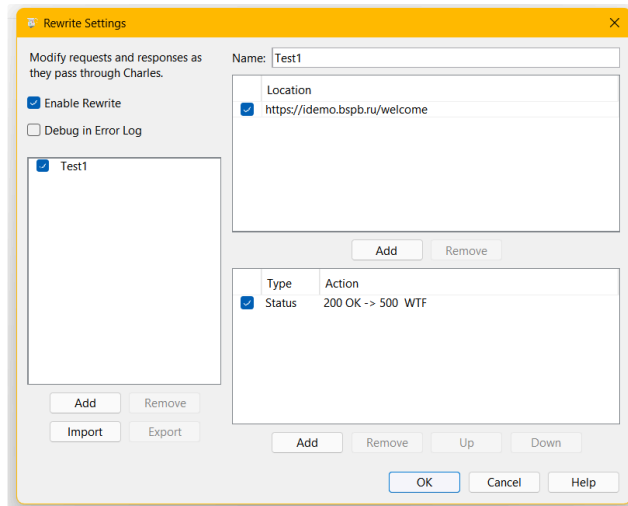
```

```

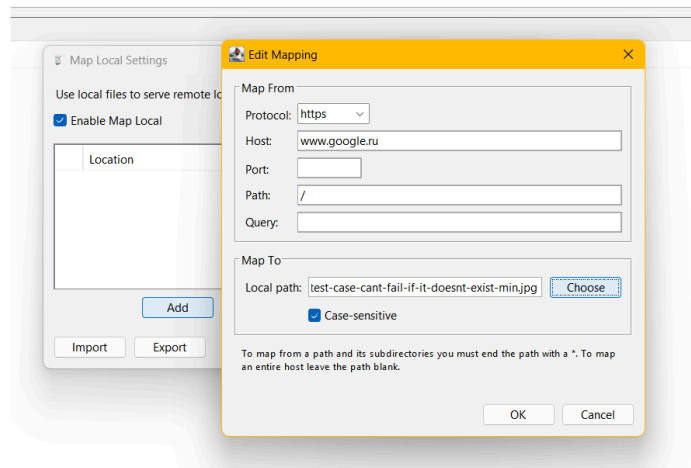
170     },
171     {
172         "id": "4a38672e-943e-459e-a521-e08b1d63faf8",
173         "name": "create Users",
174         "url": "https://petstore.swagger.io/v2/user",
175         "time": 373,
176         "responseCode": {
177             "code": 200,
178             "name": "OK"
179         },
180         "tests": {
181             "Status code is 200": true,
182             "Response time is less than 400ms": true,
183             "Тест содержит Content-Type": true
184         },
185         "testPassFailCounts": {
186             "Status code is 200": {
187                 "pass": 1,
188                 "fail": 0
189             },
190             "Response time is less than 400ms": {
191                 "pass": 1,
192                 "fail": 0
193             },
194             "Тест содержит Content-Type": {
195                 "pass": 1,
196                 "fail": 0
197             }
198         },
199         "times": [
200             373
201         ],
202         "allTests": [
203             {
204                 "Status code is 200": true,
205                 "Response time is less than 400ms": true,
206                 "Тест содержит Content-Type": true
207             }
208         ]
209     }
210 ],
211 "count": 1,
212 "totalTime": 1360,
213 "collection": {
214     "requests": [
215         {
216             "id": "b097b49a-72aa-41dd-bd2d-32c4bbab9b24",
217             "method": "POST"
218         },
219         {
220             "id": "a0fdc73e-5248-4201-bb85-b5c7ef746404",
221             "method": "GET"
222         },
223         {
224             "id": "fa5e712c-aabc-467d-b5f3-12f6795820a6",
225             "method": "PUT"
226         },
227         {
228             "id": "d12f91cc-bfa4-4c05-8c07-ecd046c1f7aa",
229             "method": "GET"
230         },
231         {
232             "id": "4a38672e-943e-459e-a521-e08b1d63faf8",
233             "method": "POST"
234         }
235     ]
236 }
237 }

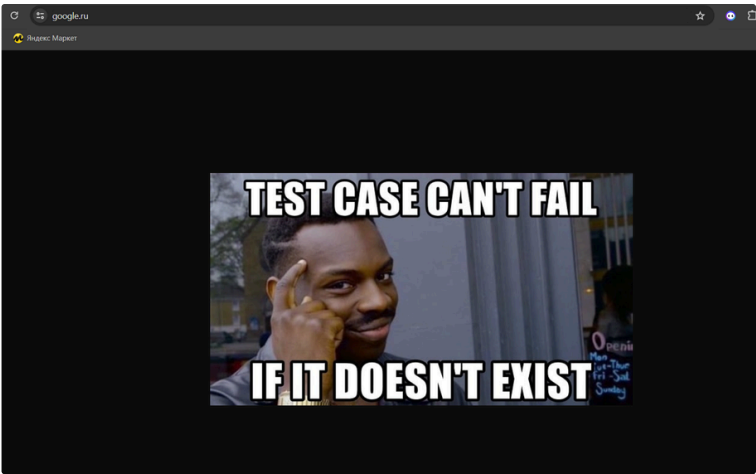
```



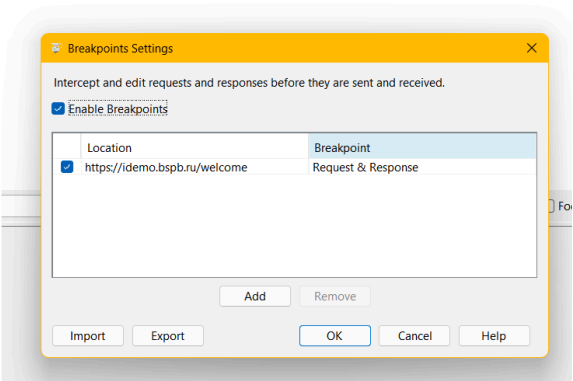


✓ Подмена тела ответа с помощью "Map local"





▼ Подмена header и статус-кода с помощью "Breakpoint"



## Карты

❗ Срок действия карты \*2329 заканчивается 04.06.2024 А МОЖЕТ И РАНЬШЕ : )

❗ Срок действия карты \*1123 заканчивается 05.05.2024

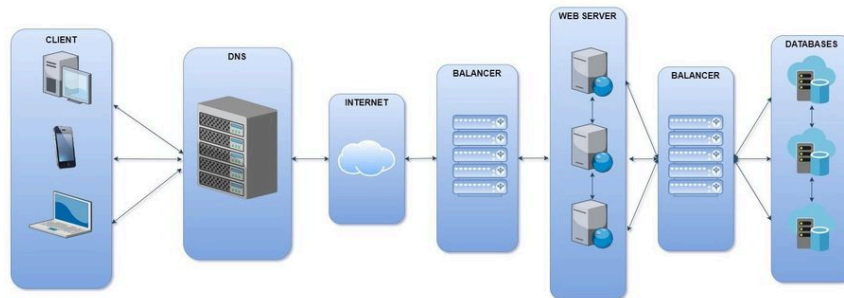
**Зарплатный \*2279** 350 711.00 ₽

- Travel \*6192
- Золотая \*2224
- Яркая \*9074
- Яркая \*9074 (заблокирована)
- Премимальная карта BLACK \*4524 (закрита)
- ДЕЕЕЕТСКАЯ \*4123

❗ Срок действия карты продлен до 07.2024. Для покупок в и дату как на карте.

[illegible]

## КЛИЕНТ-СЕРВЕРНАЯ АРХИТЕКТУРА



## CLIENT-SERVER ARCHITECTURE