

## Решение тестового задания на должность [“Специалист по сопровождению ПО”](#) в компанию “Алтек”

\*В описании к задаче не было указано, на каком языке программирования требуется выполнить задание, поэтому язык я выбрал сам. Задание было решено с учетом синтаксиса языка JavaScript

1. Напишите функцию/процедуру, принимающую в качестве параметра целое число и возвращающую название дня недели по этому номеру. А в случае если параметр некорректен (не попадает в диапазон [1-7]), возвращающую строку «Нет такого дня недели!».

Решение:

```
// объявление переменной (пользователь вводит значение во всплывающее окно)
let numberDay = Number(prompt('Введите число от 1 до 7', 'Ваше число'))
switch (numberDay) {
  case 1:
    alert('Понедельник');
    break; // проверка на то, что пользователь введет число 1

  case 2:
    alert('Вторник');
    break; // проверка на то, что пользователь введет число 2

  case 3:
    alert('Среда');
    break; // проверка на то, что пользователь введет число 3

  case 4:
    alert('Четверг');
    break; // проверка на то, что пользователь введет число 4

  case 5:
    alert('Пятница');
    break; // проверка на то, что пользователь введет число 5

  case 6:
    alert('Суббота');
    break; // проверка на то, что пользователь введет число 6

  case 7:
    alert('Воскресенье');
    break; // проверка на то, что пользователь введет число 7

  default:
    alert('Нет такого дня недели!') // если проверки не прошли
}
```

ne 33, Column 71

*Код программы на JavaScript*

Ввод данных на сайте 127.0.0.1

Введите число от 1 до 7

Ваше число

Продолжить

Отмена

Пример работы программы (1)

Ввод данных на сайте 127.0.0.1

Введите число от 1 до 7

2|

Продолжить

Отмена

Пример работы программы (2)

Уведомление от сайта 127.0.0.1

Вторник

☐ Больше не показывать сообщения от этого сайта

Закреть

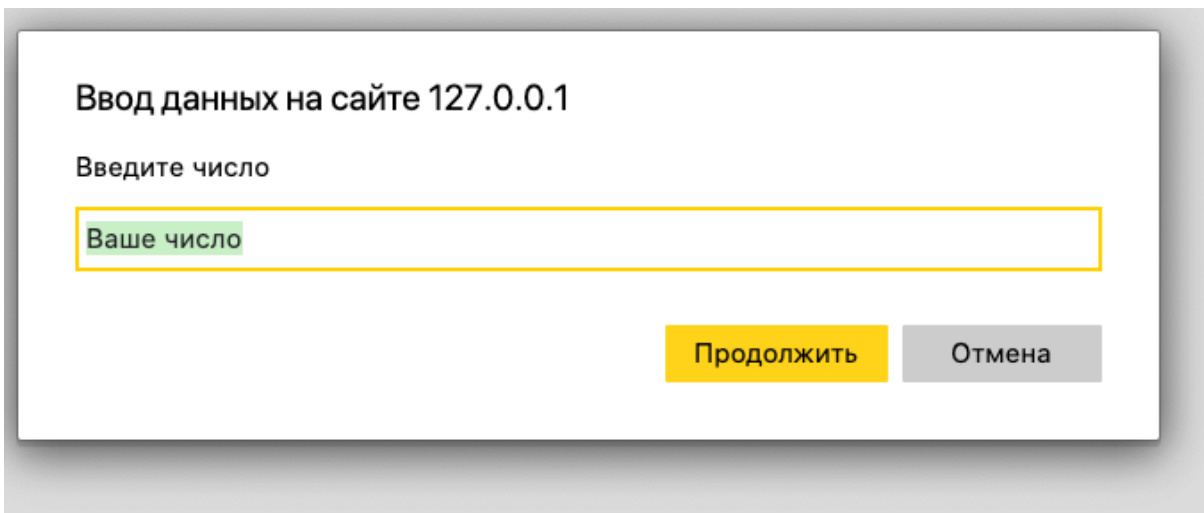
Пример работы программы (3)

2. Напишите функцию/процедуру, принимающую в качестве параметра целое число, и возвращающую строковое значение по следующему алгоритму:
- если параметр кратен 3, функция возвращает слово «Three»;
  - если параметр кратен 5, функция возвращает слово «Five»;
  - если параметр одновременно кратен и 3, и 5, то функция возвращает строку «Fifteen»;
  - если параметр не удовлетворяет ни одному из вышеописанных условий, то функция возвращает сам параметр.

```
// объявление переменной (пользователь вводит значение во всплывающее окно)
let number = Number(prompt('Введите число', 'Ваше число'));

if(number % 3 == 0 && number % 5 == 0) {
    alert("Fifteen"); // проверка на то, что число делится на 3 и на 5 одновременно
} else if (number % 5 == 0) {
    alert("Five"); // проверка на то, что число делится на 5
} else if( number % 3 == 0) {
    alert("Three"); // проверка на то, что число делится на 3
} else {
    alert(number); // если проверки не прошли, то возвращается число
}
```

*Код программы на JavaScript*



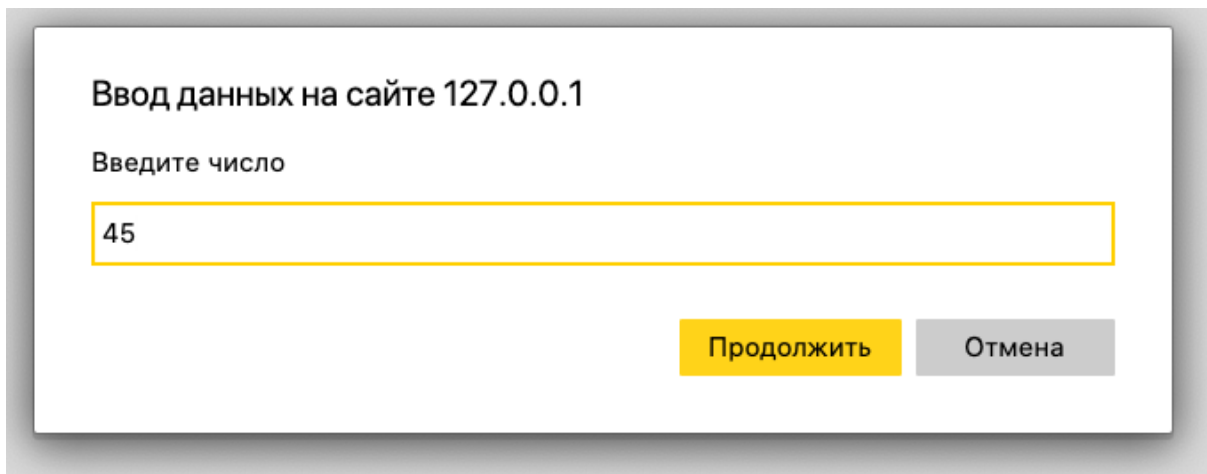
Ввод данных на сайте 127.0.0.1

Введите число

Ваше число

Продолжить Отмена

*Пример работы программы (1)*



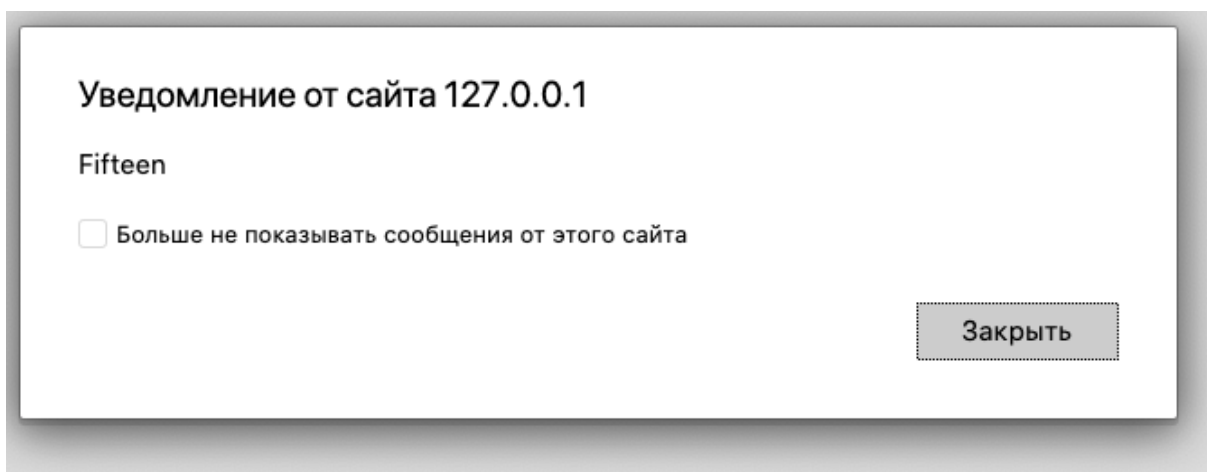
Ввод данных на сайте 127.0.0.1

Введите число

45

Продолжить Отмена

*Пример работы программы (2)*



Уведомление от сайта 127.0.0.1

Fifteen

☐ Больше не показывать сообщения от этого сайта

Закрыть

*Пример работы программы (3)*

### 3. SQL

\* Задания по SQL были выполнены на удаленном сервере при помощи синтаксиса СУБД MySQL

```

-- Создание таблиц и связи между ними
CREATE TABLE CUSTOMERS
(
    CUSTOMERID INT PRIMARY KEY AUTO_INCREMENT,
    CUSTOMER VARCHAR(30),
    CITY VARCHAR(30)
);

CREATE TABLE ORDERS
(
    ORDERID INT PRIMARY KEY AUTO_INCREMENT,
    ORDERNO VARCHAR(30),
    ORD_DATE DATE,
    PRICE DECIMAL(8, 2),
    CUSTOMERID INT,
    CONSTRAINT orders_custommerid_fk
    FOREIGN KEY(CUSTOMERID) REFERENCES CUSTOMERS(CUSTOMERID)
);

```

*Создание таблиц*

```

-- Заполнение таблиц данными из условия
INSERT INTO ORDERS (`ORDERID`, `ORDERNO`, `ORD_DATE`, `PRICE`, `CUSTOMERID`)
VALUES
(1, "ord1", "11.11.16", 2636.37, 1),
(2, "ord2", "11.11.16", 4352.32, 2),
(3, "ord3", "13.11.16", 6472.1, 6),
(4, "ord4", "14.11.16", 4342.34, 1),
(5, "ord5", "15.11.16", 7374.92, 4),
(6, "ord6", "17.11.16", 5647.34, 8),
(7, "ord7", "03.12.16", 8838.81, 8),
(8, "ord8", "05.12.16", 7234.12, 5),
(9, "ord9", "07.12.16", 6543.84, 7),
(10, "ord10", "07.12.16", 2983.94, 8);

INSERT INTO CUSTOMERS (`CUSTOMERID`, `CUSTOMER`, `CITY`)
VALUES
(1, "Иванов", "Москва"),
(2, "Петров", "Питер"),
(3, "Семенов", "Москва"),
(4, "Васильев", "Самара"),
(5, "Макаров", "Питер"),
(6, "Безруков", "Тамбов"),
(7, "Башаров", "Ростов"),
(8, "Куценко", "Питер"),
(9, "Михалков", "Челябинск");

```

*Заполнение таблиц данными из условия задачи*

```
-- Вывести номер и заказчика самого дорогого заказа
SELECT CUSTOMERS.CUSTOMER AS Заказчик, ORDERS.ORDERNO AS Номер_заказа
FROM
CUSTOMERS INNER JOIN ORDERS
ON CUSTOMERS.CUSTOMERID = ORDERS.CUSTOMERID
WHERE ORDERS.PRICE IN (
    SELECT MAX(ORDERS.PRICE)
    FROM ORDERS
);
```

*Вывести номер и заказчика самого дорогого заказа*

Заказчик	Номер_заказа
Куценко	ord7

*Результат*

```
-- Вывести список заказчиков, у которых нет покупок в алфавитном порядке
SELECT CUSTOMERS.CUSTOMERID, CUSTOMERS.CUSTOMER, CUSTOMERS.CITY
FROM
CUSTOMERS LEFT JOIN ORDERS
ON CUSTOMERS.CUSTOMERID = ORDERS.CUSTOMERID
WHERE ORDERS.PRICE IS null
ORDER BY CUSTOMER;
```

*Вывести список заказчиков, у которых нет покупок в алфавитном порядке*

CUSTOMERID	CUSTOMER ▲ 1	CITY
9	Михалков	Челябинск
3	Семенов	Москва

*Результат*

```
-- Вывести сумму заказов каждого города в порядке убывания суммы
SELECT CUSTOMERS.CITY AS Город, SUM(ORDERS.PRICE) AS Сумма_заказов
FROM CUSTOMERS INNER JOIN ORDERS
ON CUSTOMERS.CUSTOMERID = ORDERS.CUSTOMERID
GROUP BY CUSTOMERS.CITY
ORDER BY Сумма_заказов DESC;
```

*Вывести сумму заказов каждого города в порядке убывания суммы*

Город	Сумма_заказов
Питер	29056.53
Самара	7374.92
Москва	6978.71
Ростов	6543.84
Тамбов	6472.10

*Результат*