

1. Создайте веб-страницу (*order.html*), содержащую форму заказа почтовой доставки кофе, в которой должны быть следующие поля:
  - a. Две радиокнопки типа кофе: арабика и робуста. При выборе одной из этих кнопок – в выпадающем списке из следующего пункта – должны меняться сорта кофе.
  - b. Поле с выпадающим списком, содержащим известные Вам сорта кофе (минимум 3 варианта для каждого типа из пункта «а») (арабики и робусты));
  - c. Чекбокс «подарочная упаковка».
  - d. Текстовые поля:
    - индекс;
    - дата доставки;
    - имя и фамилия заказчика;
    - телефон заказчика;
    - адрес электронной почты заказчика.Реализуйте с помощью регулярных выражений следующие проверки вводимых данных в поля формы:
    - поле «Индекс» должно быть длиной только 7 цифр и введенные данные должны быть числом, причем первая цифра только четверкой или пятеркой;
    - формат даты должен быть ДД-ММ-ГГГГ или ДД-ММ-ГГ;
    - длина поля имени и фамилии заказчика должна быть от 5 до 35 символов;
    - в поле ввода телефона заказчика реализуйте следующую ситуацию: если в телефонном номере шесть цифр, то номер делится дефисом на три части из двух цифр (пример 23-45-67), если семь – сначала через дефис две группы по две цифры, затем три цифры (пример 12-34-567), в остальных случаях оставлять номер без изменений. Введенные данные должны быть числом. При этом, если пользователь введёт номер телефона в формате 23-45-67 или 12-34-234, ошибки возникать не должно;
    - адрес электронной почты должен проверяться на корректность.В случае неверного заполнения полей формы рядом с полем ввода должно показываться информационное сообщение с описанием ошибки пользователя (*span*).
  - e. Внизу страницы создайте кнопку, при нажатии на которую будет осуществляться проверка корректности введенных данных во все поля формы, в случае наличия ошибок выдающая сообщение об ошибке. В случае, если всё ок – в текстовое поле внизу должен выводиться состав заказа (выбранный тип кофе, сорт кофе, наличие подарочной упаковки) (*onblur, onchange, innerHTML, test*)
  - f. Вместо кофе может быть любой другой тип товаров, главное – сохранить те же поля (радио, выпадающий список, чекбокс и текстовые поля)
2. Создайте веб-страницу (*strings.html*), содержащую следующие элементы:
  - Текстовое поле с абзацем из случайного художественного произведения. я. Определите количество первых двух букв вашей фамилии (для фамилии «Иванов» – это «и» и «в», для «Долгов» – это «д» и «о» и т.п.) в этой фразе с учётом и без учёта регистра (т.е. должна быть кнопка, по нажатию на которую в *innerHTML* на странице выведется эта информация). *indexOf*
  - Кнопку, по нажатию на которую будет показано количество слов во фразе из этого текстового поля.
  - *<div>*, который содержит *<p>*, в котором располагается строка из нескольких предложений, к которым будут применяться преобразования из следующих двух пунктов.
    - Кнопку, по нажатию на которую будет выполнено преобразование строки: каждое нечётное слово было приведено к верхнему регистру, а каждое чётное – к нижнему.
    - Кнопку, по нажатию на которую будет выполнено удаление из строки всех букв, совпадающих с 4 буквой из вашей фамилии (для фамилии «Иванов» – это «н», для «Долгов» – это «г» и т.п.), с использованием метода *replace* объекта *String*.
  - Текстовое поле, которое будет проверять, является ли введенная строка именем файла с расширением *doc, xls* или *ppt*. (примеры корректных имён файлов: *FirmReport.doc, updated\_table.xls*, некорректных имён: *templPresent.ppt.tmp, img\_0344.png*). В случае, если имя файла в строке корректно, рядом с полем ввода должно отображаться сообщение зелёного цвета «Имя файла корректно». В случае, если неправильное – красного цвета «Неверное имя файла». (*onkeydown, onclick, onpaste*)



3. Используйте для оформления вашей страницы фреймворк Bootstrap

### Контрольные вопросы

1. Для чего используются регулярные выражения?
2. Какой символ в регулярных выражениях означает повторение предыдущего символа 0 и более раз?  
А какой – один и более раз?
3. Опишите принцип работы «запоминающих» скобок в регулярных выражениях.
4. Приведите примеры строк, которые подходят под регулярное выражение `^mission\s\w+\s\d{3}\s[a-g][d-r][f-z]\.`
5. Какие флаги отвечают за глобальный поиск в строке и игнорирование регистра?
6. Как можно объявить регулярное выражение в коде JavaScript?
7. Что обозначают спец символы \$ и ? в регулярных выражениях?
8. Опишите механизм работы методов `test` и `exec` в JavaScript.



## Контрольные вопросы

1. Для чего используются регулярные выражения?
2. Какой символ в регулярных выражениях означает повторение предыдущего символа 0 и более раз? А какой – один и более раз?
3. Опишите принцип работы «запоминающих» скобок в регулярных выражениях.
4. Приведите примеры строк, которые подходят под регулярное выражение `^mission\s\w+\s\d{3}\s[a-g][d-r][f-z]\.`
5. Какие флаги отвечают за глобальный поиск в строке и игнорирование регистра?
6. Как можно объявить регулярное выражение в коде JavaScript?
7. Что обозначают спец символы `$` и `?` в регулярных выражениях?
8. Опишите механизм работы методов `test` и `exec` в JavaScript.