



Программа расчета налогового вычета на лечение для физических лиц

Краснов Филипп Андреевич

**С этим приложением расчет
налогов будет производить в
разы легче)**


Это поможет многим людям!





01

Цель проекта

-  Создание программы расчета налогового вычета на лечение для физических лиц (далее НВ).

Основные требования

Требования к надежности

Программа должна быть стабильной и не выдавать ошибок при работе.

Требования к производительности

Для создания графического интерфейса можно использовать готовые библиотеки.

Требования к техническому обеспечению

Программа должна работать быстро и выдавать результат расчета за 5 секунд.

Основные требования к программе



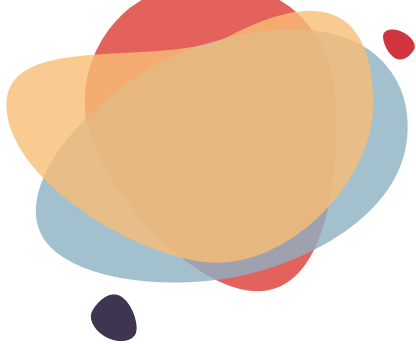
Учет видов налоговых вычетов, предусмотренных законодательством, таких как налоговый вычет на лечение.



Расчет налогового вычета в соответствии с действующим законодательством.



Возможность сохранения результатов расчета в файле для последующего использования.



Требования к интерфейсу

- 🍌 Интерфейс программы должен быть интуитивно понятным и удобным в использовании.
- 🍌 Интерфейс программы должен содержать элементы для ввода данных, требуемых для расчета налогового вычета.
- 🍌 Интерфейс должен содержать элементы для запуска процесса расчета «Рассчитать НВ».
- 🍌 Интерфейс должен содержать элементы для отражения результата расчёта.

Дополнительно

- Программа должна быть совместима с операционными системами Windows, MacOS и Linux.
- Программа должна иметь возможность выбора языка интерфейса (Русский, Английский).
- Добавить возможность изменения показателя «Лимит», при соблюдении условия:
 - если показатель не задан, то использовать значение 120 000;
 - если показатель = 0, то не использовать лимит в формуле расчета, то есть не сравнивать сумму лечения и лимит, всегда брать в расчет стоимость лечения;
 - если показатель > 0, то использовать значение нового лимита.

Используемые технологии

```
1  var language = "ru";
2  var maxDeduction = 120000;
3
4  document.getElementById("save").addEventListener("click", function() {
5      var expensesInput = document.getElementById("exp");
6      var expenses = parseFloat(expensesInput.value);
7      var previousDeductionsInput = document.getElementById("prev-deductions");
8      var previousDeductions = parseFloat(previousDeductionsInput.value) || 0;
9      var remainingDeduction = maxDeduction - previousDeductions;
```

Над проектом я работал +- часов 12, при работе над ним я использовал такие языки как html, css и js. Эти языки я использовал потому, что имею опыт работы с ними и они наиболее понятны мне. Использовал только базовые функции языка – чистый js, css и html.

Анализ процесса

`var language = "ru";` - объявляет переменную `language` и присваивает ей значение `"ru"`. Эта переменная указывает на текущий язык интерфейса.

`var maxDeduction = 120000;` - объявляет переменную `maxDeduction` и присваивает ей значение `120000`. Это максимальное значение налогового вычета.

`document.getElementById("calculate").addEventListener("click", function() {...})` - добавляет обработчик событий клика на кнопку "Calculate". Когда пользователь нажимает на эту кнопку, выполняется функция, которая рассчитывает налоговый вычет и выводит результат на страницу.

`document.getElementById("save").addEventListener("click", function() {...})` - добавляет обработчик событий клика на кнопку "Save". Когда пользователь нажимает на эту кнопку, выполняется функция, которая сохраняет результат расчета в текстовый файл.

`document.getElementById("expenses").addEventListener("input", function() {...})` - добавляет обработчик событий ввода для поля "Expenses". Когда пользователь вводит значение в это поле, выполняется функция, которая проверяет корректность введенных данных и выводит сообщение об ошибке при необходимости.

`document.getElementById("previous-deductions").addEventListener("input", function() {...})` - добавляет обработчик событий ввода для поля "Previous deductions". Когда пользователь вводит значение в это поле, выполняется функция, которая проверяет корректность введенных данных и выводит сообщение об ошибке при необходимости.

`document.getElementById("language-selector").addEventListener("click", function() {...})` - добавляет обработчик событий клика на кнопку "Language selector". Когда пользователь нажимает на эту кнопку, выполняется функция, которая меняет язык интерфейса.

`function getLocalizedString(text) {...}` - функция, которая возвращает локализованный текст в зависимости от текущего языка интерфейса.

`function updateInterfaceLanguage() {...}` - функция, которая обновляет язык интерфейса, изменяя текст на элементах страницы в соответствии с текущим языком.

Результаты работы над проектом

В результате получился многофункциональный web-сайт, который может вести учет видов налоговых вычетов, предусмотренных законодательством, таких как налоговый вычет на лечение, расчет налогового вычета в соответствии с действующим законодательством, а также возможность сохранения результатов расчета в файле для последующего использования.

The screenshot shows a web application titled "Расчет налогового вычета" (Calculation of tax deduction). It features a dark-themed interface with the following elements:

- Расходы:** A text input field containing the value "13000".
- Предыдущие вычеты:** A text input field containing the value "1000".
- Максимальная сумма лечения(в тысячах):** A horizontal slider control with a blue track and a white handle. The value "120" is displayed below the slider.
- Buttons:** Two green buttons labeled "Рассчитать" (Calculate) and "Сохранить" (Save) are positioned vertically on the left side.
- Result:** The calculated result is displayed in the center: "Налоговый вычет - 690 рублей 0 копеек." (Tax deduction - 690 rubles 0 kopecks).
- Language:** A small button labeled "English" is located in the bottom right corner.



КРАСНОВ ФИЛИПП



phillipskrasnov@yandex.ru

Удачи!

Надеюсь материал был полезен!?

