Частное учереждение образования

«Колледж бизнеса и права»

Отчет по лабораторной работе №22

Т.091004.401

Преподаватель ( Рогалевич А.В. )

Обучающийся ( Зеневич А.О. )

2024

**Задание №1-4:**

1. Сравнить два фрагмента кода.

Первый использует **finally** для выполнения кода после **try…catch**:

try {

    начать работу

    работать

} catch (e) {

    обработать ошибку

} finally {

    очистить рабочее пространство

}

Второй фрагмент просто ставит очистку после try..catch:

try {

    начать работу

    работать

} catch (e) {

    обработать ошибку

}

очистить рабочее пространство

Нужна очистка после работы, неважно возникли ошибки или нет.

Есть ли здесь преимущество в использовании **finally** или оба фрагмента кода одинаковы? Если такое преимущество есть, то дать пример, когда оно проявляется.

1. Написать функцию, которая будет извлекать корень из числа и при этом выбрасывать исключение, если корень извлекается из отрицательного числа.
2. Пусть JSON выдается следующим образом:

let json = `[

    {

        "name": "user1",

        "age": 25,

        "salary": 1000

    },

    {

        "name": "user2",

        "age": 26,

        "salary": 2000

    },

    {

        "name": "user3",

        "age": 27,

        "salary": 3000

    }

]`;

Проверить этот JSON на общую корректность при разборе, а после разбора проверить, что в результате получается массив, а не что-то другое. Если в результате получается не массив ‑ выбросить исключение.

1. Пользователь вводит три числа. Если одно из чисел меньше ноля, вернуть и обработать первое исключение, если одно из чисел равно нолю, вернуть и обработать второе исключение, если пользователь не ввёл одно или несколько чисел, вернуть и обработать третье исключение. По завершении работы в любом случае вывести сообщение о завершении работы алгоритма.

**Код задания:**

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>Лабораторная работа №22</title>

  </head>

  <body>

    <script>

      // Задание 1

      try {

        // начать работу

        console.log("Начать работу");

        // работать

        throw new Error("Ошибка");

      } catch (e) {

        // обработать ошибку

        console.error("Обработка ошибки:", e.message);

      } finally {

        // очистить рабочее пространство

        console.log("Очистить рабочее пространство");

      }

      try {

        // начать работу

        console.log("Начать работу");

        // работать

        throw new Error("Ошибка");

        // очистить рабочее пространство

      } catch (e) {

        // обработать ошибку

        console.error("Обработка ошибки:", e.message);

      } finally {

        console.log("Очистить рабочее пространство");

      }

      // Задание 2

      function calculateSquareRoot(number) {

        if (number < 0) {

          throw new Error("Нельзя извлечь корень из отрицательного числа");

        }

        return Math.sqrt(number);

      }

      try {

        const result = calculateSquareRoot(9);

        console.log("Корень из числа:", result);

      } catch (e) {

        console.error("Ошибка при вычислении корня:", e.message);

      }

      // Задание 3

      let json = `[

            {

                "name": "user1",

                "age": 25,

                "salary": 1000

            },

            {

                "name": "user2",

                "age": 26,

                "salary": 2000

            },

            {

                "name": "user3",

                "age": 27,

                "salary": 3000

            }

        ]`;

      try {

        const parsedJson = JSON.parse(json);

        if (!Array.isArray(parsedJson)) {

          throw new Error("JSON не преобразован в массив");

        }

        console.log("Разбор JSON:", parsedJson);

      } catch (e) {

        console.error("Ошибка при разборе JSON:", e.message);

      }

      // Задание 4

      try {

        const num1 = prompt("Введите первое число:");

        const num2 = prompt("Введите второе число:");

        const num3 = prompt("Введите третье число:");

        if (num1 === null || num2 === null || num3 === null) {

          throw new Error("Пользователь не ввел одно или несколько чисел");

        }

        const number1 = parseFloat(num1);

        const number2 = parseFloat(num2);

        const number3 = parseFloat(num3);

        if (isNaN(number1) || isNaN(number2) || isNaN(number3)) {

          throw new Error("Некорректный ввод. Введите числа.");

        }

        if (number1 < 0 || number2 < 0 || number3 < 0) {

          throw new Error("Одно из чисел меньше ноля");

        }

        if (number1 === 0 || number2 === 0 || number3 === 0) {

          throw new Error("Одно из чисел равно нолю");

        }

        console.log("Введенные числа:", number1, number2, number3);

      } catch (e) {

        console.error("Ошибка:", e.message);

      } finally {

        console.log("Завершение работы алгоритма");

      }

    </script>

  </body>

</html>

**Результат задания:**







