Podaj 2 przykłady wykorzystania instrukcji warunkowej SWITCH, a następnie stwórz schemat blokowey tej funcji

let day = 2;

let dayName;

switch (day) {

  case 1:

    dayName = "Poniedziałek";

    break;

  case 2:

    dayName = "Wtorek";

    break;

  case 3:

    dayName = "Środa";

    break;

  case 4:

    dayName = "Czwartek";

    break;

  case 5:

    dayName = "Piątek";

    break;

  default:

    dayName = "Nieznany dzień";

}

console.log(dayName);

let randomNumber = Math.floor(Math.random() \* 8);

let eightBall = '';

switch (randomNumber) {

  case 0:

    eightBall = "Tak";

    break;

  case 1:

    eightBall = "Nie";

    break;

  case 2:

    eightBall = "Tak";

    break;

  case 3:

    eightBall = "Nie";

    break;

  case 4:

    eightBall = "Tak";

    break;

  case 5:

    eightBall = "Nie";

    break;

  case 6:

    eightBall = "Tak";

    break;

  case 7:

    eightBall = "Nie";

    break;

}

console.log(eightBall);

Skonstruuj pętle for i while, które działają w ten sam sposób wyświetlając aktualną zmienną pomocniczą i. Pamiętaj, aby zadeklarować zmienną i jako równą 0 (i=0)

for (let i = 0; i < 5; i++) {

    console.log(`Wartość i w pętli for: ${i}`);

  }

let i = 0;

while (i < 5) {

  console.log(`Wartość i w pętli while: ${i}`);

  i++;

}

Uzasadnij róznice pomiędzy zwykla petla while oraz petla do while

Warunek początkowy:

* W pętli while warunek jest sprawdzany przed rozpoczęciem wykonania ciała pętli. Oznacza to, że jeśli warunek jest fałszywy na początku, to ciało pętli w ogóle nie zostanie wykonane.
* W pętli do...while ciało pętli jest wykonywane przynajmniej raz, zanim warunek jest sprawdzany. Oznacza to, że ciało pętli do...while zostanie wykonane co najmniej raz, niezależnie od wartości warunku.

Kolejność działania:

* W pętli while najpierw sprawdzany jest warunek, a następnie wykonywane jest ciało pętli (jeśli warunek jest spełniony).
* W pętli do...while najpierw wykonywane jest ciało pętli, a następnie sprawdzany jest warunek. Ciało pętli do...while będzie wykonane przynajmniej raz, niezależnie od warunku.

Przykład 1: Pętla while

START

 |

 V

Sprawdź warunek

 |

 V

Jeśli warunek jest fałszywy, zakończ pętlę

 |

 V

Jeśli warunek jest spełniony, wykonaj ciało pętli

 |

 V

Inkrementuj i

 |

 V

Powrót do sprawdzenia warunku

 |

 V

KONIEC

Przykład 1: Pętla do...while

do {

    i = i + 1;

    result = result + i;

  } while (i < 5);

START

 |

 V

Wykonaj ciało pętli

 |

 V

Inkrementuj i

 |

 V

Sprawdź warunek

 |

 V

Jeśli warunek jest fałszywy, zakończ pętlę

 |

 V

Powrót do wykonania ciała pętli

 |

 V

KONIEC

(5) Opisz algorytm, który wyświetli tylko liczby parzyste z zakresu od 0 do 10.

let parzyste = '';

for (let i = 1; i <= 10; i++) {

    if (i % 2 === 0) {

        parzyste += i + ' ';

    }

}

(6) Opisz algorytm, który wyświetli tylko liczby nieparzyste z zakresu od 0 do 10.

let nieparzyste = '';

for (let i = 1; i <= 10; i++) {

    if (i % 2 === 1) {

        nieparzyste += i + ' ';

    }

}

(7) Opisz algorytm, który wypisze co czwartą liczbę z zakresu z zakresu od 1 do 50 posługując się już nabytą wiedzą.

for(i=1; i<=50; i+=4){

    console.log(i)

}

8. Palindrom jak wyraz od tyłu jest taki sam czy nie

function palindrom(wyraz){

    wyraz1=wyraz

    wyraz =  wyraz.toLowerCase()

    wyraz1= wyraz1.split('').reverse().join('')

    return wyraz === wyraz1

}