Spis treści

[Zadanie 1. Zadania z funkcji. 1](#_Toc183687040)

[Zadanie 2. Zadanie z dekompozycji. 2](#_Toc183687041)

[Zadanie 3. Klasy/Obiekty 3](#_Toc183687042)

[Zadanie 4. Ćwiczenia formularze (html + js) 4](#_Toc183687043)

[Zadanie 5. Zadanie z modułów 6](#_Toc183687044)

[Zadanie 6. Zadanie ze zbiorem set. 6](#_Toc183687045)

[Zadanie 7. Instrukcja związana z Node.JS, TypeScript, Express.js, Pug ,Jest. 7](#_Toc183687046)

# Zadanie 1. Zadania z funkcji.

**1.1: Tworzenie funkcji.**

Napisz funkcję `calculateRectangleArea()`, która przyjmuje dwa parametry: `width` (szerokość) i `height` (wysokość) i zwraca pole prostokąta. Następnie wywołaj tę funkcję i wyświetl wynik.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

*[Odpowiedź kod + zrzut ekranu]*

**function calculateRectangleArea(width, height) {**

**return width \* height;**

**}**

**const szerokosc = 5;**

**const wysokosc = 3;**

**const pole = calculateRectangleArea(szerokosc, wysokosc);**

**console.log(`Pole prostokąta o szerokości ${szerokosc} i wysokości ${wysokosc} wynosi: ${pole}`);**

**1.2: Funkcje strzałkowe.**

Przepisz funkcję `calculateRectangleArea()` z poprzedniego zadania na funkcję strzałkową. Porównaj obie implementacje i upewnij się, że wynik jest nadal poprawny.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

*[Odpowiedź kod + zrzut ekranu]*

**function calculateRectangleArea(width, height) {**

**return width \* height;**

**}**

**const calculateRectangleAreaArrow = (width, height) => width \* height;**

**const szerokosc = 5;**

**const wysokosc = 3;**

**const pole1 = calculateRectangleArea(szerokosc, wysokosc);**

**console.log(`Wynik funkcji tradycyjnej: ${pole1}`);**

**const pole2 = calculateRectangleAreaArrow(szerokosc, wysokosc);**

**console.log(`Wynik funkcji strzałkowej: ${pole2}`);**

**1. 3: Dodawanie parametrów.**

Napisz funkcję `greet()`, która przyjmuje dwa parametry: `name` (imię) i `greeting` (powitanie), i zwraca spersonalizowane powitanie. Następnie wywołaj tę funkcję i wyświetl wynik.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

*[Odpowiedź kod + zrzut ekranu]*

function greet(name, greeting) {

    return `${greeting}, ${name}!`;

}

console.log(greet("Anna21", "Cześć"));

console.log(greet("Marek", "Dzień dobry"));

**1. 4: Parametry domyślne.**

Napisz funkcję `calculateCircleArea()`, która przyjmuje jeden parametr `radius` (promień) i zwraca pole koła. Dodaj domyślną wartość `radius` (np. 1) dla przypadku, gdy nie zostanie podany żaden argument. Wywołaj tę funkcję z różnymi wartościami promienia.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

*[Odpowiedź kod + zrzut ekranu]*

function calculateCircleArea(radius = 1) {

return Math.PI \* radius \* radius;

}

console.log(`Pole koła z domyślnym promieniem: ${calculateCircleArea()}`);

console.log(`Pole koła o promieniu 5: ${calculateCircleArea(5)}`);

console.log(`Pole koła o promieniu 3.5: ${calculateCircleArea(3.5)}`);

console.log(`Pole koła o promieniu 10: ${calculateCircleArea(10)}`);

**1.5: Funkcja z dowolną liczbą parametrów.**

Napisz funkcję `sumNumbers()`, która przyjmuje dowolną liczbę parametrów i zwraca ich sumę. Skorzystaj z operatora spread (`...`) w parametrach funkcji do obsługi wielu argumentów. Wywołaj funkcję z różnymi zestawami argumentów.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

*[Odpowiedź kod + zrzut ekranu]*

function sumNumbers(...numbers) {

    return numbers.reduce((sum, number) => sum + number, 0);

}

console.log(`Suma 2 liczb (2, 3): ${sumNumbers(2, 3)}`);

console.log(`Suma 4 liczb (1, 2, 3, 4): ${sumNumbers(1, 2, 3, 4)}`);

console.log(`Suma 6 liczb (10, 20, 30, 40, 50, 60): ${sumNumbers(10, 20, 30, 40, 50, 60)}`);

console.log(`Bez argumentów: ${sumNumbers()}`);

const liczby = [1, 2, 3, 4, 5];

console.log(`Suma liczb z tablicy: ${sumNumbers(...liczby)}`);

**1.6. Proszę opracować 5 funkcji według własnego pomysłu.**

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie*

# Zadanie 2. Zadanie z dekompozycji.

Za pomocą dekompozycji, proszę wyciągnąć następujące wartości z obiektu:

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

let ***car*** = {  
 brand: "Toyota",  
 model: "Camry",  
 year: 2022,  
 engine: {  
 type: "V6",  
 displacement: "3.5L"  
 },  
 features: {  
 safety: {  
 airbags: true,  
 abs: true  
 },  
 entertainment: {  
 radio: "Bluetooth",  
 touchscreen: true  
 }  
 }  
};

*[Odpowiedź kod + zrzut ekranu]*

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie

# Zadanie 3. Klasy/Obiekty

**3.1. Proszę za pomocą klas ES6 odzwierciedlić diagram uml:**

**Obraz zawierający tekst, diagram, ptak

Opis wygenerowany automatycznie**

*[Odpowiedź kod + zrzut ekranu]*

**3.2. Proszę opracować 5 klas według własnego pomysłu, wykorzystując mechanizm dziedziczenia. Należy wykonać według standardu ES6 (używając class).**

*Obraz zawierający zrzut ekranu, tekst, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie*

# Zadanie 4. Ćwiczenia formularze (html + js)

Wykonaj następujące ćwiczenia wykorzystując formularze html oraz JavaScript. Proszę dodać dodatkowo style css jako dekorację formularza. Proszę wykorzystać mechanizmy standardu ES6 (EcmaScript 6).

**4.1.** Napisz prosty konwerter godzin na minuty i odwrotnie. Pobierz liczbę godzin od użytkownika w formularzu, przelicz je na minuty i wypisz. Następnie pobierz od użytkownika informację o liczbie minut i zamień je na godziny i wypisz je w przeglądarce.

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający tekst, oprogramowanie, Strona internetowa, zrzut ekranu

Opis wygenerowany automatycznie*

**4.2.** Proszę napisać program rozwiązujący klasyczne równanie kwadratowe. (oblicza deltę i na jej podstawie wylicza miejsca zerowe - jeśli są). Do pierwiastka należy użyć Math.sqrt(delta) a do potęgowania

Math.pow().

Pobierz dane a, b i c z formularza.

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Ikona komputerowa

Opis wygenerowany automatycznie*

**4.3. Kalkulator opłat bankowych.**

Poproś użytkownika o podanie w formularzu salda konta bankowego i oblicz opłatę za prowadzenie konta, przyjmując, że opłata wynosi 10% salda.

accountBalance = 5000.0;

Obraz zawierający zrzut ekranu, tekst, oprogramowanie, System operacyjny

Opis wygenerowany automatycznie

*[Odpowiedź kod + zrzut ekranu]*

**4.4. Wybierz miesiąc.**

Poproś użytkownika o podanie numeru miesiąca (1 - styczeń, 12 - grudzień) i wyświetl pełną nazwę tego miesiąca.

month = 7;

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający zrzut ekranu, tekst, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie*

**4.5. Kalkulator BMI:**

Stwórz program, który obliczy BMI (Wskaźnik Masy Ciała) na podstawie masy (kg) i wzrostu (m) oraz poda informację o stanie zdrowia.

masa = 70.5;

wzrost = 1.75;

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie*

**4.6. Kalkulator podatku dochodowego:**

Napisz program, który na podstawie dochodu użytkownika obliczy należny podatek dochodowy.

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający zrzut ekranu, tekst, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie*

**4.7**. Napisz program który przyjmie dwie wartości od użytkownika oraz informację o działaniu arytmetycznym, które ma wykonać program (np.. +, -, \*, /).

Należy wykorzystać instrukcję switch w celu sprawdzenia znaku operacji i wykonania operacji arytmetycznej.

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający zrzut ekranu, tekst, oprogramowanie, Ikona komputerowa

Opis wygenerowany automatycznie*

**4.8.** Proszę stworzyć formularz html przyjmujący dane z informacjami o pracowniku. Odbierz dane z formularza po stronie javascript i odebrane dane zapisuj jako instancję klasy Pracownik. Docelowo dodawaj nowy obiekt do listy obiektów.

Wygeneruj dane pracowników z listy na stronę.

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Ikona komputerowa

Opis wygenerowany automatycznie*

# Zadanie 5. Zadanie z modułów

1.1. **Proszę stworzyć 2 moduły zawierające klasy, metody oraz zaimportować je w głównym pliku projektu. (1 plik główny projektu i 2 pliki - moduły).**

*[Odpowiedź kod + zrzut ekranu]*

*Obraz zawierający zrzut ekranu, tekst, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie*

# Zadanie 6. Zadanie ze zbiorem set.

Proszę napisać program symulujący grę w lotto.

6.1. przyjmujemy od użytkownika 6 różnych (unikalnych) liczb z przedziału od 1 do 49.

Set [1, 2, 3, 4, 5, 6]

\*Zweryfikować czy użytkownik wprowadza poprawne dane.

6.2. wylosowanie 6 różnych (unikalnych) liczb z przedziału od 1 do 49.

Set [1, 2, 3, 4, 5, 6]

6.3. Sprawdzenie ile liczb się pokrywa.

6.4. Wypisać informację dotyczącą trafionych liczb oraz informację dotyczącą nagrody.

*[Odpowiedź kod + zrzut ekranu]*

Obraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie

# Zadanie 7. Instrukcja związana z Node.JS, TypeScript, Express.js, Pug ,Jest.

Instrukcja, którą należy zrealizować znajduje się w pliku o nazwie:

***„Przyklad node\_TS+Express+Pug+Jest.pdf”.*** **Wynik instrukcji, czyli cały folder projektu należy wrzucić do archiwum i przesłać do oceny.**