

Wymagania edukacyjne

Tworzenie stron i aplikacji internetowych

Klasa II i klasa III – technik programista

Sposoby sprawdzania i oceniania osiągnięć edukacyjnych uczniów.

Ocenie podlega zarówno wiedza teoretyczna, jak i nabyte w trakcie nauki umiejętności.

Oceniane są:

- Ćwiczenia, zadania wykonywane na lekcji

Ocenie podlega: wykonanie wszystkich poleceń zgodnie z treścią, stopień samodzielności wykonywania zadania, pilność, końcowy efekt pracy (jakość pracy), umiejętność pracy w zespole.

- Aktywność podczas pracy na lekcji.

Ocenie podlega: aktywność ucznia w czasie zajęć, stopień zaangażowania podczas wykonywania zajęć, zainteresowanie tematem lekcji.

- Kartkówki, sprawdziany pisemne lub praktyczne.

Ocena z prac pisemnych zgodna ze Statutem Szkoły

- Zadania dodatkowe, prace projektowe

JavaScript

DOPUSZCZAJĄCY	DOSTATECZNY	DOBRY	BARDZO DOBRY	CELUJĄCY
<ul style="list-style-type: none"> • Zna podstawową składnię JavaScript (zmiennie, typy danych, operatory). • Potrafi napisać proste skrypty, ale ma trudności w zapewnieniu ich poprawności. • Umie tworzyć funkcje, ale nie zawsze potrafi je zastosować w kontekście aplikacji. • Zna podstawowe struktury kontrolne (instrukcje warunkowe, pętle), ale używa ich w sposób ograniczony. • Potrafi manipulować prostymi danymi (tablice, obiekty) w JavaScript, ale wymaga 	<p style="text-align: center;">tak jak na ocenę dopuszczającą oraz:</p> <ul style="list-style-type: none"> • Potrafi napisać proste skrypty w JavaScript, wykorzystując zmienne, typy danych i operatory. • Zna podstawowe struktury danych, takie jak tablice i obiekty, i potrafi je wykorzystywać. • Umie tworzyć i wywoływać funkcje w JavaScript. • Potrafi stosować instrukcje warunkowe (if, else) oraz pętle (for, while) w kontekście prostych zadań. • Potrafi przetwarzać dane z formularzy 	<p style="text-align: center;">tak jak na ocenę dostateczną oraz:</p> <ul style="list-style-type: none"> • Potrafi tworzyć bardziej złożone skrypty w JavaScript, wykorzystując tablice, obiekty oraz funkcje. • Umie obsługiwać bardziej zaawansowane zdarzenia DOM, takie jak formularze, kliknięcia czy zmiany wartości. • Potrafi wykorzystywać mechanizmy asynchroniczne w JavaScript, takie jak setTimeout, setInterval, • Umie manipulować DOM, dodawać, usuwać oraz modyfikować elementy strony. • Potrafi tworzyć 	<p style="text-align: center;">tak jak na ocenę dobrą oraz:</p> <ul style="list-style-type: none"> • Potrafi tworzyć zaawansowane aplikacje webowe z wykorzystaniem JavaScript, w tym interaktywne formularze, animacje czy aplikacje wykorzystujące dane z serwera. • Ma dobrą znajomość manipulacji DOM, może dynamicznie zmieniać elementy strony na podstawie działań użytkownika. • Zna podstawy programowania obiektowego w JavaScript (np. klasy, prototypy). • Potrafi korzystać z API przeglądarki, takich jak 	<p style="text-align: center;">tak jak w przypadku oceny bardzo dobrej oraz:</p> <ul style="list-style-type: none"> • Potrafi projektować i implementować pełnoprawne aplikacje webowe z wykorzystaniem JavaScript oraz nowoczesnych frameworków i bibliotek (np. React, Angular, Vue). • Zna wzorce projektowe i najlepsze praktyki programowania JavaScript, takie jak wzorce projektowe

JavaScript

<p>Umie wprowadzać dane do formularzy HTML, ale ma trudności z ich prawidłowym przetwarzaniem w JavaScript.</p>	<p>HTML oraz manipulować DOM (Document Object Model) w celu zmiany treści strony.</p> <ul style="list-style-type: none">• Umie obsługiwać podstawowe zdarzenia (np. kliknięcia, wprowadzenie danych) za pomocą JavaScript.	<p>interaktywne elementy na stronie, takie jak walidacja formularzy w czasie rzeczywistym.</p>	<p>(np. geolokalizacja). Zna i stosuje techniki asynchroniczne, takie jak AJAX, Fetch API, Promise i async/await. Potrafi pracować z danymi w formacie JSON oraz potrafi przesyłać dane między serwerem a klientem. Potrafi korzystać z narzędzi do debugowania</p>	
---	--	--	---	--

PHP				
DOPUSZCZAJĄCY	DOSTATECZNY	DOBRY	BARDZO DOBRY	CELUJĄCY
<ul style="list-style-type: none"> • przedstawia informacje związane z historią języka PHP • określa możliwości języka PHP • Zna podstawowe struktury PHP, ale ma trudności w ich używaniu. • Potrafi tworzyć proste skrypty, ale ma problemy z ich poprawnością. • Zna podstawowe funkcje PHP, jednak ich zastosowanie nie zawsze jest prawidłowe. • Rozumie pojęcia takie jak zmienne, typy danych, tablice, funkcje i struktury warunkowe. • Potrafi wprowadzić dane do formularzy, ale nie zawsze rozumie sposób ich przetwarzania. 	<p>tak jak na ocenę dopuszczającą oraz:</p> <ul style="list-style-type: none"> • Rozumie podstawową składnię języka PHP i potrafi pisać proste skrypty. • Zna podstawowe struktury danych (tablice, łańcuchy znaków) i potrafi je wykorzystywać. • Tworzy funkcje, potrafi wywoływać funkcje w PHP. • Umie korzystać z instrukcji warunkowych, pętli oraz funkcji wbudowanych w PHP. • Ma podstawową znajomość pracy z formularzami HTML i PHP. • Umie przechwytywać dane z formularzy i je przetwarzać. 	<p>tak jak na ocenę dostateczną oraz:</p> <ul style="list-style-type: none"> • Potrafi tworzyć skrypty, które przetwarzają dane z formularzy HTML i zapisują je do plików lub baz danych. • Ma dobrą znajomość funkcji wbudowanych w PHP oraz operacji na tablicach i łańcuchach znaków. • Zna i stosuje podstawowe zasady programowania obiektowego (klasy, obiekty, metody, dziedziczenie). • Potrafi korzystać z sesji i ciasteczek w PHP. • Zna podstawy pracy z bazami danych MySQL, w tym wykonywanie zapytań SELECT, INSERT, UPDATE, DELETE. • Potrafi tworzyć prostą aplikację webową z wykorzystaniem PHP i HTML. 	<p>tak jak na ocenę dobrą oraz:</p> <ul style="list-style-type: none"> • Potrafi tworzyć zaawansowane skrypty PHP, integrujące je z bazami danych (MySQL, PostgreSQL). • Zna zaawansowane techniki programowania obiektowego, w tym interfejsy, przestrzenie nazw, autoloading. • Potrafi tworzyć aplikacje z wykorzystaniem PHP oraz wzorców projektowych. • Potrafi stosować zasady programowania funkcjonalnego w PHP. • Potrafi efektywnie debugować aplikacje PHP oraz optymalizować kod. 	<p>tak jak w przypadku oceny bardzo dobrej oraz:</p> <ul style="list-style-type: none"> • Potrafi tworzyć zaawansowane aplikacje webowe oparte na PHP, z wykorzystaniem różnych narzędzi i • Potrafi projektować architekturę aplikacji PHP, uwzględniając wzorce projektowe, skalowalność i bezpieczeństwo. • Potrafi integrować aplikacje PHP z systemami zewnętrznymi • Ma doświadczenie w pracy z dużymi systemami bazodanowymi, optymalizując zapytania SQL oraz architekturę bazy danych.