



WYMAGANIA EDUKACYJNE

Zespół Szkół Technicznych w Pile

Przedmiot:	IM5.J4.Programowanie aplikacji internetowych po stronie klienta
Nauczyciel/Nauczyciele	Renata Nagiewicz
Rok szkolny:	2026/2027
Klasa:	3
OCENA	WYMAGANIA EDUKACYJNE
Dopuszczający (2)	- zna i przestrzega podstawowe zasady regulaminu pracowni; uruchamia wskazane środowisko pracy i narzędzia. - rozpoznaje HTML, CSS i JavaScript oraz określa ich podstawowe zastosowanie; - tworzy witryny internetowe o wyglądzie określonym w specyfikacji częściowo poprawnie, zna i stosuje znaczniki HTML oraz kaskadowe arkusze stylów, popełnia błędy - rozpoznaje składnię JavaScript i z pomocą nauczyciela tworzy proste instrukcje. - rozpoznaje zmienne, stałe i podstawowe typy danych; stosuje je częściowo poprawnie. - rozpoznaje podstawowe operatory; buduje proste wyrażenia. - tworzy prostą instrukcję warunkową - rozpoznaje zastosowanie pętli i tworzy prostą pętlę. - tworzy prostą funkcję i wywołuje ją w podanym miejscu. - rozpoznaje tablicę i wykonuje podstawowe operacje na niej z pomocą nauczyciela. - rozpoznaje obiekt i jego właściwości; korzysta z prostego przykładu - wykonuje podstawowe operacje na tekstach z pomocą nauczyciela. - rozpoznaje podstawowe funkcje i metody wbudowane; korzysta z nich według przykładu. - zapisuje prosty algorytm w JavaScript. - rozpoznaje dokument DOM i z pomocą nauczyciela wyszukuje wskazany element. - z pomocą nauczyciela zmienia treść lub atrybut wskazanego elementu. - rozpoznaje podstawowe zdarzenia - rozpoznaje podstawowe elementy formularza - rozpoznaje podstawowe błędy w skrypcie i z pomocą nauczyciela poprawia wskazane miejsce. - z pomocą nauczyciela wykorzystuje narzędzia deweloperskie do znalezienia prostego błędu. - rozpoznaje obiekt window i korzysta z prostych funkcji przeglądarki według instrukcji.

Dostateczny (3)

- z pomocą nauczyciela odczytuje datę lub uruchamia prosty timer.
 - tworzy prosty interaktywny element interfejsu.
 - łączy wskazany skrypt z dokumentem HTML i wykonuje prostą zmianę interfejsu.
 - rozpoznaje podstawowe zasady czytelnego interfejsu i korzysta z gotowych reguł.
 - wykonuje proste testy według instrukcji i zgłasza zauważone błędy.
 - z pomocą nauczyciela wyszukuje informacje potrzebne do wykonania zadania.
 - tworzy częściowo działającą aplikację zgodną z prostą specyfikacją.
-
- tworzy witryny internetowe o wyglądzie określonym w specyfikacji, zna i stosuje znaczniki HTML oraz kaskadowe arkusze stylów.
 - stosuje zasady bezpiecznej organizacji pracy; korzysta z podstawowych funkcji edytora kodu i przeglądarki.
 - wyjaśnia rolę HTML, CSS i JavaScript; tworzy prostą stronę zawierającą skrypt klienta.
 - stosuje podstawowe instrukcje i zapisuje proste fragmenty kodu JavaScript.
 - deklaruje zmienne i stałe oraz stosuje podstawowe typy danych w prostych zadaniach.
 - stosuje operatory arytmetyczne, porównania i logiczne w prostych skryptach.
 - stosuje if/else i podstawowe warianty warunków w prostych zadaniach.
 - stosuje podstawowe pętle do powtarzania operacji na danych.
 - tworzy funkcje wskazane przez nauczyciela, stosuje parametry i zwracanie wartości.
 - tworzy tablice i wykonuje podstawowe operacje na ich elementach.
 - tworzy proste obiekty i odczytuje lub modyfikuje ich właściwości.
 - stosuje podstawowe właściwości i metody napisów.
 - wywołuje wskazane funkcje i metody oraz interpretuje ich podstawowe działanie.
 - implementuje prosty algorytm zgodny z podaną specyfikacją.
 - wyszukuje podstawowe elementy dokumentu i odczytuje ich właściwości.
 - modyfikuje treść, atrybuty i klasy elementów za pomocą JavaScript.
 - obsługuje podstawowe zdarzenia, np. kliknięcie, zmianę wartości i przesłanie formularza.
 - tworzy prosty formularz i pobiera wartości jego pól za pomocą JavaScript.
 - stosuje podstawowe warunki do sprawdzania poprawności danych formularza.
 - korzysta z prostych mechanizmów obsługi błędów i komunikuje problem użytkownikowi.
 - korzysta z podstawowych funkcji konsoli przeglądarki i narzędzi deweloperskich.
 - wykorzystuje podstawowe właściwości i metody obiektu window.
 - stosuje podstawowe operacje na datach oraz proste timery.
 - tworzy podstawowe interakcje użytkownika zgodnie z podanym opisem.
 - łączy HTML, CSS i JavaScript w prostej aplikacji.

	• stosuje podstawowe zasady responsywności i poprawia czytelność interfejsu. • sprawdza podstawowe przypadki działania aplikacji i poprawia wskazane błędy. • wyszukuje potrzebne informacje w dokumentacji języka lub API. • korzysta z podstawowych zasobów Internetu i dobiera informacje do prostego zadania. • tworzy prostą aplikację zgodną ze specyfikacją, wykorzystując poznane mechanizmy. • bierze udział w realizacji projektu i prezentuje wykonane zadania.
Dobry (4)	• tworzy witryny internetowe o wyglądzie określonym w specyfikacji, zna i prawidłowo stosuje znaczniki HTML oraz kaskadowe arkusze stylów • samodzielnie dobiera podstawowe narzędzia do realizacji zadania i poprawnie organizuje projekt. • samodzielnie tworzy strukturę projektu i prawidłowo łączy dokument HTML, arkusz CSS i skrypt JavaScript. • samodzielnie wykorzystuje instrukcje warunkowe i iteracyjne do rozwiązania zadania. • dobiera wbudowane typy danych do problemu i poprawnie wykonuje operacje na danych. • samodzielnie buduje poprawne wyrażenia i wykorzystuje je w algorytmach. • samodzielnie buduje wielowariantowe warunki zgodne ze specyfikacją. • samodzielnie dobiera rodzaj pętli i poprawnie przetwarza kolejne elementy danych. • samodzielnie tworzy własne funkcje, dobiera parametry i wywołuje je zgodnie ze specyfikacją. • samodzielnie przetwarza tablice, wykorzystując iterowanie i metody tablicowe. • samodzielnie wykorzystuje obiekty do modelowania danych aplikacji. • samodzielnie przetwarza teksty, wyszukuje fragmenty i buduje komunikaty aplikacji. • samodzielnie dobiera i wywołuje funkcje oraz metody do realizacji zadania. • samodzielnie projektuje i implementuje algorytm dla określonego problemu. • samodzielnie modyfikuje elementy DOM, ich treść i atrybuty. • samodzielnie tworzy, usuwa i modyfikuje elementy HTML w reakcji na działanie użytkownika. • samodzielnie projektuje obsługę zdarzeń i reakcje interfejsu na działania użytkownika. • samodzielnie obsługuje formularz i przygotowuje dane do dalszego przetwarzania. • samodzielnie tworzy walidację pól i wyświetla użytkownikowi odpowiednie komunikaty. • samodzielnie lokalizuje typowe błędy i stosuje właściwe komunikaty. • samodzielnie analizuje działanie skryptu i usuwa błędy logiczne oraz składniowe. • samodzielnie programuje interfejs obsługi aplikacji internetowej. • samodzielnie integruje warstwę struktury, prezentacji i zachowania zgodnie ze specyfikacją. • samodzielnie tworzy interfejs dostosowujący się do różnych rozmiarów ekranu. • samodzielnie testuje aplikację zgodnie ze specyfikacją i usuwa wykryte nieprawidłowości. • samodzielnie korzysta z dokumentacji i dokumentuje własny kod w podstawowym zakresie.

Bardzo dobry (5)

- samodzielnie wykorzystuje dokumentację, przykłady i zasoby Internetu podczas pracy projektowej.
 - samodzielnie projektuje i bezbłędnie tworzy aplikację zgodną ze specyfikacją oraz prezentuje rozwiązanie.
 - aktywnie uczestniczy w pracy zespołowej, wykonuje przydzielone zadania i poprawnie przedstawia rozwiązanie.
-
- tworzy witryny internetowe o wyglądzie określonym w specyfikacji, zna i prawidłowo stosuje znaczniki HTML oraz kaskadowe arkusze stylów, tworzy zaawansowane layouty
 - sprawnie organizuje środowisko pracy, korzysta z dokumentacji i samodzielnie rozwiązuje typowe problemy konfiguracyjne.
 - projektuje strukturę aplikacji zgodnie ze specyfikacją i dobiera sposób organizacji plików do zadania.
 - sprawnie łączy różne konstrukcje języka w rozwiązania problemowe i dobiera je do specyfikacji.
 - sprawnie wykorzystuje typy danych w złożonych rozwiązaniach i analizuje skutki ich doboru.
 - dobiera operatory do rozwiązania problemu i poprawnie analizuje złożone warunki.
 - projektuje czytelne i poprawne rozwiązania warunkowe dla złożonych przypadków.
 - wykorzystuje iterowanie w zadaniach problemowych i kontroluje warunki zakończenia pętli.
 - projektuje podział programu na funkcje i wykorzystuje je do budowy czytelnych rozwiązań.
 - dobiera sposób przetwarzania kolekcji do zadania problemowego i tworzy czytelne rozwiązania.
 - projektuje struktury obiektowe adekwatne do specyfikacji i sprawnie nimi zarządza.
 - wykorzystuje operacje na tekstach w zadaniach problemowych, uwzględniając poprawność danych.
 - sprawnie wykorzystuje dokumentację funkcji i metod w zadaniach problemowych.
 - analizuje problem, dobiera właściwe konstrukcje i testuje poprawność algorytmu.
 - projektuje dynamiczne zmiany struktury dokumentu zgodnie ze specyfikacją.
 - analizuje przyczyny błędów i projektuje rozwiązania odporne na nieprawidłowe dane.
 - organizuje pracę podczas tworzenia projektu, wspiera innych i samodzielnie prezentuje działanie aplikacji.

Celujący (6)

- tworzy witryny internetowe o wyglądzie określonym w specyfikacji, zna i prawidłowo stosuje znaczniki HTML oraz kaskadowe arkusze stylów, tworzy zaawansowane layouty
- twórczo wykorzystuje konstrukcje języka, proponując alternatywne i efektywne rozwiązania.
- świadomie dobiera struktury i typy danych do większej aplikacji oraz tworzy rozwiązania wykraczające poza przykłady z zajęć.
- twórczo wykorzystuje wyrażenia i operatory w optymalnych rozwiązaniach problemowych.
- twórczo upraszcza i optymalizuje logikę warunkową w aplikacji.
- twórczo projektuje algorytmy iteracyjne i optymalizuje ich działanie.

- samodzielnie i twórczo projektuje zestaw funkcji wielokrotnego użycia oraz rozszerza rozwiązanie o własne mechanizmy.
- twórczo łączy funkcje i metody wbudowane, dobierając rozwiązania efektywne i wielokrotnego użycia.
- twórczo projektuje alternatywne rozwiązania i uzasadnia wybór najlepszego z nich.
- twórczo buduje rozbudowane mechanizmy manipulacji DOM i dba o ich czytelność oraz wydajność.
- twórczo tworzy rozbudowane, dynamiczne interfejsy i stosuje własne rozwiązania zwiększające użyteczność.
- systematycznie diagnozuje problemy, dobiera zaawansowane metody testowania i pomaga innym w debugowaniu.
- twórczo wykorzystuje możliwości środowiska przeglądarki do budowy własnych funkcji aplikacji.
- twórczo wykorzystuje mechanizmy czasu do budowy interaktywnych funkcji aplikacji.
- twórczo tworzy własny interfejs o wysokich walorach użytkowych i rozszerza jego funkcjonalność..
- twórczo wykorzystuje różnorodne zasoby i samodzielnie poszerza wiedzę poza materiał zajęć.
- tworzy własną aplikację posiadającą walory użytkowe, twórczo rozszerza jej funkcjonalność i może pełnić rolę lidera zespołu.

Podpis nauczyciela/nauczycieli