

# PROGRAMOWANIE JAVAScript

**32 GODZINY**

# PROGRAM

| LP       | Nazwa                                       | Plan. czas trw. (godz.) |
|----------|---|-------------------------|
| Wymagane |   |                         |
| 1        | Podstawowe elementy JavaScript              | 1                       |
| 2        | Praca ze zmiennymi i stałymi                | 1                       |
| 3        | Wyrażenia warunkowe                         | 2                       |
| 4        | Pętle                                       | 3                       |
| 5        | Organizacja kodu                            | 1                       |
| 6        | Korzystanie z funkcji                       | 2                       |
| 7        | Obiektowy Model Dokumentu                   | 2                       |
| 8        | Obiekty w JavaScript                        | 4                       |
| 9        | Interakcja z przeglądarką                   | 2                       |
| 10       | Wykorzystanie JavaScript w technologii AJAX | 3                       |
| 11       | Testowanie skryptów                         | 1                       |

łącznie 22 godz.

# PROGRAM

| Opcjonalne         |   |   |
|--------------------|---|---|
| 1                  | Tablice   | 2 |
| 2                  | Obsługa błędów  | 1 |
| 3                  | Wyrażenia regularne   | 1 |
| 4                  | JSON  | 1 |
| 5                  | Minimalizowanie rozmiaru kodu poprzez jego kompresję              | 1 |
| Jquery             |   |   |
| 1                  | Tworzenie animacji i elementów interaktywnych za pomocą JQuery    | 2 |
| 2                  | Definiowanie adaptacyjnych i responsywnych kontrolki użytkownika  | 2 |
| 3                  | AJAX za pomocą JQuery i WebAPI                                    | 2 |
| JavaScript w HTML5 |   |   |
| 1                  | Walidacja kontrolki za pomocą HTML5 i JavaScript                  | 2 |
| 2                  | File API  | 2 |
| 3                  | Drag & Drop   | 2 |
| 4                  | Canvas i JavaScript   | 2 |
| 5                  | Geolokalizacja  | 2 |
| 6                  | Wsparcie offline dla witryny                                      | 2 |
| 7                  | Przechowywanie danych lokalnie po stronie klienta w Local storage | 2 |
| 8                  | Wykonywanie czynności w tle za pomocą Web workers                 | 2 |
| 9                  | Multimedia  | 1 |

łącznie 29 godz.

# PROGRAM

| Wynikowy |  |   |
|----------|--|---|
| 1        | Podstawy HTML, CSS i Visual Studio       | 1 |
| 2        | Podstawowe elementy JavaScript           | 1 |
| 3        | Praca ze zmiennymi i stałymi             | 1 |
| 4        | Wyrażenia warunkowe                      | 2 |
| 5        | Pętle                                    | 3 |
| 6        | Funkcje                                  | 2 |
| 7        | Obiektowy Model Dokumentu                | 2 |
| 8        | Interakcja z przeglądarką                | 2 |
| 9        | Organizacja kodu                         | 1 |
| 10       | Tablice                                  | 2 |
| 11       | Wyrażenia regularne                      | 1 |
| 12       | Definiowanie i rozszerzanie obiektów     | 3 |
| 13       | JSON                                     | 2 |
| 14       | JQuery                                   | 2 |
| 15       | AJAX (JavaScript i JQuery)               | 4 |
| 16       | Validacja danych za pomocą JavaScript    | 1 |
| 17       | Obsługa błędów, debugowanie i testowanie | 2 |

łącznie 32 godz.

# ZADANIA

## Zadanie 1 – *Utworzenie strony*

Za pomocą Visual Studio dodaj nową witrynę HTML. Zdefiniuj jej strukturę, aby stosowała ona semantyczne tagi pozycjonowania HTML5 (np. article).

### **Materiały pomocnicze:**

- <http://www.w3schools.com/html/default.asp>
- [http://www.w3schools.com/html/html5\\_semantic\\_elements.asp](http://www.w3schools.com/html/html5_semantic_elements.asp)

# ZADANIA

## Zadanie 2 – *Zmienne, operatory*

Napisz aplikację, która pobiera dane z dwóch pól tekstowych. Następnie po kliknięciu na przycisk w paragrafie wypisywana jest ich średnia. Dodaje pogrubienie do wypisywanego tekstu.

### Materiały pomocnicze:

- [http://www.w3schools.com/html/html\\_form\\_elements.asp](http://www.w3schools.com/html/html_form_elements.asp)
- [http://www.w3schools.com/js/js\\_output.asp](http://www.w3schools.com/js/js_output.asp)

# ZADANIA

## Zadanie 3 – Stałe

Napisz aplikację, która po załadowaniu strony wypisuje „Hello World: *Aktualna Data* oraz PI aktualnie wynosi *wartość PI ze stałej* 😊”.

Dopisz kod dwukrotnie powiększający rozmiar tekstu z komunikatem.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_where.to.asp](http://www.w3schools.com/js/js_where.to.asp)
- [http://www.w3schools.com/js/js\\_intro.asp](http://www.w3schools.com/js/js_intro.asp)
- [http://www.w3schools.com/js/js\\_math.asp](http://www.w3schools.com/js/js_math.asp)
- <https://developer.mozilla.org/pl/docs/Web/JavaScript/Referencje/Polecenia/const>

# ZADANIA

## **Zadanie 4** - *Wyrażenia warunkowe, Komentarze*

Napisz aplikację, która dla podanej liczby sprawdza czy jest ona liczbą parzystą. Wprowadź stosowny komentarz opisujący działanie algorytmu.

### **Materiały pomocnicze:**

- [http://www.w3schools.com/js/js\\_statements.asp](http://www.w3schools.com/js/js_statements.asp)
- [http://www.w3schools.com/js/js\\_if\\_else.asp](http://www.w3schools.com/js/js_if_else.asp)
- [http://www.w3schools.com/js/js\\_comparisons.asp](http://www.w3schools.com/js/js_comparisons.asp)
- [http://www.w3schools.com/js/js\\_comments.asp](http://www.w3schools.com/js/js_comments.asp)



# ZADANIA

## Zadanie 5 - *pętle*

Napisz aplikację, która dla podanej liczby wylicza silnię.

**Materiały pomocnicze:**

- [http://www.w3schools.com/js/js\\_loop\\_for.asp](http://www.w3schools.com/js/js_loop_for.asp)

# ZADANIA

## Zadanie 6 – *pętle i instrukcje warunkowe*

Napisz aplikację, która dla podanej liczby sprawdza czy jest ona liczbą pierwszą.

### [Opcjonalne]

Dokonaj optymalizacji algorytmu.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_if\\_else.asp](http://www.w3schools.com/js/js_if_else.asp)
- [http://www.w3schools.com/js/js\\_loop\\_for.asp](http://www.w3schools.com/js/js_loop_for.asp)
- [http://www.w3schools.com/js/js\\_comparisons.asp](http://www.w3schools.com/js/js_comparisons.asp)
- [http://www.w3schools.com/js/js\\_break.asp](http://www.w3schools.com/js/js_break.asp)

# ZADANIA

## Zadanie 7 [opcjonalne] – *pętle i instrukcje warunkowe*

Napisz aplikację wyliczającą ile należy mieć dzisiaj pieniędzy na lokacie 3%, aby za 10 lat zostać milionerem. Weź pod uwagę inflację.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_if\\_else.asp](http://www.w3schools.com/js/js_if_else.asp)
- [http://www.w3schools.com/js/js\\_loop\\_for.asp](http://www.w3schools.com/js/js_loop_for.asp)
- [http://www.w3schools.com/js/js\\_comparisons.asp](http://www.w3schools.com/js/js_comparisons.asp)
- [http://www.w3schools.com/js/js\\_break.asp](http://www.w3schools.com/js/js_break.asp)

# ZADANIA

## Zadanie 8 [opcjonalne] – *pętle i instrukcje warunkowe*

Napisz aplikację, w której jako parametry podawane są: Posiadana kwota pieniędzy, Oprocentowanie lokaty, Inflacja, Procent podatku, oraz Liczba lat. Aplikacja jako rezultat ma podawać kwotę, która zostanie osiągnięta dla lokaty dla podanej liczby lat.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_if\\_else.asp](http://www.w3schools.com/js/js_if_else.asp)
- [http://www.w3schools.com/js/js\\_loop\\_for.asp](http://www.w3schools.com/js/js_loop_for.asp)
- [http://www.w3schools.com/js/js\\_comparisons.asp](http://www.w3schools.com/js/js_comparisons.asp)
- [http://www.w3schools.com/js/js\\_break.asp](http://www.w3schools.com/js/js_break.asp)

# ZADANIA

## Zadanie 9 - *funkcje*

Napisz aplikację, która dla podanych 2 liczb wypisuje jaki jest dla nich największy wspólny dzielnik. Algorytm napisz jako osobną metodę.

## Zadanie 10 – *funkcje, organizacja kodu*

Stosując algorytm Euklidesa zoptymalizuj działanie aplikacji szukającej największy wspólny dzielnik dla 2 liczb. Nowy algorytm napisz jako osobną metodę. Dla metod zastosuj przeciążenie. Zdefiniuj kod jako bibliotekę.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_functions.asp](http://www.w3schools.com/js/js_functions.asp)
- [http://www.w3schools.com/js/js\\_loop\\_while.asp](http://www.w3schools.com/js/js_loop_while.asp)

# ZADANIA

## Zadanie 11 [opcjonalne] - funkcje

Napisz aplikację, w której jako parametry podawane są:

- Posiadana kwota pieniędzy
- Oprocentowanie lokaty
- Inflacja
- Procent podatku
- Kwota, która pozwoli utrzymać się przez rok

Aplikacja ma wypisywać ile pozostało jeszcze kwoty w danym roku i zatrzymać się (czyt. przestać wypisywać) w momencie, w którym skończą się środki (czyt. wartość  $\leq 0$ ).

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_functions.asp](http://www.w3schools.com/js/js_functions.asp)
- [http://www.w3schools.com/js/js\\_loop\\_while.asp](http://www.w3schools.com/js/js_loop_while.asp)

# ZADANIA

## Zadanie 12 – *Przetwarzanie tekstu*

Napisz aplikację, w której użytkownik w pole tekstowe wprowadza tekst *nazwa produktu: cena waluta*, np. laptop: 2000 zł. Po kliknięciu na przycisk wypisywane są pod sobą: nazwa produktu, cena i waluta.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_strings.asp](http://www.w3schools.com/js/js_strings.asp)
- [http://www.w3schools.com/js/js\\_string\\_methods.asp](http://www.w3schools.com/js/js_string_methods.asp)

# ZADANIA

## Zadanie 13 – *Operowanie na liczbach*

Dla aplikacji „liczba parzysta” lub innej z podawaniem w pole tekstowe wartości liczbowej dodaj sprawdzanie czy użytkownik podał liczbę.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_numbers.asp](http://www.w3schools.com/js/js_numbers.asp)



# ZADANIA

## Zadanie 14 – *Operowanie na datach*

Napisz aplikację, w której użytkownik w kontrolce kalendarza podaje datę urodzenia. Aplikacja następnie wyświetla czy jest osobą pełnoletnią.

### Materiały pomocnicze:

- [http://www.w3schools.com/html/html\\_form\\_input\\_types.asp](http://www.w3schools.com/html/html_form_input_types.asp)
- [http://www.w3schools.com/js/js\\_date\\_methods.asp](http://www.w3schools.com/js/js_date_methods.asp)

# ZADANIA

## Zadanie 15 - DOM

Utwórz stronę z polem tekstowym dla wprowadzenia wzrostu osoby. Dodaj obsługę stosownego zdarzenia weryfikującego, że wprowadzane są wyłącznie liczby.

W przypadku nie wprowadzania przez użytkownika liczby obok pola niech wyświetla się na czerwono komunikat „Proszę wprowadzać tylko liczby”. Po wyświetleniu komunikatu, jeśli użytkownik wprowadzi liczbę komunikat ma zniknąć.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_htmlDOM\\_document.asp](http://www.w3schools.com/js/js_htmlDOM_document.asp)
- [http://www.w3schools.com/js/js\\_htmlDOM\\_events.asp](http://www.w3schools.com/js/js_htmlDOM_events.asp)
- [http://www.w3schools.com/js/js\\_htmlDOM\\_eventlistener.asp](http://www.w3schools.com/js/js_htmlDOM_eventlistener.asp)
- [http://www.w3schools.com/js/js\\_htmlDOM\\_css.asp](http://www.w3schools.com/js/js_htmlDOM_css.asp)

# ZADANIA

## Zadanie 16 – BOM (okno, zdarzenia)

Napisz kod pytający się użytkownika czy chce zamknąć okienko (zakładkę), jeśli ma ono mniej niż 1000px.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_window\\_screen.asp](http://www.w3schools.com/js/js_window_screen.asp)
- [http://www.w3schools.com/tags/ref\\_eventattributes.asp](http://www.w3schools.com/tags/ref_eventattributes.asp)
- [http://www.w3schools.com/js/js\\_popup.asp](http://www.w3schools.com/js/js_popup.asp)

# ZADANIA

## **Zadanie 17** – *BOM (dane użytkownika)*

Napisz kod w trakcie ładowania strony sprawdzający i wypisujący czy przeglądarka użytkownika obsługuje JS i Ciasteczka.

### **Materiały pomocnicze:**

- [http://www.w3schools.com/js/js\\_window\\_navigator.asp](http://www.w3schools.com/js/js_window_navigator.asp)

# ZADANIA

## Zadanie 18 – BOM (ciasteczka)

Napisz stronę z kodem zapisujący do ciasteczka twój ulubiony kolor. Wartość koloru z ciasteczka ma być wczytywana przy każdym załadowaniu strony i na nią ma być ustawiane tło strony.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_cookies.asp](http://www.w3schools.com/js/js_cookies.asp)

# ZADANIA

## Zadanie 19 – BOM (zadania w tle)

Dodaj do strony zegar.

**Materiały pomocnicze:**

- [http://www.w3schools.com/js/js\\_timing.asp](http://www.w3schools.com/js/js_timing.asp)

# ZADANIA

## Zadanie 20 – *Dobre praktyki pisania kodu*

Sprawdź dla aplikacji „Największego wspólnego dzielnika”, czy zastosowałeś konwencje i dobre praktyki pisania kodu.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_conventions.asp](http://www.w3schools.com/js/js_conventions.asp)
- [http://www.w3schools.com/js/js\\_best\\_practices.asp](http://www.w3schools.com/js/js_best_practices.asp)

# ZADANIA

## **Zadanie 21 – *Tablice***

Napisz aplikację w której losowane są liczby z danego przedziału (analogicznie do lotto) i przechowujący je w tablicy.

## **Zadanie 22 [opcjonalne] - *Tablice***

Dla aplikacji dopisz kod sortujący wylosowane liczby i następnie je wypisujący.

## **Zadanie 23 [opcjonalne] - *Tablice***

Dla aplikacji dopisz kod pozwalający sprawdzić czy dana liczba została wylosowana.

## **Zadanie 24 [opcjonalne] - *Tablice***

Dla aplikacji dopisz kod wyliczający średnią, medianę oraz dominantę dla wylosowanych liczb.

### **Materiały pomocnicze:**

- [http://www.w3schools.com/js/js\\_math.asp](http://www.w3schools.com/js/js_math.asp)
- [http://www.w3schools.com/js/js\\_arrays.asp](http://www.w3schools.com/js/js_arrays.asp)
- [http://www.w3schools.com/js/js\\_array\\_methods.asp](http://www.w3schools.com/js/js_array_methods.asp)



# ZADANIA

## Zadanie 25 - *Tablice*

Dodaj stronę o nazwie wylicz koszt zamówienia, na której za pomocą JavaScriptu i formularza wyliczany będzie koszt produktu (cena \* liczba sztuk), gdzie kwoty produktów przechowywane są w tablicy.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_arrays.asp](http://www.w3schools.com/js/js_arrays.asp)
- [http://www.w3schools.com/js/js\\_array\\_methods.asp](http://www.w3schools.com/js/js_array_methods.asp)

# ZADANIA

## Zadanie 26 – *JSON, tablice, zdarzenia przeglądarka*

Napisz aplikację dla testowania wiedzy. Pytania mają być przechowywane w tablicy jako obiekty JSON, interfejs testu generowany na jej podstawie dynamicznie. Test ma być wyświetlany zaraz po załadowaniu strony.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_html\\_dom\\_events.asp](http://www.w3schools.com/js/js_html_dom_events.asp)
- [http://www.w3schools.com/jquery/jquery\\_dom\\_add.asp](http://www.w3schools.com/jquery/jquery_dom_add.asp)
- [http://www.w3schools.com/json/json\\_syntax.asp](http://www.w3schools.com/json/json_syntax.asp)
- [http://www.w3schools.com/json/json\\_eval.asp](http://www.w3schools.com/json/json_eval.asp)
- [http://www.w3schools.com/js/js\\_arrays.asp](http://www.w3schools.com/js/js_arrays.asp)

# ZADANIA

## **Zadanie 27** – *Wyrażenia regularne*

Napisz formularz rejestracyjny użytkownika. Za pomocą wyrażeń regularnych dokonaj sprawdzenia formatu e-mail oraz siły hasła (opcjonalne; min. 7 znaków, 1 z dużej, 1 cyfra, 1 znak specjalny).

### **Materiały pomocnicze:**

- [http://www.w3schools.com/jsref/jsref\\_obj\\_regexp.asp](http://www.w3schools.com/jsref/jsref_obj_regexp.asp)

# ZADANIA

## Zadanie 28 – *Definiowanie obiektów*

Zdefiniuj klasę koszyka sklepu internetowego, zawierająca:

- Dane kupującego (min.: imię, nazwisko, adres)
- Listę pozycji w koszyku
- Metodę dodania i usunięcia pozycji z koszyka

Za pomocą kodu utwórz obiekt koszyka i dodaj kilka pozycji do niego.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_object\\_definition.asp](http://www.w3schools.com/js/js_object_definition.asp)
- [http://www.w3schools.com/js/js\\_properties.asp](http://www.w3schools.com/js/js_properties.asp)
- [http://www.w3schools.com/js/js\\_object\\_methods.asp](http://www.w3schools.com/js/js_object_methods.asp)
- [http://www.w3schools.com/js/js\\_object\\_prototypes.asp](http://www.w3schools.com/js/js_object_prototypes.asp)

# ZADANIA

## **Zadanie 29** – *Oprogramowanie obiektów*

Zdefiniuj listę produktów (3-5) jako tabelkę. Dla każdej z pozycji dodaj kod ze zdarzeniem pozwalającym dodać produkt do koszyka (stosownego obiektu).

Dodaj przycisk wypisujący zawartość koszyka.

### **Materiały pomocnicze:**

- [http://www.w3schools.com/js/js\\_htmlDOM\\_elements.asp](http://www.w3schools.com/js/js_htmlDOM_elements.asp)
- [http://www.w3schools.com/js/js\\_htmlDOM\\_document.asp](http://www.w3schools.com/js/js_htmlDOM_document.asp)
- [http://www.w3schools.com/js/js\\_htmlDOM\\_events.asp](http://www.w3schools.com/js/js_htmlDOM_events.asp)
- [http://www.w3schools.com/js/js\\_htmlDOM\\_eventlistener.asp](http://www.w3schools.com/js/js_htmlDOM_eventlistener.asp)

# ZADANIA

## **Zadanie 30** – *Rozszerzanie obiektów*

Zmień, aby w klasie koszyka jako pole wewnętrzne przechowywana była data dodania ostatniej pozycji do koszyka. Dodaj do prototypu koszyka metodę pobrania daty ostatniej dodanej pozycji

### **Materiały pomocnicze:**

- [http://www.w3schools.com/js/js\\_object\\_prototypes.asp](http://www.w3schools.com/js/js_object_prototypes.asp)

# ZADANIA

## Zadanie 31 - *jQuery*

Za pomocą jQuery napisz kod powodujący powiększanie się tekstu w przypadku najechania na poszczególne wypunktowania z listy zainteresowań.

### Materiały pomocnicze:

- [http://www.w3schools.com/jquery/jquery\\_selectors.asp](http://www.w3schools.com/jquery/jquery_selectors.asp)
- [http://www.w3schools.com/jquery/jquery\\_events.asp](http://www.w3schools.com/jquery/jquery_events.asp)
- [http://www.w3schools.com/jquery/jquery\\_css.asp](http://www.w3schools.com/jquery/jquery_css.asp)

# ZADANIA

## Zadanie 32 - *jQuery*

Przygotuj listę zainteresowań wraz z ich opisem. Za pomocą jQuery dodaj funkcjonalność pozwalającą rozwijać i zwijać poszczególne opisy – analogicznie jak dla kontrolki Accordion.

Zmień sposób ukrywania i pojawiania się elementów na stosujący animację (fade lub slide).

### Materiały pomocnicze:

- [http://www.w3schools.com/jquery/jquery\\_hide\\_show.asp](http://www.w3schools.com/jquery/jquery_hide_show.asp)
- [http://www.w3schools.com/jquery/jquery\\_fade.asp](http://www.w3schools.com/jquery/jquery_fade.asp)
- [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp)
- [http://www.w3schools.com/jquery/jquery\\_fade.asp](http://www.w3schools.com/jquery/jquery_fade.asp)
- [http://www.w3schools.com/jquery/jquery\\_slide.asp](http://www.w3schools.com/jquery/jquery_slide.asp)



# ZADANIA

## Zadanie 33 – *jQuery (animation)*

Za pomocą jQuery zdefiniuj animowane logo.

### Materiały pomocnicze:

- [http://www.w3schools.com/jquery/jquery\\_animate.asp](http://www.w3schools.com/jquery/jquery_animate.asp)

# ZADANIA

## Zadanie 34 – *jQuery (animation, chaining)*

Korzystając z jQuery dodaj funkcjonalność pozwalającą poprzez najechania na strzałkę po prawej stronie witryny powrócić do jej początku.

### Materiały pomocnicze:

- [http://www.w3schools.com/jquery/jquery\\_selectors.asp](http://www.w3schools.com/jquery/jquery_selectors.asp)
- [http://www.w3schools.com/jquery/jquery\\_events.asp](http://www.w3schools.com/jquery/jquery_events.asp)
- [http://www.w3schools.com/jquery/jquery\\_css.asp](http://www.w3schools.com/jquery/jquery_css.asp)
- [http://www.w3schools.com/jquery/jquery\\_animate.asp](http://www.w3schools.com/jquery/jquery_animate.asp)

# ZADANIA

## Zadanie 35 [opcjonalne] – *jQuery UI*

Za pomocą jQuery UI dodaj okienko tooltipa dla wybranej kontrolki na stronie kontakt.

### Materiały pomocnicze:

- <https://learn.jquery.com/jquery-ui/getting-started>

# ZADANIA

## **Zadanie 37 – AJAX (GET)**

Napisz kod sprawdzający po każdym wprowadzeniu znaku z klawiatury, za pomocą AJAX-u (XMLHttpRequest, JavaScript) czy login jest dostępny– sprawdzenie względem serwisu WCF. Zastosuj GET.

## **Zadanie 38 – AJAX (POST)**

Napisz kod sprawdzający po każdym wprowadzeniu znaku z klawiatury, za pomocą AJAX-u (XMLHttpRequest, JavaScript) czy hasło spełnia wymagania kompleksowości – sprawdzenie względem serwisu WCF. Zastosuj POST.

### **Materiały pomocnicze:**

- <http://www.codeproject.com/Articles/167159/How-to-create-a-JSON-WCF-RESTful-Service-in-sec>
- [http://www.w3schools.com/ajax/ajax\\_xmlhttprequest\\_create.asp](http://www.w3schools.com/ajax/ajax_xmlhttprequest_create.asp)
- [http://www.w3schools.com/ajax/ajax\\_xmlhttprequest\\_send.asp](http://www.w3schools.com/ajax/ajax_xmlhttprequest_send.asp)
- [http://www.w3schools.com/ajax/ajax\\_xmlhttprequest\\_response.asp](http://www.w3schools.com/ajax/ajax_xmlhttprequest_response.asp)
- [http://www.w3schools.com/ajax/ajax\\_xmlhttprequest\\_onreadystatechange.asp](http://www.w3schools.com/ajax/ajax_xmlhttprequest_onreadystatechange.asp)
- <http://www.w3schools.com/ajax/ajax.asp>

# ZADANIA

## **Zadanie 39** – JQuery *AJAX (GET)*

Napisz kod sprawdzający po każdym wprowadzeniu znaku z klawiatury, za pomocą AJAX-u (JQuery) czy login jest dostępny– sprawdzenie względem serwisu WCF. Zastosuj GET.

## **Zadanie 40** – JQuery *AJAX (POST)*

Napisz kod sprawdzający po każdym wprowadzeniu znaku z klawiatury, za pomocą AJAX-u (JQuery) czy hasło spełnia wymagania kompleksowości – sprawdzenie względem serwisu WCF. Zastosuj POST.

### **Materiały pomocnicze:**

- [http://www.w3schools.com/jquery/jquery\\_ajax\\_get\\_post.asp](http://www.w3schools.com/jquery/jquery_ajax_get_post.asp)
- [http://www.w3schools.com/jquery/ajax\\_ajax.asp](http://www.w3schools.com/jquery/ajax_ajax.asp)

# ZADANIA

## Zadanie 41 [Opcjonalne] – *WebApi, Canvas, AJAX*

Napisz stronę wyświetlającą wyniki sprzedaży za pomocą stosując: Web API, AJAX, JavaScript (jQuery), Charts.js i Canvas.

### Materiały pomocnicze:

- <http://www.codeproject.com/Articles/549152/Introduction-to-ASP-NET-Web-API>
- <http://www.codeproject.com/Articles/424461/Implementing-Consuming-ASP-NET-WEB-API-from-JQuery>
- [http://www.w3schools.com/html/html5\\_canvas.asp](http://www.w3schools.com/html/html5_canvas.asp)
- <http://www.chartjs.org>

# ZADANIA

## Zadanie 42 - *Walidacja*

Oprogramuj stronę kontaktu tak, aby nie można było wprowadzić w pole wiek, wartości innej niż z przedziału 18-100.

**Materiały pomocnicze:**

- [http://www.w3schools.com/js/js\\_validation.asp](http://www.w3schools.com/js/js_validation.asp)

# ZADANIA

## Zadanie 43 – *Walidacja HTML5*

Dodaj do strony kontrolkę pozwalającą utworzyć konto użytkownika (zasymulować) – czyt. podać login oraz hasło. Napisz własny kod walidacji HTML5, stosując `setCustomValidity` zgłaszający błąd jeśli użytkownik jako login podał administrator czy admin.

### Materiały pomocnicze:

- <https://msdn.microsoft.com/en-us/library/windows/apps/hh441292.aspx>



# ZADANIA

## Zadanie 44 – Obsługa błędów

Dla aplikacji „Lotto” napisz obsługę wszelkich możliwych błędów:

- Wyświetlanych stosowne komunikaty w oknie alert
- Modyfikujących wygląd kontrolek w przypadku wystąpienia błędu

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_errors.asp](http://www.w3schools.com/js/js_errors.asp)
- [http://www.w3schools.com/js/js\\_validation\\_api.asp](http://www.w3schools.com/js/js_validation_api.asp)

# ZADANIA

## Zadanie 45 – *Testowanie*

Zdebuguj i przetestuj kod generowania testu dla sprawdzenia wiedzy.

### Materiały pomocnicze:

- [http://www.w3schools.com/js/js\\_debugging.asp](http://www.w3schools.com/js/js_debugging.asp)

# ZADANIA

## Zadanie 46 – *Kompresja*

Zadbaj, aby kod dla generowania testu był umieszczony w osobnej bibliotece. Dokonaj kompresji biblioteki.

**Materiały pomocnicze:**

- <http://jscompress.com>

# ZADANIA

## **Zadanie 47** – *File System, Drag& Drop*

Dodaj na stronie kontaktu możliwość wczytania za pomocą JavaScript zdjęcia, które zostanie dołączone do wysyłanego żądania.

Dodaj funkcjonalność pozwalającą wczytać zdjęcie poprzez przeciągnięcie jego z pulpitu na stronę (Drag and Drop).

### **Materiały pomocnicze:**

- <http://www.html5rocks.com/en/tutorials/file/filesystem>
- [http://www.w3schools.com/html/html5\\_draganddrop.asp](http://www.w3schools.com/html/html5_draganddrop.asp)

# ZADANIA

## **Zadanie 48 - *Multimedia***

Osadź na stronie start film prezentujący ofertę firmy za pomocą tagu video.

Oprogramuj tag video, tworząc własny player.

**Materiały pomocnicze:**

- [http://www.w3schools.com/html/html5\\_video.asp](http://www.w3schools.com/html/html5_video.asp)

# ZADANIA

## Zadanie 49 – *Geolocation*

Dodaj na stronie mapę z zaznaczoną Twoją lokalizacją – skorzystaj z Google Maps.

Za pomocą funkcjonalności Geolokalizacji dodaj na mapie pozycje, w której znajduje się klient.

### Materiały pomocnicze:

- [http://www.w3schools.com/html/html5\\_geolocation.asp](http://www.w3schools.com/html/html5_geolocation.asp)
- <https://developers.google.com/maps/documentation/javascript/tutorial>

# ZADANIA

## Zadanie 50 - *WebStorage*

Zdefiniuj dla strony kontakt przechowywanie danych po stronie klienta (localStorage), tak aby w przypadku zamknięcia i ponownego otworzenia przeglądarki dane zostały odtworzone.

### Materiały pomocnicze:

- [http://www.w3schools.com/html/html5\\_webstorage.asp](http://www.w3schools.com/html/html5_webstorage.asp)

# ZADANIA

## Zadanie 51 – *Offline support*

Zdefiniuj dla witryny wsparcie offline, tak aby strona kontaktu była cachowana po stronie klienta, i dostępna w przypadku problemów z siecią.

### Materiały pomocnicze:

- [http://www.w3schools.com/html/html5\\_app\\_cache.asp](http://www.w3schools.com/html/html5_app_cache.asp)



# ZADANIA

## Zadanie 52 - SVG

Za pomocą SVG zdefiniuj zdefiniuj stoper pokazujący czas z animacją co 10 sekund.

**Materiały pomocnicze:**

- [http://www.w3schools.com/html/html5\\_svg.asp](http://www.w3schools.com/html/html5_svg.asp)

# ZADANIA

## Zadanie 53 - *Canvas*

Za pomocą Canvas-u zdefiniuj stoper pokazujący czas z animacją co 10 sekund.

### Materiały pomocnicze:

- [http://www.w3schools.com/html/html5\\_canvas.asp](http://www.w3schools.com/html/html5_canvas.asp)

# ZADANIA

## Zadanie 54 – *Web Workers*

Napisz zadanie w tle (Web Worker) co minutę pobierające z serwera WCF informacje o aktualnościach.

### Materiały pomocnicze:

- [http://www.w3schools.com/html/html5\\_webworkers.asp](http://www.w3schools.com/html/html5_webworkers.asp)

# ZADANIA

## Zadanie 55 – *Web sockets*

Stosując HTML5 i C# opracuj mechanizm pozwalający na czatowanie na witrynie pomiędzy odwiedzającymi (WebSockets).

### Materiały pomocnicze:

- [http://www.w3schools.com/html/html5\\_serversentevents.asp](http://www.w3schools.com/html/html5_serversentevents.asp)