

Sprawozdanie nr 02

Temat: Programowanie obiektowe, aplikacja Todo

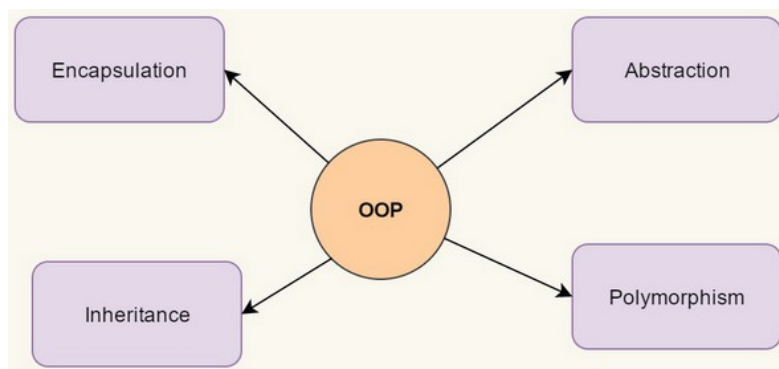
Teoria:

Programowanie obiektowe (ang. Object-oriented Programming) jest paradygmatem programowania, w którym programy definiuje się za pomocą obiektów.

Obiekty są elementami, będącymi reprezentantami (tzw. instancjami klas).

Klasa to niejako definicja swojego własnego typu danych wraz z dostępnymi dla niej atrybutami i procedurami (metodami).

Według paradygmatu obiektowego, o **programie komputerowym** powiemy – zbiór obiektów komunikujących się pomiędzy sobą w celu wykonywania zadań, a każdy obiekt stworzony jest według przepisu podanego w klasie.



Głównymi cechami paradygmatu obiektowego są:

- Abstrakcja danych (Data Abstraction)
 - Hermetyzacja (Encapsulation)
 - Polimorfizm (Polymorphism)
 - Dziedziczenie (Inheritance)
-

Przebieg zadania:

Główna część aplikacji.

Your Personal To Do List

Done	Priority	To Do	Added
------	----------	-------	-------

Create a database

Add a task

Priority of the task:

Name of the task:

Add the entry

Refresh the table

Your last action will be displayed here.

The site was made by Filip Maćkowiak

Opis poszczególnych okien aplikacji

Przycisk tworzenia bazy danych oraz/ lub tabeli, do użycia w razie awarii.

Formularz umożliwiający dodawanie rekordów do listy. Ustawiamy w nim:

- Priorytet zadania (im niższy tym ważniejszy - wyżej się wyświetla)
- Treść zadania

Paragraf wyświetlający ostatnio wykonaną akcję, tj.:

- Utworzenie bazy danych i/ lub tabeli
- Dodanie rekordu
- Usunięcie rekordu
- Edycja rekordu

Główne okienko - Lista (tabela) zadań do zrobienia

Przycisk odświeżania listy, do użycia po wykonaniu akcji.

Your last action will be displayed here.

The site was made by Filip Maćkowiak

Po dodaniu rekordów

Przy każdym wpisie, checkbox do zaznaczenia, czy wykonaliśmy już dane zadanie

Data dodania zadania

Ikonka edycji wpisu

Ikonka usuwania wpisu

Widoczny komunikat po dodaniu wpisu.

The site was made by Filip Maćkowiak

Done	Priority	To Do	Added
<input type="checkbox"/>	1	Write a report	2021-05-11 22:33:48
<input type="checkbox"/>	3	Exercise	2021-05-11 22:34:07

Create a database

Add a task

Priority of the task:

Name of the task:

Add the entry

Refresh the table

Successfully added your task

Wykorzystane w projekcie technologie webowe:

- HTML;
- CSS;
- JavaScript;
- PHP

Dodatkowo została również użyta technika **AJAX**, dla interakcji użytkownika z serwerem, bez potrzeby przeładowywania strony oraz biblioteka **jQuery** dla użycia funkcji `$(document).ready()`.

W projekcie, wszystkie zapisane zadania znajdują się na bazie danych o nazwie: todoapp

W tabeli: todolist

W lokalizacji 'localhost'.

O następującej strukturze:

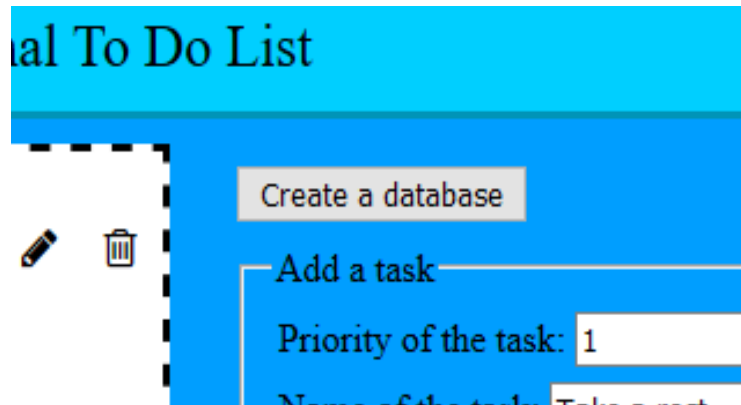
```
todoapp. todolist
```

Id	int	PK
Nr	int	
ToDo	text	NULL
M_Date	datetime	NULL

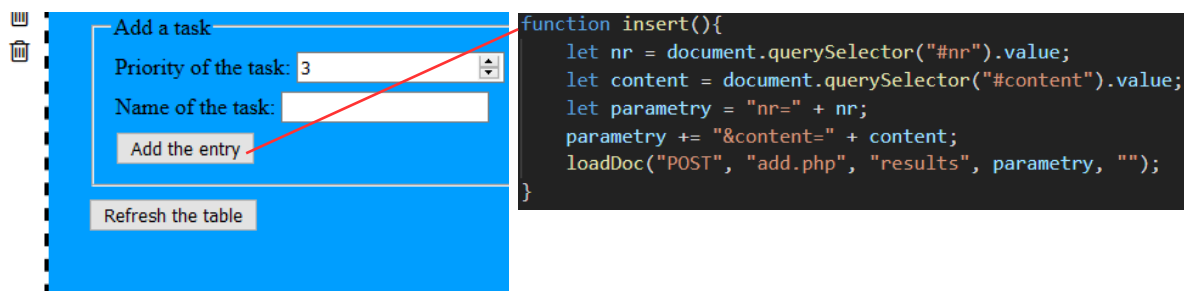
Główne pliki projektu, to:

- index.html – Strona główna (strona startowa)
- style.css – Kaskadowy arkusz stylów
- ajax_core.js – Obsługa AJAX
- jquery.js – Biblioteka jQuery

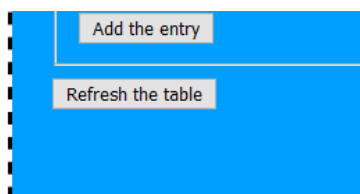
- script.js – Funkcjonalności po stronie przeglądarki (m. in. podpięcie zdarzeń 'click' pod poszczególne przyciski oraz wykonanie funkcji za pośrednictwem AJAX)
- create.php – Skrypt tworzący bazę danych oraz tabelę (jeśli nie istnieją), wywoływany jest on zawsze przy załadowaniu strony oraz można go uruchomić przyciskiem: „**Create a Database**”



- add.php – Skrypt wykonujący operację INSERT – wstawiania do bazy nowego rekordu z zadaniem. Uruchamiany za pomocą przycisku: „**Add the entry**” oraz za pośrednictwem funkcji `insert()` w pliku script.js, wysyłającej wymagane parametry do add.php.



- select.php – Skrypt wykonujący zapytanie SELECT do bazy. Wyciągający z niej wszystkie rekordy. W tym pliku, dynamicznie budowane jest ciało tabeli, za każdym razem kiedy jest on wywoływany. Wyciągnięte dane są zapisywane do obiektu `$tasks`, który jest instancją klasy `Entry` zadeklarowanej na początku pliku. Skrypt wywoływany jest przy każdym załadowaniu strony, przy dodawaniu rekordu do bazy oraz przede wszystkim po kliknięciu przycisku: „**Refresh the table**”




```
while ($record = $result->fetch_assoc()){
    $tasks->setId($record['Id']);
    $tasks->setNumber($record['Nr']);
    $tasks->setContent($record['ToDo']);
    $tasks->setMdate($record['M_Date']);
    $data .= "<tr><td><input type='checkbox' id='ch'></td><td>{$tasks->getNumber()}</td><td>{$tasks->getContent()}</td><td>{$tasks->getMdate()}</td><td><i class='icon-pencil' onclick=editR({$tasks->getId()},\"{$tasks->getContent()}\")></i></td><td><i class='icon-trash-empty' onclick=deleteR({$tasks->getId()})></i></td></tr>";
    $i++;
}
```

Skrypt zapisujący ściągnięte dane i tworzący za ich pomocą, dynamicznie, ciało tabeli.

W tym miejscu również przypisywane jest zdarzenie *onclick* prowadzące: dla ikony śmietnika do funkcji usuwania oraz ołówka do funkcji edycji rekordu.

- delete.php – Skrypt wykonujący operację DELETE FROM na bazie, usuwający rekord bazując na id. Uruchamiany jest on ikonką śmietnika przy zadaniu. Po jej kliknięciu, aktywowana zostaje funkcja deleteR, do której wysyłane jest Id danego rekordu. Następnie ten parametr wysyłany jest dalej do pliku delete.php metodą POST, gdzie wykonywana jest operacja usuwania.

 **onclick**

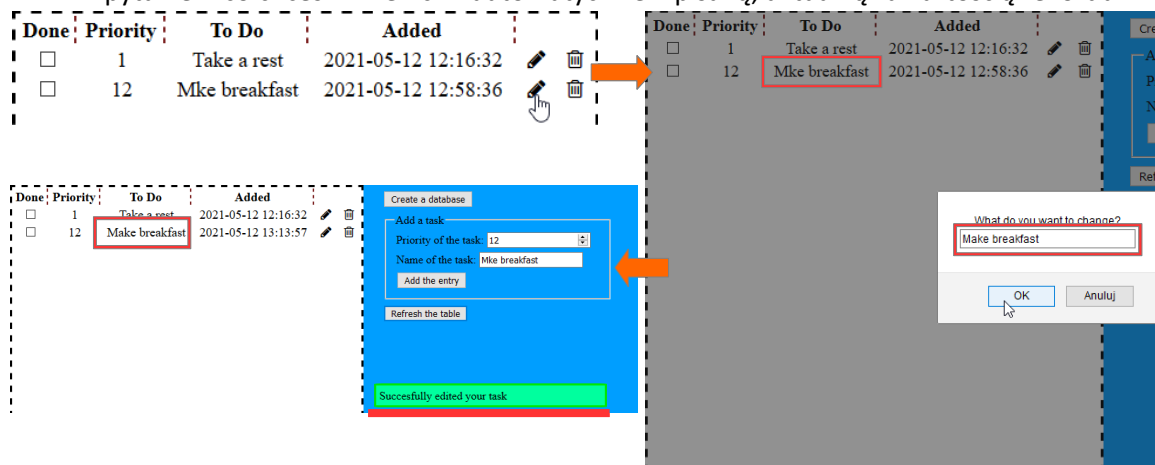
```
function deleteR(id){
    parametry = "id=" + id;
    loadDoc("POST", "delete.php", "results", parametry, "");
}
```

Skrypt JavaScript

```
$sql = "DELETE FROM todolist WHERE Id={$id_to_delete}";
$result = $link->query($sql);
```

Zapytanie w php

- edit.php – Skrypt wykonujący operację UPDATE, umożliwiającą edycję utworzonego już rekordu. Uruchamiany jest on kliknięciem ikonki ołówka. Po jej kliknięciu pojawia się okno z pytaniem co chcesz zmienić i z automatycznie wpisaną, aktualną zawartością rekordu.



```
function editR(id, content){
    let newContent = prompt("What do you want to change?", content);
    if (newContent != null)
    {
        let parametry = "id=" + id;
        parametry += "&content=" + newContent;
        loadDoc("POST", "edit.php", "results", parametry, "");
    }
}
```

Skrypt JS (w razie kliknięcia cancel/ anuluj skrypt edit.php się nie wywoła)