



# Moduł File System

## Ćwiczenia #1

© PIOTR SIEWNIAK 2021

### Zadanie 1

W folderze aplikacji utworzyć katalog o nazwie `data` (`./data`). W nowoutworzonym katalogu `data` utworzyć plik tekstowy o nazwie `data.dat` (`./data/data.dat`). W pliku `data.dat` zapisać pełną nazwę szkoły. Wyświetlić kontrolnie zawartość pliku `data.dat` w konsoli.

Wykonać aplikację w trzech wariantach – utworzyć plik `data.dat` przy wykorzystaniu:

- 1) metody `open()` (*wariant I*);
- 2) metody `writeFile()` (*wariant II*);
- 3) metody `appendFile()` (*wariant III*).

### Zadanie 2

Przygotować w folderze `data` (`./data`) plik tekstowy `data.dat` zawierający w kolejnych wierszach wykaz zawodów, w których realizowane jest kształcenie zawodowe w szkole (np. technik programista, technik informatyk itd.). Nazwa każdego z zawodów powinna być zapisana w osobnym wierszu.

Wyświetlić zawartość pliku `data.dat`:

- a) na stronie WWW;
- b) w konsoli.

Wykonać aplikację w różnych wariantach odpowiadających metodzie wykorzystanej do odczytu pliku `data.dat` – np. `readFile()`, `readFileSync()`, `read()`.

### Zadanie 3

Treść, jak w **zadaniu 2**. Dodatkowo, dopisać do pliku `data.dat` na jego końcu (w nowych wierszach) następujące zawody: technik teleinformatyk, technik elektryk. Wyświetlić kontrolnie zawartość pliku `data.dat` w konsoli i na stronie WWW.

Uwzględnić system komunikatów kontrolnych dla użytkownika (np. w przypadku wystąpienia błędu).

### Zadanie 4

Przygotować stronę główną wraz z zestawem kilku podstron witryny WWW dotyczącej wybranej tematyki, np. hobby, szachów, samochodów, programowania komputerów itp. Zapisać strony w wybranym katalogu w folderze aplikacji, np. katalogu `./pages`.

Napisać aplikację pozwalającą na uruchomienie (wyświetlenie) zaprojektowanej witryny (wraz z wszystkimi jej składnikami – podstronami) w przeglądarce użytkownika. Uwzględnić oczywiście samodzielnie zaimplementowany serwer HTTP.

### Zadanie 5

---

Przygotować plik `data.json` zawierający dane samochodu osobowego ujęte w formacie JSON. Zapisać plik `data.json` w folderze `./data`. Uwzględnić następujące dane samochodu: marka, model, rok produkcji, cena, data pierwszej rejestracji oraz wykaz powiatów, w których samochód był dotychczas rejestrowany.

Zaimplementować aplikację pozwalającą na odczytanie zawartości pliku `data.json` i wyświetlenie jej:

- a) w konsoli;
- b) na stronie WWW.

### Zadanie 6

---

Utworzyć w folderze aplikacji (`./`) podkatalogi: `./krakow` i `./wroclaw`. Ponadto, utworzyć w każdym z tych katalogów plik `uczelnie.json` z danymi w formacie JSON zawierające wykaz uczelni technicznych w danym mieście. Wyświetlić kontrolnie zawartość pliku `uczelnie.json` z każdego wymienionego powyżej katalogu na stronie WWW. Wszystkie operacje wykonać z poziomu aplikacji Node.

Dodatkowo, wyświetlić na wspomnianej stronie WWW:

- a) rozmiar każdego z plików `uczelnie.json`;
- b) zawartość katalogu głównego aplikacji (`./`) oraz katalogów `./krakow` i `./wroclaw`.

### Zadanie 7

Wykonać dodatkowo wszystkie zadania (ćwiczenia) dotyczące buforów (zasobów klasy `Buffer`) – z tym, że zamiast łańcuchów znaków wykorzystać pliki tekstowe.

### Zadanie 8

Napisać aplikację pozwalającą na odczyt w zadanej kolejności zawartości trzech plików tekstowych w sposób asynchroniczny. Wykonać aplikację w dwóch wariantach – odczyt każdego z plików zrealizować:

- a) w całości (*wariant I*);
- b) kawałkami (porcja danych po porcji danych) (*wariant II*).

### Zadanie 9

Wykonać **zadanie 4** korzystając z promisywnego API modułu `fs`.

### Zadanie 10

Wykonać **zadanie 6** korzystając z promisywnego API modułu `fs`.