



Рис. 1 Нанofільтр для опріснення морської води [7].

Висновки . В даній статті нам вдалося з'ясувати чи шкідливо вживати морську воду для живого організму та як впливає морська вода на організм людини, а також було розглянуто деякі методи та установки опріснення морської води. Було з'ясовано, що морську воду можна пити, якщо її опріснити за допомогою нанofільтру, який створили фізики з США.

ЛІТЕРАТУРА:

1. Опріснення морської води як альтернативне джерело отримання питної води. [Електронний ресурс]. Режим доступу: <http://nuft.edu.ua/vstupnyku/majster-klasi/oprsnennya-morsko-vodi>
2. Як опріснити морську воду. [Електронний ресурс]. Режим доступу: <http://zapitay.com.ua/yak-oprisniti-morsku-vodu.html>
3. Медична енциклопедія. Опріснення води. [Електронний ресурс]. Режим доступу: http://medical-enc.com.ua/opresnenie_vody.html
4. Знесолення та опріснення води. [Електронний ресурс]. Режим доступу: https://stud.com.ua/27748/tovarovnavstvo/znesolennya_oprisnennya_vodi
5. Мембранні фільтри для води. [Електронний ресурс]. Режим доступу: <http://voday.com.ua/Fil-try-dlya-vody/Membrann-f-l-tri-dlya-vodi.html>
6. Американські фізики винайшли дешевий спосіб опріснення морської води. [Електронний ресурс]. Режим доступу: <http://ecotown.com.ua/news/Amerykanski-fizyky-vynayshly-deshevyuy-sposib-oprisnennya-morskoyi-vody/>
7. Water desalination with a single-layer MoS₂ nanopore. [Електронний ресурс]. Режим доступу: <https://www.nature.com/articles/ncomms9616>

Напованець Ю.

Науковий керівник – Вельгач А. В.

ПОРІВНЯННЯ ШВИДКОДІЇ СНІПЕТІВ ВИВЕДЕННЯ РЕСУРСІВ У CMS MODX REVOLUTION

При роботі з CMS ModX Revolution виникає задача виведення на одній web-сторінці групи ресурсів з використанням того чи іншого шаблону. Наприклад, виведення переліку постів блогової сторінки, сторінки портфолію, стрічки повідомлень, тощо. Для розв'язання задач подібного типу в системі ModX Revolution використовуються сніпети. **Сніпет** (англ. *snippet* — фрагмент, уривок) в практиці програмування — невеликий фрагмент вихідного коду або тексту, придатний для повторного використання. Сніпети не є заміною процедур, функцій або інших подібних понять структурного програмування. Вони зазвичай використовуються для більш легкого читання коду функцій, які без їх використання виглядають занадто перевантаженими деталями, або для усунення повторення окремої частини ділянки коду. Сніпет в контексті CMS ModX Revolution це частина PHP коду, що забезпечує виведення користувацького динамічного контенту, що сайт повинен генерувати за запитом.

Сніпети дозволяють додавати таку функціональність як навігацію, «хлібні крихти», записи соціальних мереж, пошук, фотогалереї, форми контактів, коментарі і т.п. у ModX сайт. Кожен встановлений на систему управління сайтом сніпет має своє призначення і параметри, за допомогою яких можна змінити шаблон та формат виведення даних на сторінку сайту. Сніпети можна створювати як власноруч, так і встановлювати готові з бібліотеки сніпетів. На даний момент в бібліотеці ModX Revolution нараховується більше тисячі сніпетів, які відрізняються чи функціональним призначенням чи принципами реалізації.

Метою даної статті є порівняння швидкодії і гнучкості налаштувань двох ідентичних за своїм призначенням сніпетів `GetResources` і `PdoResources`.

Сніпет `PdoResources` розроблявся з метою заміни `GetResources`. В ньому виправили недоліки останнього і оснастили додатковим набором параметрів, які практично не обмежені в можливості вибірки ресурсів різних класів з бази даних любих полів і TV. Завдяки спрощеному зверненню до баз даних сніпет `PdoResources` при великих обсягах інформації працює швидше.

Синтаксис розглянутих сніпетів практично не відрізняється. Відмінності проявляються лише тоді, коли необхідно підключити TV-параметри у вивід ресурсів чи вивести дату публікації. Для сніпета `GetResources` у параметрі `includeTVs` значення 1 включає TV-параметри у вивід, а 0 не включає (`&includeTVs=1`). У випадку з `PdoResources` необхідно вказати імена TV-параметрів, які необхідно вивести. Якщо їх декілька, то необхідно перелічити всі через кому (`&includeTVs=img-news, tags`). Для дати в сніпеті `GetResources` використовується модифікатор `strtotime` для форматування дати та часу. Всі ресурси за замовчуванням зберігаються в вигляді `timestamp`, але конвертуються в нормальні дати через використання об'єктів `modResource`. Але дату необхідно перевести назад в `timestamp`. Таким чином, подвійна конвертація сповільнює роботу сніпета. Сніпет `PdoResources` працює одразу з базою даних, тому в чанк надходить `timestamp`, який непотрібно додатково обробляти.

Швидкодія цих двох сніпетів напряму залежить від типу вибірки з бази даних. Розглянемо, за який час ці два сніпети відобразять 10, 50, 100 та 500 аналогічних ресурсів (для експерименту використано он-лайн сервіс `Page Speed Insights`). Спочатку візьмемо просту вибірку інформації без оформлення і додаткових TV-параметрів. Код виклику сніпетів має наступний вигляд:

Для сніпета `GetResources`:

```
[[!getResources?&parents=`2`&limit=`n`]]
```

Для сніпета `PdoResources`:

```
[[!pdoResources? &parents=`2`&limit=`n`]]
```

Порівняння швидкодії відображено в наступній таблиці:

Таблиця 1

Кількість сторінок	<code>getResources</code>	<code>pdoResources</code>
10	0.1592 с.	0.1304 с.
50	0.3502 с.	0.1495 с.
100	0.5500 с.	0.1762 с.
500	2.1891 с.	0.3045 с.

У наступному кроці нашого дослідження у вибірку додано оформлення. Для цього додатково створено чанк, всередині якого описано динамічний HTML код, згідно якого буде здійснюватись відображення (`<p>[[+idx]] [[+pagetitle]]</p>`). У цьому випадку отримано наступні результати:

Таблиця 2

Кількість сторінок	<code>getResources</code>	<code>pdoResources</code>
10	0.1721 с.	0.1384 с.
50	0.3994 с.	0.1687 с.
100	0.6915 с.	0.2047 с.
500	2.7538 с.	0.4851 с.

Таким чином результати майже не змінилися. Сніпет `PdoResources` швидше виконує поставлене завдання. При додаванні ще двох TV-параметрів порівняння швидкодії дає наступний результат:

Таблиця 3

Кількість сторінок	<code>getResources</code>	<code>pdoResources</code>
10	0.2029 с.	0.1383 с.
50	0.5014 с.	0.1747 с.
100	0.8860 с.	0.2230 с.
500	4.0989 с.	0.5248 с.

У даному випадку сніпет `GetResources` працює повільніше, адже приєднання TV-параметрів здійснюється через додатковий запит, у той час як сніпет `PdoResources` приєднує ці параметри до основного

запиту. Подальші дослідження показали, що в залежності від кількості додаткових TV-параметрів та умов вибірки розрив в швидкодії буде лише збільшуватись.

Існує ще декілька відмінностей між цими двома сніпетами. В `GetResources` є деякі параметри, які поки що не увійшли до `PdoResources`, а саме:

Чанки `@FILE` і `@INLINE`. Перший префікс дозволяє використовувати файл замість чанка в базі даних у якості шаблону. Другий префікс дозволяє задати розмітку для використання в якості шаблону безпосередньо в значенні параметру.

Параметр `&toSeparatePlaceholders`. Надає кожному виведеному ресурсу власне додаткове ім'я плейсхолдера у вигляді суфікса із значення цього параметра та його порядкового номера.

Параметр `&tvFilters` на даний момент не реалізований, але є можливість вказати TV в `&where`, наприклад

```
[[!pdoResources? &includeTVs=`test` &where={`"test:!=":""}]]
```

Параметри `&sortByTV`, `&sortdirTV` і `&sortByTVType` також не реалізовані на даний момент.

Параметри `&prepareTVs`, `&prepareTVList`, `&processTVs`, `&processTVList` відсутні і на даний момент альтернативи немає.

Параметр `&debug` замінений на більш продуктивний і зручніший `&showLog=`1``.

Відсутні параметри сортування `&sortByAlias`, `&sortByEscaped`.

Відсутні параметри `&tplCondition`, `&conditionalTpls` і `&tplOperator`.

Інші параметри ідентичні, найбільш часто використовувани з них наведені в наступній таблиці:

Таблиця 4

Функція параметра	виклик в <code>GetResources</code>	виклик в <code>PdoResources</code>
Ім'я чанка з шаблоном	<code>&tpl</code>	<code>&tpl</code>
Ім'я чанка з шаблоном для першого елемента	<code>&tplFirst</code>	<code>&tplFirst</code>
Ім'я чанка з шаблоном для останнього елемента	<code>&tplLast</code>	<code>&tplLast</code>
Ім'я чанка з шаблоном для кожного другого елемента	<code>&tplOdd</code>	<code>&tplOdd</code>
Список ідентифікаторів батьківських ресурсів	<code>&parents</code>	<code>&parents</code>
Список ідентифікаторів ресурсів для включення в результат вибірки	<code>&resources</code>	<code>&resources</code>
Глибина пошуку дочірніх ресурсів в батьківському	<code>&depth</code>	<code>&depth</code>
Масив додаткових параметрів вибірки закодованих в JSON-формат	<code>&where</code>	<code>&where</code>
Відобразити ресурси, які приховані для меню	<code>&showHidden</code>	<code>&showHidden</code>
Обмеження кількості ресурсів вибірки	<code>&limit</code>	<code>&limit</code>
Показати видалені ресурси	<code>&showDeleted</code>	<code>&showDeleted</code>
Пропуск результатів від початку	<code>&offset</code>	<code>&offset</code>

Висновок. Для великої вибірки ресурсів з додаванням оформлення, TV-параметрів та інших додаткових умов доцільніше використовувати сніпет `PdoResources`, адже за рахунок спрощеного запиту до бази даних, відсутності додаткових конвертувань та роботі напряду з базою даних, час відображення групи ресурсів на сторінці значно менший. Аналогічні висновки стосуються не лише цих сніпетів, а й інших, наприклад: `getPage` замінили на `pdoPage`, `Wayfinder` на `pdoMenu`, `GoogleSitemap` на `pdoSitemap`, `getUser` на `PdoUser`, `Breadcrumbs` на `pdoCrumb`s та ін.. Нові сніпети більш продуктивніші. Це досягається шляхом роботи через PDO без затрат на створення додаткових запитів до бази даних.

ЛІТЕРАТУРА

1. MODX 2011. Сайт по веб-розробці [Електронний ресурс]. Поняття сніпета. Режим доступу: <http://modx.ws/uroki/uroki-modx-revolution/urok-snippet.html>
2. MODX 2011. Сайт по веб-розробці [Електронний ресурс]. `PdoResources`. Режим доступу: <https://docs.modx.pro/komponentyi/pdotools/snippetyi/pdoresources>

3. MODX 2011. Сайт по веб-розробці [Електронний ресурс]. GetResources. Режим доступу: <http://modx.ws/getresources>
4. MODX 2018. Making Awesome Easy Since 2004. [Електронний ресурс]. PdoTools. Режим доступу: <https://bezumkin.ru/sections/components/1498/>
5. REALADMIN.RU 2018. [Електронний ресурс]. Заміна GetResources на PdoResources. Режим доступу: <https://realadmin.ru/saytostroy/pdoresources.html>

Плюсквік-Баран Ю.

Науковий керівник – доц. Галан В. Д.

ІСТОРИЧНИЙ ТА ЛОГІКО-МАТЕМАТИЧНИЙ АСПЕКТ ВИКЛАДАННЯ ГЕОМЕТРІЇ У ШКОЛИ

Формування вмінь та навичок, засвоєння глибоких і міцних знань, а найголовніше, застосування їх у житті, на практиці – основне завдання школи в плані вивчення геометрії та математики в цілому. Як домогтися того, щоб на уроках геометрії не було байдужих учнів, а усі рівноцінно брали активну участь в процесі? Розуміння навчально-пізнавального процесу є відповіддю на дане запитання, адже саме він є рушійною силою у навчальній діяльності учня.

Процес пізнання не може принести результатів без мотивації та виникнення мотивів до вивчення геометрії. «Формування мотивів навчання – це створення у школі умов для появи внутрішніх спонукань до навчання, усвідомлення їх учнем і подальшим саморозвитком його своєї мотиваційної сфери.» [3] Стимулювати розвиток мотивації учнів необхідно за допомогою психологічних прийомів. Мета розвитку навчальної мотивації школярів полягає в тому, щоби перетворювати байдуже ставлення учня до навчання на позитивне сприйняття учнем навчального процесу.

Значну роль під час вивчення геометрії відіграє створення цілісної системи підкріплення пізнавальної діяльності учнів. Тобто сукупність спонукань до знань, допитливості, пізнавальної діяльності, зацікавленості в предметі а також в пошуку істини. Одним з методів розвитку пізнавального інтересу учнів є використання елементів історії науки в процесі вивчення курсу «Геометрії».

«Використання принципу історизму впродовж усього курсу «Геометрія» в основній школі допоможе розвинути стійкий пізнавальний інтерес в учні основної школи до вивчення геометрії.» [4] Учні 7 класу розпочинають знайомство із новим предметом – «Геометрія». У цьому курсі змінюються способи оформлення та розв'язування задач, доведеться міркувати іншим чином для доведення теорем, до процесу логічного мислення додається просторова уява (вміння уявляти геометричні фігури на площині та в просторі).

Окремим питанням історії математики присвячено праці таких авторів, як Бевз Г. П., Бевз В. Г., Віленкін Н. Я., Бородін А. І. та ін. Історію математики в школі розглянуто у роботах Тадеєва В. О., Зоріної Л. Я., Глейзера Г. І.

Підручники В. О. Тадеєва з геометрії для 7-11 класів містять матеріали з історії геометрії у повному обсязі. У них описано походження науки «Геометрія», розділів «Планіметрія» та «Стереометрія», різних математичних термінів, зокрема таких, як *аксіома, теорема, паралельний, перпендикулярний, синус, косинус, тангенс*, а також про походження назв геометричних фігур. У книгах присутні історичні довідки в яких висвітлюються питання становлення геометрії, аксіоми Евкліда, описано стародавні задачі, які неможливо розв'язати за допомогою циркуля та лінійки та ін. Підручники містять біографії відомих математиків: Евкліда, Піфагора, М. В. Остроградського, Р. Декарта та інших учених.

На перших уроках геометрії у 7 класі основне завдання вчителя – зацікавити учнів. У цьому йому може допомогти історія математики, яка пояснить походження назви предмета, а також покаже, що ця наука зародилась в процесі життєдіяльності людини (полегшувала працю, допомагала вигравати війни), тобто має пряме практичне значення. Для кращого запам'ятовування термінів та символів пропонується пояснювати походження слів, а також показувати різні форми запису символів, які виникали в продовж періоду розвитку математики як науки.

Розглядаючи питання з історії науки необхідно чітко дотримуватись хронології подій, використовувати сучасні інформаційні технології, що дозволить використовувати принцип наочності на уроках геометрії. Важливо демонструвати портрети учених, а не просто називати їх прізвища. Також важливим аспектом вивчення історії науки в курсі «Геометрія» є самостійна пошуково-дослідницька робота учнів, що полягає у дослідженні питань походження математичної символіки та виникнення термінів. Самостійна дослідницька робота допомагає підтримувати пізнавальний інтерес учнів.