

- «Ділове досьє. Персонал»;
- «Персонал Про», «Персонал ВНЗ», «Персонал – Бізнес»;
- «Кадри».

Отже, наявність реальної інформаційної системи дає широкі можливості, наприклад: здійснювати планування наймання, адаптації, атестації кадрового резерву, навчання, мотивації; підвищити активність керівників підрозділів у керуванні персоналом; оцінювати ефективність роботи служби персоналу і керування персоналом у цілому в організації; керувати кваліфікацією співробітників та інше. Забезпечити таку функціональність інформаційної системи потрібно на усіх підприємствах.

АСИНХРОННИЙ JAVASCRIPT

Одегова Є.О.

*Науковий керівник – Погребняк Б.І., канд. техн. наук, доцент
(Харківський національний університет радіоелектроніки)*

Нехай маємо наступний фрагмент коду, що складається з визначення двох функцій: `outOne()` та `outTwo()`, які відображають в консолі числа 1 і 2, відповідно.

```
function outOne() {
  console.log(1);
}
function outTwo() {
  console.log(2);
}
outOne();
outTwo();
```

Після його запуску, наприклад, в консолі Google Chrome, як і очікувалося, на виході отримаємо:

```
1
2
```

У цьому сенсі JavaScript нічим не відрізняється від будь-якої іншої мови програмування – функція `outTwo()` почне виконуватися після завершення функції `outOne()` [1].

Далі трохи змінимо нашу задачу, максимально наблизивши її до реальних умов основного застосування JavaScript. Наприклад, нехай відправляється запит до деякого Web-ресурсу для отримання

даних, які, потім, будуть оброблятися. Зрозуміло, що від моменту запити до моменту отримання даних пройде деякий час.

Для моделювання цієї ситуації в функції `outOne()` перед виведенням результату виконаємо деяку затримку, наприклад, на 200 мілісекунд. У мові програмування JavaScript є функція `setTimeout()`, яка викликає передану їй в якості першого параметра функцію із заданою затримкою (другий параметр). Тепер код буде виглядати так:

```
function outOne() {
  // Імітуємо затримку
  setTimeout(function() {
    console.log(1);
  }, 200);
}
function outTwo() {
  console.log(2);
}
outOne();
outTwo();
```

Основна ідея тут – виконання команди `console.log(1)`; було відкладено на 200 мілісекунд. В результаті при запуску цього коду ми отримаємо:

```
2
1
```

Хоча функція `outOne()` як і раніше викликається першою, її вивід з'явився другим, після виведення функції `outTwo()`. Але JavaScript не порушує порядок виклику функцій: він просто не чекає завершення функції `outOne()`, а відразу рухається далі – запускає на виконання функцію `outTwo()`. В JavaScript не можна просто запускати функції в потрібному порядку і сподіватися, що вони в будь-якому випадку будуть виконуватися в тому ж порядку.

Якщо згадати, що JavaScript був розроблений для «оживлення» Web-сторінок, то така його поведінка є повністю логічною і закономірною. Якби він був синхронним, то користувач не зміг би взаємодіяти зі сторінкою, поки не отримає результат виданого запиту.

Тому практично кожна корисна JavaScript-програма створюється з залученням асинхронних методів розробки. Асинхронний код,

в порівнянні з синхронним, зазвичай складніше писати, читати і підтримувати. Іноді він перетворюється в зовсім моторошні структури. Однак у реальній розробці без нього не обійтися!

Для розуміння асинхронної природи JavaScript необхідно усвідомити, що функції в ньому є фундаментом мови. Функції в JavaScript – це повноцінні об'єкти. Тобто, функції можуть приймати інші функції в якості аргументів, а також вони можуть повертатися в якості результату. Функції, які вмюють це робити, називаються функціями вищого порядку.

1. Флэнаган Д. JavaScript. Подробное руководство, 6-е издание. – Пер. с англ. – СПб: СимволПлюс, 2012. – 1080 с., ил.

БАЗИ ДАНИХ В ЛОГІСТИЦІ

Черножукова А.І.

Научний керівник – Штельма О.М.

Бази даних – інформаційні активи, багаті в обсязі, швидкості приросту і різноманітті, що включають в себе інноваційні та економічно ефективні методи обробки інформації, що сприяють поліпшенню процесу прийняття рішень і його оптимізації.

Технології БД мають широке застосування в логістиці. Аналітика БД надає інформацію про моделі поведінки покупців, ринкові тенденції, циклах технічного обслуговування. Вона також пропонує методи скорочення витрат, оптимальну цінову стратегію, стратегію з оптимізації процесів і значно полегшує процес прийняття рішень.

Сфери діяльності логістики, розвитку яких сприяє технологія БД:

- Підвищення ефективності. Самий прямолінійний спосіб застосування БД в бізнесі - підвищення рівня ефективності операцій. Кінцевий етап ланцюжка поставок часто виявляється найбільш витратним, тому логістичні системи загострюють увагу на цьому питанні.
- Покращення якості обслуговування клієнтів. Дані, отримані й оброблені за допомогою технологій БД, дозволяють зменшити втрати клієнтської бази і краще розуміти споживчий попит.
- Реалізація ефективної і нової бізнес-моделі. Типи і обсяги товарів, що доставляються в різні регіони, дозволяють визначити їх споживчий попит.

Для цих цілей служать системи управління ланцюгами поставок (SCM) в інтеграції з системами управління взаємовідносинами з клієнтами (CRM). У цих системах здійснюється управління такими даними, як транзакції клієнтів, інвентаризація, реклама, відносини з клієнтами,