ISSN 2304-120X
КОНЦЕПТ
научно-методический электронный журнал

Смольянов А. Г., Пантилейкин Н. В. Методические аспекты изучения парсинга средствами PHP в курсе «Сетевые языки и веб-программирование» // Научно-методический электронный журнал «Концепт». – 2018. – № V4. – 0,3 п. л. – URL: http://e-koncept.ru/2018/186035.htm.

ART 186035 **УДК 378.147:004.9**

Смольянов Андрей Григорьевич,
кандидат физико-математических наук, заведующий кафедрой фундаментальной информатики ФГБОУ ВО «Национальный исследовательский Мордовский государственный университет имени Н. П. Огарёва», г. Саранск
mgutech@mail.ru

Пантилейкин Никита Викторович,
магистрант ФГБОУ ВО «Национальный исследовательский Мордовский государственный университет имени Н. П. Огарёва», г. Саранск
niksar777@gmail.com

Методические аспекты изучения парсинга средствами PHP в курсе «Сетевые языки и веб-программирование»

Аннотация. *Новое поколение языков программирования привнесло в методику их преподавания новые парадигмы, идеи и понятия, без знания которых студент не сможет проявить на практике устойчивые навыки написания реальных компьютерных программ. Методика преподавания дисциплин, связанных с программированием, включает хорошую проработку дидактического материала, предполагающего изучение на лабораторных занятиях широкого спектра вопросов – от основ конкретного языка программирования до его специальных возможностей, имеющих большое значение в так называемом промышленном программировании. В настоящее время в сфере практического программирования активно используются серверные языки, такие как PHP, Ruby, Java, C, Python, Perl и другие. В статье рассматриваются методические аспекты, связанные с изучением специальных языковых средств PHP, позволяющих проиллюстрировать некоторые возможности парсинга и наглядно показать принцип действия этого механизма на практических примерах.*
Ключевые слова: *парсинг, парсер, скрипт, информер, контент.*
Раздел: *(01) отдельные вопросы сферы образования.*

За последние несколько лет инструментарий профессиональных программистов претерпел революционные изменения. Это связано с появлением и реализацией в программировании новых идей, возникновением новых парадигм и понятий. Огромное влияние на эти изменения оказало, в частности, развитие сети Интернет, которое поставило перед разработчиками программного обеспечения новые задачи и проблемы. В результате от преподавателей вузов потребовался быстрый и значительный пересмотр содержания целого ряда дисциплин компьютерного цикла и введение в учебные планы профильной подготовки студентов вузов новых дисциплин.

Большой интерес у студентов вызывают дисциплины, использующие средства веб-программирования. Именно они во многом формируют картину современного программирования. Особое место в списке языков программиста-профессионала занимают серверные языки программирования, такие как PHP, Ruby, Java, C, Python, Perl и другие [1]. Они нужны, в частности, для реализации слоя приложения, называемого бизнес-логикой.

В преподавании цикла компьютерных дисциплин могут быть рассмотрены различные аспекты конкретных серверных языков программирования. В частности, в рамках дисциплины «Сетевые языки и веб-программирование» может быть рассмотрена тема «Парсинг ресурсов сети», которая сегодня весьма актуальна и популярна с точки зрения

подготовки специалиста в области IT. Эта тема требует понимания взаимодействия различных программных механизмов, объектов и ресурсов сети. Очень важным аспектом этой темы является изучение той части языкового инструментария, которая необходима для решения задач парсинга. Одна из интересных и практически значимых возможностей таких языков – автоматизированный сбор информации с какого-либо источника с целью его дальнейшей обработки и преобразования для своих собственных потребностей. Такая технология называется парсингом [2]. Программа, которая используется для анализа и обработки данных, называется парсером. Готовые данные, как правило, выкладываются в базу данных, представляются в виде обычного файла или файла в формате XML. Примером парсинга может быть обработка сайта интернет-магазина, результатом которой является список товаров, представленных по категориям. Парсингом могут заниматься поисковые роботы, анализируя страницы и сохраняя полученные данные о них в собственной базе. Эти данные затем используются поисковой системой для ранжирования проанализированных сайтов и формирования выдачи. Парсинг лежит в основе многочисленных сервисов для различных специалистов (например, маркетологов), позволяющих анализировать сайты из поисковой выдачи. В некоторых случаях целью парсинга является не получение каких-то данных из обработанного контента, а сам контент, представленный в нужной заказчику форме.

Скрипты, которые выполняют подобные задачи, называются парсерами. Как правило, они пишутся для получения данных с какого-то конкретного, регулярно обновляемого источника. Целями парсинга могут быть ссылки на разные страницы какого-либо сайта, изображения, видео, текст и другой контент.

При изучении данной темы на лабораторном занятии требуется обсудить с обучающимися различные программные механизмы, языковые средства и объекты, позволяющие получить доступ к ресурсам сети с целью их обработки. Результативность изучения материала будет значительно повышена благодаря практической иллюстрации возможностей парсинга на основе PHP.

Для иллюстрации таких возможностей рассмотрим парсинг RSS ленты с помощью SimpleXML. RSS – семейство XML-форматов, созданных для описания новостных лент, заметок, статей, изменений в блогах и т. п. Информация из разных источников, представленная в формате RSS, может быть подобрана, подвергнута обработке и представлена в некотором удобном виде специальными программами-агрегаторами либо интернет-сервисами типа Feedly, Inoreader, BlinkFeed. Расширение SimpleXML предоставляет очень простой и легкий в использовании набор инструментов для преобразования XML в объект, с которым можно далее работать через его свойства и с помощью итераторов. В свою очередь, язык XML (e**X**tensible **M**arkup **L**anguage) – это расширяемый язык, рекомендованный Консорциумом Всемирной паутины (W3C) как язык разметки. Он является подмножеством языка SGML. SimpleXML – это расширение для PHP5, встроенное в него по умолчанию. Оно представляет самый простой способ обработки XML-файлов (и файлов RSS, в частности).

Первый шаг нашей работы – анализ XML-документа и его сохранение в переменной. Это требует написания всего лишь одной строки кода, которая передает URL функции **simplexml_load_file()**:

$rss = simplexml_load_file
('http://news.yandex.ru/hardware.rss');

Функция **simplexml_load_file()** интерпретирует XML-файл в объект.

Второй шаг – нахождение имени канала.

Имя всего канала находится в дочернем элементе **title** от элемента **channel**, порожденного корневым элементом **rss**. Этот заголовок можно загрузить так, как будто

ISSN 2304-120X
КОНЦЕПТ
научно-методический электронный журнал
Смольянов А. Г., Пантилейкин Н. В. Методические аспекты изучения парсинга средствами PHP в курсе «Сетевые языки и веб-программирование» // Научно-методический электронный журнал «Концепт». – 2018. – № V4. – 0,3 п. л. – URL: http://e-koncept.ru/2018/186035.htm.

XML-документ был просто последовательной формой объекта класса **rss** с полем **channel**, которое, в свою очередь, имело бы поле **title**. Используя регулярный PHP-синтаксис ссылки на объект, запишем команду нахождения заголовка:

$title = $rss->channel->title;

Далее найдем элементы канала. Выражение, выполняющее эту задачу, записывается также просто:

$rss->channel->item;

Однако каналы могут содержать более одного элемента, либо их может не быть вовсе. Следующая команда возвращает массив, который можно обработать с помощью цикла **for-each**:

```
foreach ($rss->channel->item as $item)
{ echo "<h2>". $item->title. "</h2>";
echo "<p>". $item->description. "</p>"; }
```

Это все, что нужно для написания простой программы чтения RSS средствами PHP. Пример выполнения скрипта показан на рис. 1.

Рис. 1. Результаты парсинга

Для улучшения и расширения возможностей данного скрипта можно вывести дополнительные информационные элементы, например дату новости, добавить некоторые свойства CSS, оформить информацию в табличном виде. Результат дополнительной обработки показан на рис. 2.

□	**HTC Hima Ace и Hima Ultra получат процессор MediaTek**	HTC уже не первый год экспериментирует с различными процессорами в модификациях одного и того же устройства — из последнего можно вспомнить Desire 620 и Desire 620G. А вот удешевленный Hima Ace независимо от модели получит чип от MediaTek, как понятно из сообщения UpLeaks.	17 Dec 2014 16:52:00 +0300
	Индийская компания	Сегодня в столице Индии компания Micromax представила	18 Dec

Рис. 2. Дополнительное оформление результатов парсинга

Еще один пример парсинга, который способствует повышению интереса и мотивации студентов при изучении данной темы, – создание собственного информера погоды.

Информеров погоды много, но у них есть недостаток: такие сервисы в случае необходимости не позволяют изменить внешний вид информера. Это приходится делать в случае необходимости программисту самостоятельно.

В качестве примера воспользуемся сервисами Яндекс и Yahoo.

Сервис Яндекс предоставляет прогноз в формате XML. Данные, полученные из этого документа, представляют собой подробную сводку по всем метеопараметрам, которые есть на сайте https://pogoda.yandex.ru.

Наш информер будет состоять из двух блоков: первый будет содержать краткую информацию о погоде на сегодняшний день, второй – более подробную, для различных времен суток (утро, день, вечер, ночь).

Перейдем к непосредственному парсингу XML-документа. Для этого воспользуемся специальным расширением **SimpleXML,** которое появилось в PHP версии 5.0.

С помощью функции **simplexml_load_file()** конвертируем файл XML в объект класса **SimpleXMLElement**.

```
$xml = simplexml_load_file
        ("http://export.yandex.ru/weather-ng/
        forecasts/27760.xml");
```

В нашем примере http://export.yandex.ru/weather-ng/forecasts/27760.xml – адрес XML файла, где 27760 – id города, в данном случае Саранска.

Теперь выбираем необходимые параметры для нашего информера (город, температура, пиктограмма, тип погоды текстом, влажность, давление, направление и скорость ветра).

```
$city = $xml->attributes()->city;
$temp = $xml->fact->temperature;
$img = 'image-v3';
$pic = $xml->fact->$img;
$w_type = $xml->fact->weather_type;
$wind_direction = $xml->fact->wind_direction;
$wind_speed = $xml->fact->wind_speed;
$humidity = $xml->fact->humidity;
$pressure = $xml->fact->pressure;
```

Выводим все это на экран и добавляем CSS. Результат показан на рис. 3.

Рис. 3

Далее создаем второй блок с более подробной информацией. Файл погоды содержит прогноз на 10 дней, но для нашего примера возьмем информацию только за один день. Вывод будем осуществлять с помощью многомерного массива **$OutWeather**.

```
$OutWeather = array();
foreach ($xml->day as $day ) {
for ($i = 0; $i <= 3; $i++) {
                                    if($day->day_part[$i]->temperature == '')
{
                                    $temp_from = $day->
day_part[$i]->temperature_from;
$temp_to = $day->day_part[$i]->
temperature_to;
                                     }
else
{
                                     $temp_from = (int)$get_temp – 1;

                                     $temp_to = (int)$get_temp + 1;
                                                                          }
                                                                          if ($temp_from > 0)
{ $temp_from = '+'.$temp_from; }
                                                                          if ($temp_to > 0)
{ $temp_to ='+'.$temp_to; }
                                     $OutWeather[0]['weather'][$i]['temp_from'] =
$temp_from;
$OutWeather[0]['weather'][$i]['temp_to'] =
$temp_to;

                                    $OutWeather[0]['weather'][$i]['image'] =
$day->day_part[$i]->{'image-v3'};

                                    $OutWeather[0]['weather'][$i]['time_day'] =
$TimeDay[$i];

                                    $OutWeather[0]['weather'][$i]['wind_speed'] =
$day->day_part[$i]->wind_speed;

                                    $OutWeather[0]['weather'][$i]['wind_direction']=
$day->day_part[$i]->wind_direction;

                                    $OutWeather[0]['weather'][$i]['humidity'] =
$day->day_part[$i]->humidity;

                                    $OutWeather[0]['weather'][$i]['pressure'] =
$day->day_part[$i]->pressure;

                                    $OutWeather[0]['weather'][$i]['weather_type'] =
$day->day_part[$i]->weather_type;
                                     }
                                    }
```

Результат работы данного кода показан на рис. 4–5.

ISSN 2304-120X
КОНЦЕПТ
научно-методический электронный журнал

Рис. 4

Рис. 5

Yahoo! Weather предоставляет информацию в формате RSS на ближайшие пять дней.

Опять с помощью функции **simplexml_load_file()** конвертируем файл XML в объект класса **SimpleXMLElement**.

```
$xml =
simplexml_load_file("http://xml.weather.yahoo.com/
forecastrss?p=RSXX1460&u=c");
```

RSXX1460& – код города, c – параметр, который соответствует градусам Цельсия.

Для парсинга будем использовать XPath.

XPath (XML Path Language) – язык запросов к элементам XML-документа. Язык разработан для организации доступа к частям документа XML в файлах трансформации XSLT и является стандартом консорциума W3C. XPath призван реализовать навигацию по DOM в XML.

```
//Ветер
                    $wind = $xml->xpath("/rss/channel/yweather:wind");
                    $wind = $wind[0];
                    $wind_direction = (int)$wind['direction'];
                    $wind_speed = (int)$wind['speed'];
//Атмосферные показатели
                    $atmosphere = $xml->xpath
```

```
("/rss/channel/yweather:atmosphere");
                    $atmosphere = $atmosphere[0];
                    $humidity = (int)$atmosphere['humidity'];
                    $visibility = (int)$atmosphere['visibility'];
                    $pressure = (int)$atmosphere['pressure'];
// Время восхода и заката
                    $astronomy = $xml->xpath
('/rss/channel/yweather:astronomy');
$astronomy = $astronomy[0];
$sunrise = $astronomy['sunrise'];
                    $sunset = $astronomy['sunset'];
//Текущая температура воздуха и погода
                    $condition = $xml->xpath
('/rss/channel/item/yweather:condition');
                    $condition = $condition[0];
                    $temperature = (int)$condition['temp'];
$condition_text = (string)$condition['text'];
$condition_code = (int)$condition['code'];
```

Прогноз погоды на следующие дни:

```
$forecast = array();
$weatherFiveDays = $xml->xpath
('/rss/channel/item/yweather:forecast');
foreach($weatherFiveDays as $day) {
$forecast[] = array(
'date' => strtotime((string)$day['date']),
'low' => (int)$day['low'],
'high' => (int)$day['high'],
'text' => (string)$day['text'],
'code' => (int)$day['code']);
}
```

Давление можно перевести в миллиметры ртутного столба. Для этого **$pressure** умножим на 0,75006375541921. Результат скрипта показан на рис. 6.

Саранск: -3 °C,переменная облачность
ветер: 1 м/с
влажность: 74%,
давление: 768 мм.рт.ст.
восход: 06:57, закат: 16:30

29.10, чт: от -4 до 0°, переменная облачность

30.10, пт: от -5 до 2°, небольшой снег

31.10, сб: от -4 до 1°, переменная облачность

01.11, вс: от -1 до 2°, облачно

02.11, пн: от 1 до 4°, облачно

Рис. 6

ISSN 2304-120X
КОНЦЕПТ
научно-методический электронный журнал

Смольянов А. Г., Пантилейкин Н. В. Методические аспекты изучения парсинга средствами PHP в курсе «Сетевые языки и веб-программирование» // Научно-методический электронный журнал «Концепт». – 2018. – № V4. – 0,3 п. л. – URL: http://e-koncept.ru/2018/186035.htm.

Приведенные в примерах скрипты можно модифицировать, изменить стили, вывод непосредственно для своего сайта. Чтобы не нагружать сайты частым парсингом, можно сделать кэширование данных и обновлять их несколько раз в день.

Рассмотренные языковые инструменты и примеры способствуют более качественному усвоению студентами заявленной темы в образовательном курсе, а также дают возможность не только понять идеи механизма парсинга, но и использовать их в решении образовательных и практических жизненных задач.

Ссылки на источники

1. Веб Креатор. – URL: https://web-creator.ru/articles/server_side_languages.
2. Сетевое издание «Интернет-сайт www.seonews.ru». – URL: https://www.seonews.ru/glossary/parsing.

Andrei Smolyanov,
Candidate of Physical and Mathematical Sciences, Head of Fundamental Informatics Chair, National Research Mordovia State University named after N. P. Ogarev, Saransk
mgutech@mail.ru
Nikita Pantileykin,
Graduate Student, National Research Mordovia State University named after N. P. Ogarev, Saransk
niksar777@gmail.com

Methodological aspects of studying parsing by means of PHP in the course “Network Languages and Web Programming”

Abstract. The new generation of programming languages introduced new paradigms, ideas and concepts into their teaching methods, without knowledge of which the student would not be able to demonstrate in practice sustainable skills in writing real computer programs. The methodology of teaching disciplines related to programming includes comprehensive study of didactic material, which involves studying a wide range of issues in laboratory classes – from the basics of a particular programming language to its special features that are of great importance in so-called industrial programming. Currently in the field of practical programming, server languages are actively used, such as PHP, Ruby, Java, C, Python, Perl and others. The article considers methodological aspects related to the study of special PHP language tools, which allow to illustrate some parsing possibilities and to demonstrate the principle of this mechanism operation on practical examples.

Key words: parsing, parser, script, informer, content.

References

1. Veb Kreator. Available at: https://web-creator.ru/articles/server_side_languages (in Russian).
2. Setevoe izdanie “Internet-sajt www.seonews.ru”. Available at: https://www.seonews.ru/ glossary/parsing (in Russian).

Рекомендовано к публикации:
Горевым П. М., кандидатом педагогических наук,
главным редактором журнала «Концепт»

Поступила в редакцию *Received*	02.03.18	Получена положительная рецензия *Received a positive review*	20.03.18
Принята к публикации *Accepted for publication*	20.03.18	Опубликована *Published*	30.04.18

ISSN 2304-120X
www.e-koncept.ru