

인터넷언어와 공학계산에의 응용

백 광 연*

(단국대학교 공학부 기계공학과)

1. 머리말

최근의 컴퓨터 및 기타 통신망의 급속한 팽창 과 고속화 및 발전으로 인터넷이 없는 일상 생활은 상상할 수 도 없을 만큼 우리 와 밀접하게 연결되어 있다. 특히 타 국가에 비하여 초고속 통신망이 일반가정에까지 광 범위하게 연결되어 있는 우리나라에서는 인 터넷의 사용자가 전문가로부터 주부에 이르 기까지 그 사용자 층과 사용범위가 매우 광 범위하며 그 적용분야가 나날이 증가하고 있다. 이는 1969년 미국방성의 ARPANET 프로젝트를 시발점으로 하여 USENET, BITNET 등으로 이어지는 과거 인터넷의 전통적인 학술 및 군사적 사용목적에서 그 사용범위가 크게 확대되어 요즈음은 오히려 상업적 목적의 사용이 훨씬 더 활발하고 다 양한 양상을 띠고 있는 것과 무관하지 않다.

이러한 인터넷의 활용은 과거부터 많이 사용된 단순한 자료의 저장 및 교환(FTP) 과 Telnet을 이용한 원격지 중앙 컴퓨터(주 로 UNIX시스템)에 로그인을 통한 중앙 컴

퓨터의 프로그램의 수행으로부터 시작하여 wais와 gopher를 거쳐 WWW(world wide web)이 사용되면서 인터넷의 급속한 발전을 통하여 오늘날과 같이 우리 생활의 일부로 자리잡게 되었다.

간단히 웹이라고도 불리는 이 월드 와이드 웹(WWW)은 하이퍼텍스트(hyper text)를 기반으로 하여 문서활용에 많은 편리성을 가지면서 동시에 URL(uniform resource location)의 지정에 따라 HTTP라는 웹 프 로토콜에 의해 FTP(file transfer protocol), NNTP(network news transfer protocol), telnet 등의 기존 프로토콜을 지원하며 gateway를 통해서는 archie, wais, gopher, whois 등의 서비스를 제공한다. 웹은 HTML(hyper text markup language)이라 는 문서기술을 위주로 한 일종의 컴퓨터언 어를 이용하여 작성한 홈페이지들이 인터넷 을 통하여 서로 유기적으로 연결된 상태를 의미한다. 이러한 홈페이지들은 이 문서가 존재하는 서버에 접속한 클라이언트가 웹 브라우저를 통하여 볼 수 있게 되는 것이다. 이 HTML은 문서 이외에도 이미지나 동영상, 소리 등의 실행을 다른 도구 없이 웹 브 라우저안에서 지원해 준다. 초기의 이러한

* 정회원

E-mail : khbaek@dku.edu

기능에서 출발하여 HTML에 기반을 둔 많은 언어들(예: PHP, Perl, Python 등)이 인터넷을 통한 웹의 활용을 더욱 더 다양하고 충실하게 만들어 주고 있다. 본 원고에서는 이러한 언어들(예: PHP, Perl, Python 등)의 특징과 장단점에 대하여 알아보고 어떤 방식으로 공학에 응용될 수 있는지 간략한 예를 들어 설명하고자 한다.

2. 인터넷 프로그래밍을 위한 각종 언어들과 특징

요즈음은 각 개인들이 홈페이지를 가지게 될 정도로 HTML언어에 대한 기본적인 지식이 잘 알려져 있고 HTML언어를 이용한 문서 작성을 도와주는 도구나 응용프로그램들이 많으므로 HTML언어에 대한 설명은 생략하고자 한다. 여기서는 HTML언어를 기반으로 한 여러 가지 형태의 프로그램 언어와 그들의 특징에 대해 설명하고자 한다.

2.1 CGI

CGI(Common Gateway Interface)란 브라우저와 서버 그리고 응용프로그램간의 인터페이스를 의미한다. 즉, 사용자의 브라우저로부터 서버로 전달된 데이터를 서버의 응용프로그램이 가공하고 이 결과를 다시 사용자의 브라우저로 전달하는 인터페이스를 말하는 것이다. 클라이언트쪽에서 GET이나 POST등의 명령으로 서버쪽으로 데이터를 보내면(반드시 데이터가 필요한 것은 아님) 서버는 이 데이터를 기반으로 새로운 HTML문서를 클라이언트에 전달하게 된다. 따라서 CGI의 기본적인 특성상 서버 쪽에 부하가 많이 걸리게 된다. 웹에서 흔히 보게

되는 메시지인 “당신은 몇 번째 방문자입니다”라고 하는 카운터라는 기능은 바로 이 CGI를 이용한 초보적인 응용 예이다. 이것을 좀 더 구체적인 과정으로 나누어 본면 다음과 같다.

(1) 브라우저가 URL의 도메인 이름을 해석하여 서버와 연결을 설정.

(2) 브라우저가 URL을 제외한 나머지 정보의 부분을 서버에 전달.

(3) 서버에서 브라우저로부터 전달된 내용을 해석하여 서버내의 경로와 파일명으로 번역.

(4) 파일명이 HTML문서가 아닌 CGI프로그램을 가리키고 있음을 확인.

(5) 환경을 준비하고 스크립트(CGI 프로그램)를 실행.

(6) 스크립트가 실행결과를 표준출력에 출력 후 종료.

(7) 클라이언트의 브라우저에 서버로부터의 출력결과가 보여짐.

카운터의 경우, 새로운 클라이언트가 서버에 접속하면 서버의 데이터베이스 등에 기록되어 있던 홈페이지의 총 접속자 수에 1을 더한 수를 앞서의 (5)와 (6)의 과정을 통하여 클라이언트 쪽에 보내게 되는 것이다. 이 과정에서 CGI의 구현은 여러 가지 방법으로 가능한데 크게 인터프리터 언어, 컴파일 언어, 컴파일 되고 인터프리터 되는 언어에 의한 세가지 방법으로 구분할 수 있다.

인터프리터 언어란 실행 가능한 기계어로 된 파일이 아닌 ASCII 코드의 텍스트 파일을 서버에 존재하는 인터프리터에서 명령 하나 하나를 순차적으로 시행해 나가는 언어이다. UNIX셸(shell) 스크립트와 Perl

(practical extraction and report language) 이 대표적인 예이다. UNIX의 셸 스크립트는 DOS의 기본 명령어와 같이 시스템 명령어 프로그램과 같은 것이다. 사용이 쉽고 이식성이 뛰어나다는 장점이 있는 반면 보안상 위험이 따른다는 단점이 있다. Perl은 UNIX를 비롯한 여러 가지 운영체제에서 사용될 수 있으며 C나 UNIX셸 프로그램이 가지고 있는 문제점들을 보완하여 여러 가지 데이터 포맷이나 복잡한 데이터 구조를 쉽게 다룰 수 있는 등 유용한 유틸리티 언어이다.

두 번째의 컴파일 언어에는 C/C++, Visual Basic등이 있으며 컴파일 된 실행 파일로 CGI명령을 수행하게 된다. 따라서 실행파일의 크기가 작고 수행속도가 빠르며 보안성이 나은 장점을 가진다. 반면에 운용될 모든 플랫폼과 운영체제에 대한 실행 파일을 독립적으로 만들어야 하므로 이식성이 떨어지고 불편하게 된다.

세 번째의 컴파일 되고 인터프리터 되는 대표적 언어에는 Sun Microsystem사의 JABAscript와 Microsoft사의 Visual Basic script가 있다. JABAscript는 표준 아스키 텍스트 스크립트 형태로 직접 HTML문서 내에 프로그램되어 웹 브라우저에 의해 직접 실행된다. 즉 클라이언트의 브라우저에서 실행되므로 진정한 의미의 CGI언어라고 할 수는 없으나 클라이언트에서 데이터를 미리 처리하게 되므로 서버의 부담을 덜어주게 된다. VB스크립트는 JABAscript와 매우 유사한 특성을 지니며 주로 비디오나 사운드 파일을 처리해 주는 기능을 수행한다. 브라우저는 웹 서버로부터 VB스크립트를 받는 동안 코드를 바로 해석, 컴파일 하여 실행된다.

행된다.

JABAscript의 기본 형식은 다음과 같은 두 가지가 있다.

```
(1) <Script language = "JavaScript">
    <!--
        JavaScript 내용...
    // -->
(2) <Script Src = "filename.js">
    ...
</Script>
```

첫 번째의 예는 브라우저가 JABAscript를 지원하지 못하는 경우에는 <!-- ...// -->로 표시되는 부분이 HTML의 주석문으로 인식되어 오류가 생기지 않게 되는 것을 보여준다. 두 번째의 예는 같은 JABAscript 내용을 "filename.js"라는 저장된 파일로부터 실행하게 되는 것을 보여주며 이 경우에는 소스코드가 브라우저의 소스보기로 보여지지 않는다. 다음의 예는 이름을 물어보고 "name"이라는 변수에 저장하여 화면에 바로 출력하는 예이다.

```
<HTML>
<BODY>
<SCRIPT LANGUAGE=JavaScript>
<!--
    var name = prompt("이름은?", "name");
    document.write("<H3>안녕하세요, "+ name + "
    씨!</H3>");
// -->
</SCRIPT>
</BODY>
</HTML>
```

자바스크립트의 주요 특징은, 상속이나 클래스는 지원하지 않으나 Built-in object를 지원하며 앞의 예에서 보듯이 HTML과 함께 사용하며 편집이 쉽고 CGI의 폼 형식과 결합시켜 사용할 수 있으며 웹 브라우저가 사용하는 속성을 이용하게 하여 비교적 편리하고 간단히 프로그래밍이 가능하다는 것이다.

2.2 ASP

ASP(active server page)는 CGI의 서버에서 수행되는 성질과 JAVAscript의 성질을 동시에 가진 것이라고 보면 된다. ASP는 데이터베이스와 연동되는 프로그램의 작성이 쉽고 객체 기반의 프로그램이 되며, 동시 접속 시 스레드로 처리되어 서버의 부하가 적다는 특징을 가진다.

ASP파일의 기본적 구성은 HTML 태그 사이에 <% ... %>태그를 사용하여 서버에서 사용되는 스크립트를 기록한다. ASP에서 사용하는 스크립트 언어에는 VBscript와 JAVAscript 두 가지 있으며 ASP파일의 제일 위에 <%@ language = "ScriptType" %>라고 시작하며 저장되는 파일의 확장자는 ".htm"이 아닌 ".asp"가 된다. 물론 이 ".asp" 확장자를 가지는 파일은 클라이언트의 브라우저에서 바로 실행되는 것이 아니고 서버의 해석을 통해 클라이언트의 브라우저에서 보여지는 것이다.

ASP의 경우 주로 객체를 이용하여 프로그램을 하는데 브라우저 쪽의 window, document, frame, form, element, link, location객체와 서버 쪽의 server, request, response, application. Session객체가 있다. 다음은 방문객의 수를 세는 VBScript를 이용한 카운터 예제이다.

```
<%@ language="VBScript" %>
<% Response.Expires = 0 %>
<html>
<body>
<h3> 카운터 예제</h3>
<hr>
<%
    if (Application("Count") = "") then
        Application("Count") = 0
    end if
    Application.Lock
    Application("Count")=Application("Count") + 1
    Application.Unlock
%>
<BR>
<%= Application("Count") %>번째 방문자.<br>
</body>
</html>
```

위에서 "Response.Expires = 0"는 이 문서가 캐시에 저장되지 않도록 하여 방문객 수를 정확히 세도록 하는 것이다. 또, "Application.Lock"은 동시에 둘 이상의 클라이언트의 접속을 막아 방문객 수가 제대로 세어지도록 만드는 것이다.

2.3 JSP

JSP(java server page)는 Microsoft사의 ASP처럼 JAVA로 구현된 서버쪽 스크립트 언어이다. JSP는 HTML, DHTML, XML과 같은 동적 콘텐츠를 처리하는 자바 플랫폼 기술이다. HTML코드에 자바 코드가 포함되어 JSP 탐지 서버가 이를 컴파일하고 스스로 해당 서블릿을 작동/삭제시킨다.

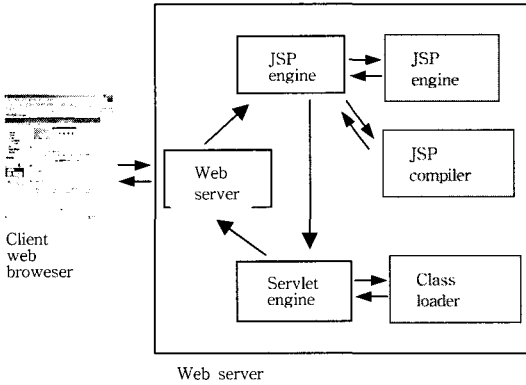


그림 1 JSP의 동작 원리

CGI의 경우 발생하는 사용자 수의 증가에 따라 성능이 감소되는 현상이 상당 부분 없�지며 JAVA를 기반으로 만들어져 이의 장점을 그대로 물려받는 장점이 있다. 이는 자바의 가장 큰 장점인 플랫폼의 독립성이 그대로 적용되어 모든 웹 서버, 웹 어플리케이션 서버에서 실행시킬 수 있으며 어떠한 웹 브라우저에서도 접근이 가능하다는 장점이 있다.

그림 1은 JSP의 작동되는 과정을 그림으로 나타낸 것이다.

클라이언트 쪽의 브라우저가 웹 서버에 JSP 파일을 요청하면 웹 서버는 이를 인지하고 JSP엔진에 요청을 넘긴다. 이 JSP 파일은 엔진에 의해 실행된 적이 있는가와 변경되었는지를 체크하고 실행된 적이 없거나 변경되었을 경우 워크 디렉터리에 자바파일(.java)을 만들고 컴파일을 실행시킨다. 이때 해당 클래스가 데이터베이스를 필요할 경우 DB로부터 데이터를 불러와 JSP파일에 붙여 넣는다. 서블릿 엔진은 JSP 엔진에서 넘어온 객체를 대개 HTML 코드로 바꾸어주게 되고, 웹 서버는 이 HTML 코드

를 클라이언트 쪽에 뿌려주게 되는 것이다.

JSP는 JAVA를 기반으로 하므로 시스템에 JVM(java virtual machine)이 구축되어 있어야 한다. 근래 웹의 필수적인 요소인 데이터베이스 연동을 JAVA는 쉽고 강력한 데이터베이스 조작 인터페이스인 JDBC를 통하여 제공하는데 Windows에서 제공하는 ODBC와 비슷한 구조를 가지고 있다.

2.4 PHP

PHP(personal homepage tools)는 HTML에 내장되어 동작하는 스크립트 언어(HTML-embedded scripting language)이다. 사용자가 HTML폼을 통하여 입력된 값을 웹 서버에서 처리한 후 그 결과를 브라우저에 전달하는ASP처럼 프로그래밍이 가능한 언어이다. PHP에는 JAVA, Perl, C등과 유사한 문장형식이 많아 이에 대한 기본 지식이 있는 사람에게는 웹 문서 작성이 비교적 쉽고 빠르다. 그림 2는 PHP의 동작 개념을 간략히 나타낸 것이다. 클라이언트가 웹 서버에 접속하여 PHP로 작성된 스크립트 문서를 요청하면 웹 서버는 요청받은

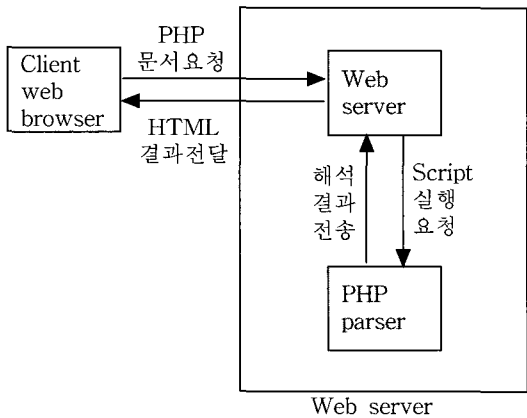


그림 2 PHP의 동작 원리

PHP 스크립트 문서를 확인 후 PHP parser 로 보낸다. parser는 이를 해석하여 그 결과를 웹 서버를 통하여 가공된 HTML 문서의 형태로 클라이언트에 보내는 것이다.

이러한 PHP의 특징으로는 우선 데이터베이스의 쉬운 연결을 들 수 있다. PHP에서 지원하는 데이터베이스에는 MySQL, dBase, ODBC를 포함한 거의 모든 데이터베이스를 지원한다. 또한 PHP는 UNIX 운영체제를 비롯하여 윈도우 운영체제하의 IIS(internet information server)나 PWS(personal web server), 윈도우용 Apache 웹 서버에서도 돌아간다. C나 Perl로 작성하던 CGI 프로그램을 똑같이 구현할 수 있다. PHP 코드는 '<?'로 시작하여 '?>'로 끝이 난다. 통상적으로 PHP에는 MySQL을 이용한 데이터베이스 연동 시스템이 많이 이용된다.

2.5 JAVA

JAVA는 1991년 Sun Microsystems사에 의해 발표되어 지금까지도 그 사용이 더욱 증가되고 있는 언어이다. 자바언어는 다양한 플랫폼과 다양한 CPU를 지원하여 널리 사용되고 있다. 자바언어의 독립성과 이의 밀바탕이 되고 있다. 이를 좀더 자세히 살펴보자. 인터넷에 연결된 다양한 운영체제와 CPU를 가지는 컴퓨터에서 모두 실행되는 언어가 되기 위하여 자바는 플랫폼에 독립적인 중간 코드인 바이트코드(bytecode)를 생성하여 이를 만족시킨다. 이 코드는 각각의 컴퓨터에서 실행되는 기계어 코드가 아니며 자바가상기계(java virtual machine)라는 인터프리터를 통하여 실행이 된다. 이는 그림 3을 참조하면 명확히 이해가 될 것이다.

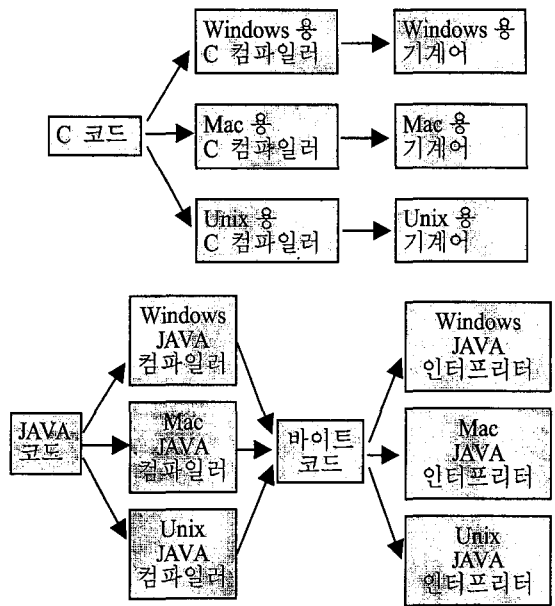


그림 3 C와 JAVA코드 구현의 비교

자바는 기본적으로 C언어의 문법을 활용하고 있으며 C++의 객체지향기능을 가지고 있다. 즉 상속, 자료의 추상화와 동적 바인딩 등이 클래스와 다형성을 통하여 구현되고 있다. 또한 TCP/IP 프로토콜을 처리하므로 분산 환경에 유리하고, 멀티 스레드 기능을 언어 자체에서 지원하여 대화형 인터넷 프로그램의 개발에도 유리하다. C++언어의 객체지향을 채택하면서도 포인터나 연산자 중복 등의 복잡한 기능을 제거하고 동적으로 할당되는 메모리를 가비지컬렉터를 통하여 자동으로 관리하여 사용이 간편한 작고 간단한 언어이다. 반면에 프로그램 실행 시 실행형태에 관한 많은 정보를 동반하므로 실행속도가 떨어질 수 있으나 프로그램의 작성에는 융통성이 생긴다.

자바에는 어플리케이션과 애플릿의 두 가

지 형태의 응용프로그램이 있다. 자바 어플리케이션은 C나 다른 언어와 마찬가지로 작동하는데 다음의 예제 소스를 보자.

```
class Example {
    public static void main(String args) {
        System.out.println("Java Application 예제");
    }
}
```

위의 소스를 example.java로 저장 후 javac 컴파일러로 컴파일하면 (예, "javac Example.java") example.class라는 자바 바이트코드로 된 파일이 생성된다. 이 파일의 클래스 example을 실행시키려면 "java Example" 하면 PC의 경우 도스창의 명령 프롬프트 다음에 "Java Application 예제" 라는 글을 출력할 것이다. C언어 등의 컴파일 및 실행과정과 매우 유사하다.

반면에 자바 애플릿은 인터넷을 통하여 자바를 실행할 수 있는 웹 브라우저에 의해 실행되는 프로그램이다. 애플릿 소스를 컴파일한 후 웹 브라우저나 JDK(java developer kit)의 Aappletviewer를 이용하여 HTML문서를 통하여 실행되게 된다.

3. 인터넷을 이용한 공학계산 예

지금까지 인터넷을 통한 정보와 자료의 전달 및 공유 등 웹 프로그래밍에 필요한, HTML언어를 기반으로 그 기능을 한 차원 높은 수준으로 활용하게 해 줄 수 있는 여러 가지 언어들의 기본적인 특징에 대하여 알아보았다. 여기서는 이러한 언어들이 어떻게 공학에 이용될 수 있는지 아주 간단한 예를 들어 설명하였다. 인터넷 프로그래밍

언어에 익숙하지 못한 분들에게는 개괄적인 개념의 정립을 돕고 추후 활용방안에 대한 아이디어를 세우는데 도움이 될 것이다.

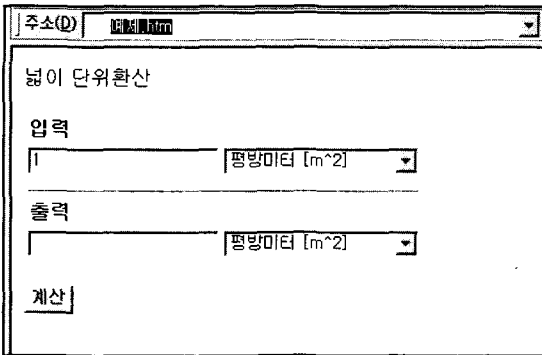
3.1 JAVAscript를 이용한 단위 환산

다음의 JAVAscript는 m²와 cm² 단위의 단면적을 cm²와 m² 단위로 바꾸어주는 소스코드이다. 자바스크립트를 전혀 모르는 사람이라도 HTML에 대한 기초 지식이 있는

```
<script language = "JavaScript">
<!--
function convert() {
    var UF = document.ConvertForm.UnitFrom.value
    var UT = document.ConvertForm.UnitTo.value
    var input= document.ConvertForm.ValueFrom.value
    document.ConvertForm.ValueTo.value =input*UF/UT
}
//--> </script></head>
<form name="ConvertForm" method="get">
<p>넓이 단위환산</p>
<table border="0">
<tr><td colspan="2">
<p><strong>입력</strong></td></tr>
<tr><td><p><input type="text" name="ValueFrom" value="1" onkeyup="convert()"></td>
<td><p>
<select name="UnitFrom" onchange="convert()">
<option value="0.0001">평방센티미터 [cm^2]</option>
<option selected value="1">평방미터 [m^2]</option></select></td></tr>
<tr><td colspan="2"><p><hr><strong>출력</strong></td></tr>
<tr><td><p><input type="text" name="ValueTo" onkeyup="convert()"></td>
<td><p><select name="UnitTo" onchange="convert()">
<option value="0.0001">평방센티미터 [cm^2]</option>
<option selected value="1">평방미터 [m^2]</option></select></td>
</tr> </table>
<br>
<input type="button" name="AreaConvert" value="계산" onclick="convert()">
</form>
```

분이라면 전반적인 구조가 어떻게 짜여져 있는지 대체적인 윤곽은 쉽게 이해되리라 생각된다.

위의 코드를 적당한 파일로 저장(예제.htm)한 후 브라우저에서 실행시키면 다음과 같은 창이 뜨며 쉽게 그 기능을 확인할 수 있을 것이다.



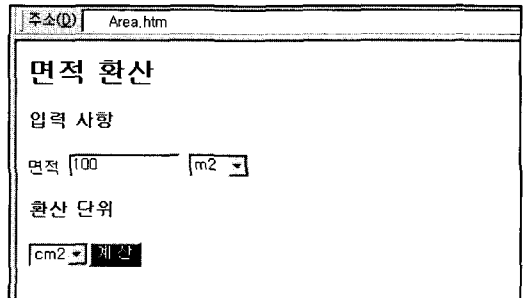
프로그램 소스의 상당 부분은 폼을 형성하는데 할애되었고 실제 계산 관련 부분은 "input*UF/UT"이 전부이다. 소음/진동분야를 포함한 공학 계산이나 기타 응용을 위해 필요한 수학 함수 등은 다른 언어들이 제공하는 수준과 거의 대등한 정도로 내장되어 지원되므로 예제에 보인 것과 같은 단순한 기능을 뛰어넘는 매우 다양한 프로그램의 작성이 가능하다.

3.2 PHP를 이용한 단위 환산

다음의 HTML소스와 PHP 소스는 앞서의 자바스크립트를 이용한 면적의 단위 환산과 동일한 기능을 하는 예제 코드이다.

```
<html>
<body bgcolor="white" text="black" link="blue"
vlink="purple" alink="red">
<form name="form1" method="post"
action="area.php3" target="output" >
<h2>면적 환산</h2>
<h4><font color="green">입력 사항</font></h4>
<p><TT>면적 <input type="text" name="s" size="12">
<select name="sUnit">
<option selected value="1">m<sup>2</sup></option>
<option
value="0.01">cm<sup>2</sup></option></select></TT></p>
</tr>
<h4><font color="green">환산 단위</font></h4>
<font color="black"><span
style="font-size:10pt;"></span></font><select
name="lUnit">
<option value="0">m<sup>2</sup></option>
<option selected value="1">cm<sup>2</sup></option>
<p><input type="submit" name="cal" value="계 산"
style="color:white;
background-color:green;"></p></form>
</body>
</html>
<?
//Area.PHP3 예제 소스
$_CF = array(1, 0.01);
$_L_Name = array("m", "cm");
echo("<p><b>계산 결과 :<small></b>");
$A = $A/pow($_CF[$_lUnit],2);
echo("<p> 면적 = $A
$_L_Name[$_lUnit]<sup>2</sup></small>");
?>
```

HTML 코드를 적당한 파일로 저장하여 실행시키면 다음과 같은 창이 뜰 것이다.



이 창에서 단위 환산에 필요한 값을 입력하고 환산 단위를 선택하여 “계산” 버튼을 눌러 수행하면 단위 환산에 필요한 변수의 이름과 값이 서버로 전해지고 서버에 존재하는 “area.php3”라는 PHP 파일이 실행되고 그 연산 결과를 HTML 코드의 형태로 클라이언트의 브라우저로 전해져 결과를 보게 된다. 물론 웹 서버와 PHP 인터프리터를 설치하지 않은 컴퓨터에서는 PHP를 이용한 예제는 실행되지 않으므로 결과는 볼 수 없다.

3.3 자바스크립트와 PHP의 비교

얼핏 보기에는 자바스크립트가 훨씬 간단해 보일 것이다. 또 프로그램의 실행 속도도 자바스크립트가 더 빠를 것이다. 기본적으로 자바스크립트로 짠 프로그램의 경우에는 연산 과정의 소스가 이미 클라이언트 쪽에 있으므로 숫자나 단위를 바꾸면 동시에 결과를 볼 수 있다. 반면에 PHP를 이용한 경우에는 인터넷을 통하여 입력 값을 서버로 보내고 그 계산결과를 HTML 형태로 바꾸어 다시 돌려보내므로 인터넷의 부하에 따라 시간도 많이 소요되고 서버 쪽에도 부하가 걸리게 된다. 그러나 이러한 단편적이고 간단한 예제로 두 언어의 모든 것을 판단하는 것은 매우 부적절하다. 만약 단위 환산과 같이 길이가 짧으면서 간단한 코드가 아니라 매우 복잡하고 긴 프로그램이라면 상황은 정반대가 될 것이다. 즉, 공학연산에 필요한 소스 코드 부분이 약 10000줄이라고 하자. 자바스크립트나 PHP 어느 경우든 큰 차이는 없을 것이다. 그러나 자바스크립트의 경우에는 전체 코드를 클라이언트에 보내야 하지만

PHP의 경우에는 필요한 입력 값과 출력 값만 주고 받는 부하와 시간이 소요되므로 PHP로 프로그램하는 것이 더 유리할 것이다. 물론 자바스크립트의 경우에는 클라이언트의 CPU를 이용하여 연산하므로 서버에 부담을 안 주게 되지만 사용하는 클라이언트의 성능에 따라 연산 속도는 큰 차이를 보이게 된다. 만약 이 소스가 크기가 매우 크고 고속의 전용 서버에서만 연산이 가능 할 정도의 프로그램이라면 자바스크립트를 이용하는 것이 불가능하다는 것은 자명하다. 소스 코드의 노출 등 보안성이나 여러 가지 면을 고려하면, 둘 중의 어느 것이 더 좋은 언어인가라는 질문은 아예 성립되지 않는 것이다.

4. 맺 음 말

본 원고에서는 자바와 자바스크립트 등을 포함한 대표적인 인터넷 언어들의 주요 특징 및 장단점에 대하여 알아보았다. 또 자바스크립트와 PHP를 이용하여 면적 환산을 하는 실제 소스 코드를 보이고 어떻게 차이가 나는지 간략히 비교하여 보았다.

본 원고의 제목은 인터넷 언어들을 공학 계산에 어떻게 응용할 수 있는가 이었지만 실제의 응용 분야는 사용자의 상상의 한계에 따를 것이라고 생각한다. 각자의 실험실에 인터넷에 연결된 서버를 놓고 수시로 인터넷이 있는 어느 곳에서나 필요한 프로그램을 수행하고 결과를 볼 수 있는 것은 물론 서버에 연결된 장비를 제어하는 프로그램을 통해서 원격 계측이나 실험 같은 것도 가능 할 것이다.