

```

#!/usr/local/bin/sybperl5 -T
# $Header: /usr/local/cvsroot/webley/agents/www/www.pl,v 1.1
1999/07/12 21:20:56 zhukoff Exp $
# Agent to get info from the web

use URI::URL;
use LWP::UserAgent;
use HTTP::Request::Common;

#print "www: $ARGV[0] $ARGV[1]\n";
#exit;
my $debug = 0;
my $debug2 = 0;

%Data = &check_local( @ARGV );
%Data = &check_global( @ARGV ) if $Data{ ERROR };
exit;

sub check_local {
    my ( $type, @param ) = @_;
    my %Data = ( ERROR => "In construction" );
    return %Data;
}

sub check_global {
    my ( $type, @param ) = @_;
    if ( $type eq "weather" ) {
        if ( $param[ 0 ] =~ /\d+/ ) {
            # zip supplied
            &print_weather( &weather_zip( $ARGV[ 1 ] ) );
        }
        else {
            # city supplied
        }
    }
    elsif ( $type eq "stock" ) {
        &print_stock( &stock_ticker( $ARGV[ 1 ] ) );
    }
}

sub stock_ticker {
    my ( $ticker ) = @_;
    my %Stock;
    %Stock = &stock_ticker_yahoo( $ticker );
    return %Stock;
}

sub weather_zip {
    my ( $zip ) = @_;
    my %Weather;
    %Weather = &weather_zip_cnn( $zip );
    &print_weather( %Weather ) if $debug2;
    return %Weather unless $Weather{ ERROR };
    undef %Weather;
    %Weather = &weather_zip_weather( $zip );
}

```

```

    &print_weather( %Weather ) if $debug2;
    return %Weather unless $Weather{ ERROR };
    undef %Weather;
    %Weather = &weather_zip_snap( $zip );
    &print_weather( %Weather ) if $debug2;
    return %Weather unless $Weather{ ERROR };
    undef %Weather;
    %Weather = &weather_zip_infospace( $zip );
    &print_weather( %Weather ) if $debug2;
    return %Weather unless $Weather{ ERROR };
    undef %Weather;
    %Weather = &weather_zip_lycos( $zip );
    return %Weather;
}
#http://cgi.cnn.com/cgi-bin/weather/redirect?zip=$zip
#http://www.weather.com/weather/us/zips/$zip.html
#http://feeds.infospace.com/info/USWS?QC=&QS=&QZ=$zip
#http://home.snap.com/main/local/1,37,-3,00.html?zipCode=60004

#http://weather.lycos.com/wcfiveday.asp?city=20000

#http://www.wunderground.com/cgi-bin/findweather/getForecast?
query=60053
#http://www.accuweather.com/weatherf/iwxpage_mainf_qx01?zipcode=
60089
#http://infoseek.go.com/Centers/News/Weather?tid=1491&query=60089
&sv=IS&lk=noframes

#http://www.webcrawler.com/weather/forecast/city/?
forecast_search=60053

#http://www3.travelocity.com/weather/searchvalidate/1,1217,,00.ht
ml?searchCity=kiev
sub get_url_content {
    my ( $url ) = @_;
    my $ua = LWP::UserAgent->new;
    $ua->agent( 'Mozilla/4.0 [en] (X11; I; FreeBSD 2.2.8-STABLE
i386)' );
    $ua->proxy( ['http', 'https'], 'http://proxy.vail:3128/' );
    $ua->no_proxy( 'webley', 'vail' );
    $url = url $url;
    print "$url\n" if $debug2;
    my $time = time;
    my $res = $ua->request( GET $url );
    print "Response: " . ( time - $time ) . "sec\n" if $debug2;
    return( $time, $res->content );
}
sub stock_ticker_yahoo {
    my ( $stock ) = @_;
    my %Stock;
    my $url = "http://finance.yahoo.com/d/quotes.csv?s=
$stock&f=sllldltlclhgv&e=.csv";

```

```

$Stock{ 'URL' } = $url;
( $Stock{ 'TIME' }, $_ ) = &get_url_content( $url );
( $Stock{ ticker },
  $Stock{ price },
  $Stock{ date },
  $Stock{ time },
  $Stock{ change },
  $Stock{ open },
  $Stock{ high },
  $Stock{ low },
  $Stock{ volume }
)
    = /\"([\^,]+)\",([\^,]+),\"([\^,]+)\",
\"([\^,]+)\",([\^,]+),([\^,]+),([\^,]+),([\^,]+),([\^,]+)/;
# split( /,/ );
#   if ( lc( $stock ) ne lc( $Stock{ ticker } ) ) {
#     if ( $Stock{ price } == 0 ) {
#       print "price $Stock{ price }\n";
#       my %Stock = ( ERROR => "$url - Could not match",
#                   content => $_ );
#       return %Stock;
#     }
#   }
return %Stock;
}
}
sub weather_zip_lycos {
    my ( $zip ) = @_;
    my $ua = LWP::UserAgent->new;
    $ua->proxy( ['http', 'https'], 'http://proxy.vail:3128/' );
    $ua->no_proxy( 'webley', 'vail' );
    my $url = url "http://weather.lycos.com/wcfiveday.asp?city=
$zip";
    print "$url\n" if $debug2;
    my $time = time;
    my $res = $ua->request(GET $url);
    print "Response: " . ( time - $time ) . "sec\n" if $debug2;
    $_ = $res->content;
    s/\n/ /g;
#   s/<br>/br/g;
#   s/<\/TD>/td/g;
#   s/<!.*?->/g;
#   s/<[\^<>]+>/g;
#   s/\&nbsp;/ /g;
#   s/\s+/ /g;
    print if $debug;
    my ( $Location,
        $current_day,
        $current_temperature_F,
        $current_weather,
        $Wind,
        $Humidity,
        $Pressure,
        $first_day_name,

```

```

    $second_day_name,
    $third_day_name,
    $fourth_day_name,
    $fifth_day_name,
    $first_day_weather,
    $second_day_weather,
    $third_day_weather,
    $fourth_day_weather,
    $fifth_day_weather,
    $first_day_high_F,
    $first_day_low_F,
    $second_day_high_F,
    $second_day_low_F,
    $third_day_high_F,
    $third_day_low_F,
    $fourth_day_high_F,
    $fourth_day_low_F,
    $fifth_day_high_F,
    $fifth_day_low_F,
    )
    =
        /Get a personalized forecast (?:.+)\. td (.+)td
    (.+) td td Current Conditions: temperature: (\S+)\&deg; F or C
    \&deg; sky: (.+) wind: (.+) relative humidity: (\S+) barometer:
    (.+) td td td 5-Day forecast: td (\S+)td (\S+)td (\S+)td (\S+)td
    (\S+)td td td td td td td (.+) td (.+) td (.+) td (.+) td (.+) td hi
    (\S+)\&deg;lo (\S+)\&deg;td hi (\S+)\&deg;lo (\S+)\&deg;td hi
    (\S+)\&deg;lo (\S+)\&deg;td hi (\S+)\&deg;lo (\S+)\&deg;td hi
    (\S+)\&deg;lo (\S+)\&deg;td td td \&#160;td Enter a city for
    forecasts/;
    if ( !$Location ) {
        my %Weather = ( ERROR => "$url - Could not match",
                        content => $_ );

        return %Weather;
    }
    $current_weather =~ s/p\//partly /;
    $current_weather =~ s/m\//mostly /;
    $current_weather =~ s/t-/thunder /;
    $Wind =~ s/N/North/g;
    $Wind =~ s/W/West/g;
    $Wind =~ s/E/East/g;
    $Wind =~ s/S/South/g;
    $Wind =~ s/mp\h/miles per hour/;
    $Wind =~ s/kph/kilometers per hour/;
    $Weather{ 'Location' } = $Location;
    $Weather{ 'current month' } = $current_month;
    $Weather{ 'current day' } = $current_day;
    $Weather{ 'current time' } = $current_time;
    $Weather{ 'current weather' } = $current_weather;
    $Weather{ 'current temperature, F' } =
$current_temperature_F;
    $Weather{ 'current temperature, C' } =

```

```

$current_temperature_C;
    $Weather{ 'Humidity' } = $Humidity;
    $Weather{ 'Wind' } = $Wind;
    $Weather{ 'URL' } = $url;
    $Weather{ 'TIME' } = $time;
    return %Weather;
}
sub weather_zip_snap {
    my ( $zip ) = @_ ;
    my $ua = LWP::UserAgent->new;
    $ua->proxy( ['http', 'https'], 'http://proxy.vail:3128/' );
    $ua->no_proxy( 'webley', 'vail' );
    my $url = url
"http://home.snap.com/main/local/1,37,-3,00.html?zipCode=$zip";
    print "$url\n" if $debug2;
    my $time = time;
    my $res = $ua->request(GET $url);
    print "Response: " . ( time - $time ) . "sec\n" if $debug2;
    $_ = $res->content;
    s/\n/ /g;
#    s/<br>/br/g;
    s/<\/td>/td/g;
    s/<!.*?->/g;
    s/<[^<>]+>/g;
    s/\&nbsp;/ /g;
    s/\s+/ /g;
    print if $debug;
    my ( $Location,
        $current_temperature_F,
        $first_day_high_F,
        $first_day_low_F,
        $current_weather,
        $Humidity,
        $Wind,
        $Pressure,
        $first_day_name,
        $second_day_name,
        $third_day_name,
        $fourth_day_name,
        $fifth_day_name,
        $first_day_weather,
        $second_day_weather,
        $third_day_weather,
        $fourth_day_weather,
        $fifth_day_weather,
        $first_day_high_F,
        $first_day_low_F,
        $second_day_high_F,
        $second_day_low_F,
        $third_day_high_F,
        $third_day_low_F,
        $fourth_day_high_F,

```

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.