

```
use strict;
use warnings;

my $text = "InternationaleBayernMunchen";
if ($text =~ /Inter/) {
    print "Inter plays the final\n";
} else {
    print "Mourinho is sad\n";
}

if ($text =~ /Internationale(.*)$/) {
    print "Inter plays $1 in the final\n";
} else {
    print "Mourinho is sad\n";
}

$text =~ s/Internationale/ACMilan/;
print "Wishful thinking: $text\n";

my @matches = ($text =~ m/a/g);
print "No. as in the text: @matches \n";

my $string = "axxxb";

if ($string =~ /[ab]..[ab]/) {
    print "Match $string\n";
} else {
    print "No match to $string\n";
}

$string = "xAbxb";
if ($string =~ /A[^xyz]/) {
    print "Match $string\n";
} else {
    print "No match to $string\n";
}

$string = " ";
if ($string =~ /(^s+)/) {
    print "Match WHITE $string: |$1|\n";
} else {
    print "No match to $string\n";
}

$string = "<A HREF=www.umu.se>";
if ($string =~ /<A HREF=(.*)>/) {
    print "Match $string: $1\n";
} else {
    print "No match to $string\n";
}

open (H, "hvidsten.com.html");
open (H2, ">hvidsten2.com.html");
while (<H>) {
    s/<h5>/<h1>/g;
    s/<\/h5>/<\/h1>/g;
    print H2 $_;
}
close (H);
close (H2);
```

```
use strict;
use warnings;

use Bio::Perl;

my @id;

open (S, "genpept.txt");
while (<S>) {
    chomp;
    push @id, split /\s/;
}
close (S);

print "@id\n";

foreach my $id (@id) {
    my $seq_object = get_sequence('genpept',$id);
    write_sequence(">>genpept.fasta",'fasta',$seq_object);
}
```

```
use strict;
use warnings;

use Bio::DB::GenPept;
use Bio::Seq;
use Bio::SeqIO;

my @id;

open (S, "genpept.txt");
while (<S>) {
    chomp;
    push @id, split /\s/;
}
close (S);

print "@id\n";

my $seqio_obj = Bio::SeqIO->new(-file => '>genpept2.fasta', -format => 'fasta'
);

my $db_obj = Bio::DB::GenPept->new;

foreach my $id (@id) {
    my $seq_obj = $db_obj->get_seq_by_id($id);
    $seqio_obj->write_seq($seq_obj);
}
```

```
use strict;
use warnings;

use Bio::SeqIO;
use Bio::Tools::Run::StandAloneBlast;

my $seqio_obj = Bio::SeqIO->new(-file => "c:\\Blast\\db\\yeast.nt", -format =>
"fasta" );

my $blast_obj = Bio::Tools::Run::StandAloneBlast->new((program => 'blastn',
database => 'yeast.nt'));

while (my $seq_obj = $seqio_obj->next_seq){
    print "RUN: ", $seq_obj->display_id, "\n";

    my $report_obj = $blast_obj->blastall($seq_obj);

    while(my $result = $report_obj->next_result ) {
        while(my $hit = $result->next_hit ) {
            while(my $hsp = $hit->next_hsp ) {
                if ($hit->name ne $seq_obj->display_id && $hsp->evaluate < 1e-10 ) {

                    print " Paralog: ", $hit->name, " Evaluate ", $hsp->evaluate, "\n";

                }
            }
        }
    }
    exit;
}
```