

```

#!/usr/bin/perl

use strict;
use warnings;

use Bio::Ensembl::Utils::Exception qw(throw warning);
use Bio::Ensembl::Registry;
use Bio::Ensembl::DBSQL::DBAdaptor;
use Getopt::Long;

# database
my $species; # = 'Homo sapiens';
my $host = 'ensembl.ensembl.org';
my $user = 'anonymous';

# genomic location
my $coord_system_name = 'chromosome';
my $coord_system_version = 'GRCh37';
my $reference_chromosome_name; # = '6';
my $chr_start = undef;
my $chr_end = undef;
my $chr_strand = undef;

# # # usage
my $help = '';

if ( !GetOptions( 'chromosome|c=s' => \$reference_chromosome_name,
                 'species|s=s' => \$species,
                 'help|h!' => \$help )
    || !( defined($reference_chromosome_name) && defined($species) )
    || $help )
{
    print <<END_USAGE;

Usage:
  $0 --species=species --chromosome=reference_chromosome_name
  $0 --help

    --species / -s  Name of species. Alternate loci are currently only
                    available for human.

    --chromosome / -c      Name of reference chromosome from Primary Assembly
                          containing alternate loci.

    --help      / -h  To see this text.

Example usage:

  $0 -s human -c 17

END_USAGE

    exit(1);
} ## end if ( !GetOptions( 'chromosome|c=s'...))

# # # usage

# connect to database:
my $registry = 'Bio::Ensembl::Registry';

```

```

$registry->load_registry_from_db( '-host' => $host,
                                '-user' => $user, );

# get adaptors
my $slice_adaptor = $registry->get_adaptor( $species, 'Core', 'Slice' );
my $asm_exception_adaptor = $registry->get_adaptor( $species, 'Core',
'AssemblyExceptionFeature' );
print "\nConnected to $species database\n\n";

# fetch reference chromosome
# $slice = $slice_adaptor-
>fetch_by_region('coord_system_name','seq_region_name', start, end, strand,
'coord_system_version');
my $ref_slice = $slice_adaptor->fetch_by_region($coord_system_name,
$reference_chromosome_name, $chr_start, $chr_end, $chr_strand,
$coord_system_version);

# and it's by calling alt_slice that we get the HAP / PAR
my @asm_except_feats = @{$asm_exception_adaptor->fetch_all_by_Slice($ref_slice)
};

# loop through these alternate loci to get the coordinates
foreach my $aef (@asm_except_feats) {
    # get the exception slice
    my $alt_slice = $aef->alternate_slice();
    # print the details
    print "Reference Name=" . $aef->seq_region_name
        . "\tReference Start=" . $aef->start
        . "\tReference End=" . $aef->end
        . "\tAlternate Locus Type=" . $aef->type
        . "\tAlternate Name=" . $aef->alternate_slice->seq_region_name
        . "\tAlternate Start=" . $aef->alternate_slice->start
        . "\tAlternate End=" . $aef->alternate_slice->end
        . "\n";
}

```