

A dark blue vertical bar on the left side of the page. A blue arrow points to the right from the bar, containing the text 'Curso 2016-2017'.

Curso 2016-2017

2ª Actividad obligatoria

Fundamentos de inteligencia
artificial

Several thin, curved lines in dark blue and light grey originate from the bottom left and curve upwards and to the right.

Héctor Luaces Novo

Índice

Información del alumno.....	3
Introducción.....	3
Descripción del dominio de conocimiento.....	3
Descripción de la metodología de desarrollo.....	4
Código de la herramienta.....	5
Inicio.pl.....	5
Util.pl.....	5
Constantes.pl.....	7
Datos_pesos.pl.....	9
Hechos.pl.....	22
Datos_bandas.pl.....	24
Bandas.pl.....	27
Usuario.pl.....	29
Pesos.pl.....	32
Titulos.pl.....	34
Titulos_datos.pl.....	40
Caso de prueba.....	44
Inicio de la aplicación.....	44
Consulta de géneros existentes.....	44
Consulta de características.....	45
Consulta de bandas.....	46
Mostrar la información de una banda.....	47
Mostrar los pesos de una banda.....	47
Generación de una banda con título especificado y características procedurales.....	48
Generación de una banda con título y características procedurales.....	49
Eliminación de bandas de la base de conocimientos.....	51
Buscar bandas por género.....	51
Generación de nombres para subgéneros.....	52
Comprobación de todos los géneros y bandas.....	53
Dificultades encontradas.....	54

Información del alumno

Nombre: Héctor Manuel Luaces Novo

DNI: 33343779N

Correo electrónico: hluaces2@alumno.uned.es
hector@luaces-novo.es

Teléfono: 123456789

Introducción

La herramienta que el alumno presenta para su evaluación en esta segunda actividad obligatoria intenta modelar una base de conocimientos basada en los **géneros musicales**.

Debido a la escala del enunciado y a las preferencias del alumno, éste ha decidido reducir el ámbito del proyecto única y exclusivamente a todos los sub géneros de la música tipo *heavy metal*.

El motivo de semejante selección -además de ser preferencia personal, como ya se mencionó- es que este estilo de música posee una enorme cantidad de sub géneros. A día de hoy, un medio no especializado como la *Wikipedia*¹ cataloga **58** subgéneros de este tipo de música (por limitaciones de tiempo no figuran todos ellos en la herramienta). Otros medios más especializados, como *los archivos del metal*² (el equivalente más cercano a la *Biblioteca de Alejandría* para este dominio de conocimiento) definen más de un centenar de sub géneros.

Muchos de estos subgéneros nacen para clasificar determinadas características de los grupos: temáticas líricas, velocidad, tipos de composición, etc. Sin embargo, a veces la clasificación se vuelve demasiado específica. Sirva como ejemplo que hay sub géneros denominados como *pirate metal*, enfocado a las líricas basadas en canciones cantadas por los antiguos piratas, o *kawaii metal*, basado en el mundo del *manga* -los populares comics Japoneses-.

En principio esto no sería problemático si no fuese porque la comunidad de amantes de este tipo de música (conocidos de forma coloquial como *metalheads* o *heavys*) suele ser especialmente puntillosa a la hora de clasificar una banda, hasta tal punto que equivocarse al hacerlo correctamente ha iniciado más de una pelea o ruptura entre amigos ¡o miembros de una banda!

Decidido a terminar con las clasificaciones arbitrarias y subjetivas, opté finalmente por crear una base de conocimientos que fuese capaz de catalogar las sutiles diferencias que, en ocasiones, separan a un género de otro, de forma que ésta pudiese utilizarse para clasificar una banda de *heavy metal* en su subgénero correspondiente sin que los criterios subjetivos de cada juez se vean involucrados.

Descripción del dominio de conocimiento

Para conseguir los objetivos ya descritos, y sabiendo que los sub géneros son muy parecidos entre ellos, decidí crear un sistema de reglas que fuese capaz de definir las características que definen a una banda de música, relacionando éstas con los distintos subgéneros mediante un valor que será conocido de ahora en adelante como **peso**.

1 https://en.wikipedia.org/wiki/Heavy_metal_genres

2 <http://www.metal-archives.com>

Este peso definirá cuán importante es una característica para cada subgénero, p.ej.: el género conocido como *black metal* se basa en su velocidad y líricas, teniendo unos pesos altos en estas características, mientras que un género distinto no valorará de forma tan alta estos valores.

Sabiendo esto y teniendo en cuenta que una banda está compuesta por varias características, la herramienta es capaz de sumar los pesos de cada subgénero para cada una de ellas, hallando así el subgénero con mayor peso para la banda, que será el que la defina y catalogue.

Esto permite a la base de conocimiento predecir, con bastante exactitud, el género al que pertenecen bandas que pueden ser creadas de forma arbitraria (característica que, de hecho, permite la herramienta a fin de ilustrar su funcionamiento).

A mayores de lo anterior, y debido a que el alumno quería explorar las capacidades de prolog para generar reglas lingüísticas, la base de conocimientos también será capaz de generar un nuevo nombre para una banda de un subgénero dado.

Esto lo hace definiendo las reglas que marcan **cómo** se compone un nombre válido a nivel semántico, así como teniendo una base de conocimientos de palabras clasificadas en categorías (sustantivos, adjetivos, artículos, conjunciones, etc.), que además están relacionadas con cada género. Con este conocimiento es posible generar un título arbitrario que encaje con las temáticas y características de cada género.

Si bien la lingüística no es el fuerte del alumno, la experiencia si le sirvió para entender el alcance de los lenguajes del paradigma lógico a la hora de crear este tipo de sistemas.

Para modelar la base de conocimientos me basé en mi experiencia personal, fervoroso amante de este tipo de música, así como en infinidad de artículos de internet, foros especializados y largas conversaciones con gente experta en los muchos conciertos a los que asistí a lo largo de mi vida.

Descripción de la metodología de desarrollo

El desarrollo fue, quizás, la parte más sencilla de la aplicación. Lo que más tiempo costó al alumno fue aprender prolog, conseguir decidirse por uno de los temas propuestos y, sobre todo, conseguir aprender a pensar en términos compatibles con lenguajes lógicos.

Para aprender prolog me basé, principalmente, en la conocida página *learn prolog now*³, la cual acompañé de infinidad de ensayos realizados sobre un entorno de pruebas.

Una vez pensé, erróneamente, que había entendido algo de lo estudiado, empecé a revisar los trabajos de años anteriores en aras de ganar más conocimientos prácticos con el idioma.

Esto no salió bien y tuve que volver a la casilla inicial, pues seguía sin entender del todo el paradigma lógico. No fue hasta que repetí todo lo que había estudiado, esta vez parándome a revisar los árboles de *backtracking* que hace prolog, que estuve preparado para asumir la tarea.

Después decidí el dominio de conocimiento a usar y comencé a documentarme sobre el mismo. Desde que decidí el tema sabía qué es lo que tendría que hacer la herramienta, pues ya de joven los muchos géneros de este tipo de música me resultaban confusos.

Finalmente comenzó el desarrollo el cual culminó con este documento.

³ <http://www.learnprolognow.org/>

Código de la herramienta

El código documentado a continuación se dividirá en sub capítulos que corresponderán con cada uno de los ficheros adjuntos.

Inicio.pl

```
:-encoding(utf8).

% Importamos útiles de sistema...
:-[util].

% Para reiniciar la base de conocimientos de ejecuciones anteriores
:-reiniciar.

% Importamos ficheros de la base de conocimientos
:-[constantes].
:-[hechos].
:-[bandas].
:-[pesos].
:-[datos_pesos].
:-[datos_bandas].
:-[titulos].
:-[titulos_datos].
:-[usuario].

% Iniciamos
:-ayuda.
```

Util.pl

```
:-encoding(utf8).

% Predicados de utilidad que se utilizan a lo largo de toda la
herramienta
% y cuya finalidad no está directamente relacionada con la base
% de conocimientos si no con utilidades para interactuar con
% listas arbitrarias, textos en pantallas, etc.

% Escribe y deja un salto de línea a continuación
writeln(X):-write(X),nl.

% Escribe todos los elementos de una lista dejando
% espacios entre ellos
writeall([]).
writeall([H|T]):- write(H),tab(1),writeall(T).

% Igual que writeall, pero además imprime un newline al final
writeallnl([]):- nl.
writeallnl([H|T]):- write(H),tab(1),writeallnl(T).

% Dada una lista, la muestra por pantalla como una lista
% desordenada
lista([]).
lista([H|T]):-
    writeallnl([" - ", H]),
    lista(T).

% Este predicado sirve para reiniciar la base de conocimientos
reiniciar:-
```

```

retractall(pesos(_,_,_)),
retractall(bandas(_,_)),
retractall(diccionario(_,_,_)).

% Determina si un elemento existe en una lista
contiene([],_):- false.
contiene([A|_], A).
contiene([_|T], X):-
    contiene(T, X).

% Determina si una lista dada es un conjunto, es decir
% si no hay ningún elemento repetido
conjunto([],_).
conjunto([H|T], Lista):-
    \+ contiene(Lista, H),
    conjunto(T, [H|Lista]).
conjunto(Lista):- conjunto(Lista, []).

% Devuelve la longitud de una lista
len([],0).
len([_|T],N) :- len(T,X), N is X+1.

% Devuelve el elemento N-ésimo de una lista
% Recorreremos recursivamente una lista con un acumulador
% que se irá incrementando.

% Caso base de recursión cuando el acumulador es igual a la
% posición Z que queremos encontrar
miembro([H|_], Z, Z, H).

% Caso de recursión, sumamos el acumulador y continuamos la
% recursión
miembro([_|T], Elemento, Acc, Z):-
    Acc2 is Acc + 1,
    miembro(T, Elemento, Acc2, Z).

% Método de utilidad para el usuario que omite el acumulador
% y realiza comprobaciones de longitud para evitar introducirse
% por donde no debe.
miembro(Lista, Elemento, Z):-
    Elemento >= 1,
    len(Lista, X),
    X >= Elemento,
    miembro(Lista, Elemento, 1, Z).

% Devuelve un elemento aleatorio de una lista
elemento_aleatorio(Lista, Z):-
    len(Lista, Longitud),
    Elemento is random(Longitud)+1,
    miembro(Lista, Elemento, Z).

% Concatena todos los elementos de una lista en un único string
% usando un carácter especificado como separador

% Caso base recursión
implode([H|[]], _, Acc, R):-
    string_concat(Acc, H, R).

% Recursión con acumulador
implode([H|T], Caracter, Acc, R):-

```

```

string_concat(H, Character, H_Character),
string_concat(Acc, H_Character, Final),
implode(T, Character, Final, R).

```

```

% Wrapper para no tener que pasar acumulador
implode(Lista, Character, Ret):-
    implode(Lista, Character, '', Ret).

```

Constantes.pl

```

:-encoding(utf8).

% Biblioteca de constantes, principalmente para definir los átomos
% usados en la aplicación.
% Constantes de las características que vamos a usar.
genero.
formacion.
lirica.
tempo.
voz.
tecnica.

% Predicados dinámicos que añadimos aquí por tenerlo todo unificado

% Listado de subgéneros de Metal con los que trabajaremos
metal.
heavy_metal.
thrash_metal.
black_metal.
unblack_metal.
symphonic_black_metal.
cristian_metal.
death_metal.
technical_death_metal.
melodic_death_metal.
grindcore.
progressive_metal.
folk_metal.
viking_metal.
power_metal.
doom_metal.
nu_metal.
industrial_metal.

% Formaciones de instrumentos habituales en un grupo de metal
formacion_clasica.                % 2 guitarras, bajo y
batería
formacion_teclado.                % 2 guitarras, bajo,
batería y teclado
formacion_folk.                   % clásica con elementos
folk (laúdes, flautas, gaitas, etc.)
formacion_orquestal.              % clásica con elementos
orquestales (órganos, baterías de instrumentos de viento, etc.)
formacion_dj.                     % clásica soportada con
una mesa de mezclas.
formacion_sintetizador.           % clásica soportada con un
sintetizador u otros elementos electrónicos
formacion_alternativa.            % otras formaciones no
puestas aquí

```

% Tipos de temática lírica	
lirica_metal.	% el metal como forma de vida
lirica_religion. de religión	% promulgación de algún tipo
lirica_anti_religion.	% crítica de la religión
lirica_muerte. terceros	% muerte, de uno mismo o
lirica_horror. infundir miedo.	% temáticas que intentan
lirica_guerra. mundo	% guerras y conflictos del
lirica_historica. ya acaecidos	% narran eventos históricos
lirica_cristianismo. como religión	% promulgan el cristianismo
lirica_fantasia. criaturas mágicas	% mundos fantásticos y
lirica_filosofia. filosóficos	% exploran dilemas
lirica_espiritualidad. espirituales	% exploran dilemas
lirica_folk_y_mitologia. mitológicas	% canciones folk o leyendas
lirica_mitologia_nordica. panteón de los dioses nórdicos	% folklore mitológico del
lirica_muerte_luto_y_desesperacion. dolor causado por estas sensaciones	% centrado en explorar el
lirica_sangre_y_visceras. visceras	% recreación en la sangre y
lirica_sociedad. sociales	% Crítica de aspectos
lirica_introspeccion. personal	% introspección o superación
lirica_otra. mencionadas	% otras líricas no
% Tempo o velocidad de la canción en 'beats per minute'	
tempo_30bpm. lento	% Esto es extremadamente
tempo_60bpm.	% Lento o una balada
tempo_90bpm. encuentra aquí	% La mayoría del metal se
tempo_120bpm.	% Punto inicio thrash metal
tempo_160bpm. death	% Media thrash, inicio black/
tempo_200bpm.	% Black metal
tempo_240bpm.	% Death metal
tempo_superior. otras cosas sin sentido	% Grindcore, brutal death y
tempo_variante. cambios de tempo	% Música con constantes
% Tipos de voces principales	
voz_limpia. demasiado hacia un lado del espectro agudo/grave	% Una voz limpia sin moverse
voz_aguda. metal más clásico.	% Las características voces del
voz_grave. gutturales.	% Voces graves sin llegar a ser


```

voz_gritos_melodicos.          % Difícil técnica de voz que
implica gritar manteniendo las notas
voz_gritos.                    % Voz caracterizada por cantar a
gritos
voz_gutural.                   % Voz gutural de ultratumba
voz_gutural_limpia.            % Voz gutural con énfasis en la
claridad, por ligeramente más lentas que las guturales
voz_aullidos.                  % Gritos descontrolados y
prolongados
voz_death_growl.               % El 'death growl' es una técnica
de canto basada en gritos guturales rápidos y profundos
voz_otros.

% Distintas técnicas que pueden caracterizar a grupos
tecnica_solos_largos.          % Solos, de cualquier instrumento,
que suelen ocupar buena parte de las canciones de la banda
tecnica_musica_tecnica.        % Composición y ejecución de la
melodía de forma muy técnica, buscando sonidos innovadores
tecnica_double_kick_drumming.  % Una técnica de voz asociada a
los metales extremos que
tecnica_blast_beat.            % Técnica para tocar la batería
caracterizada por su velocidad y violencia sónica
tecnica_tremolo_riff.          % Técnica para tocar la guitarra
donde una misma nota se toca varias veces a altas velocidades
tecnica_fuerte_distorsion.     % Distorsión fuerte de las
guitarras y bajos para convertirlas en sonidos muchos más pesados
tecnica_shred_guitar.          % Técnica para tocar la guitarra
caracterizada por su velocidad y la complejidad de la técnica
requerida para su ejecución
tecnica_tapping.               % Técnica para tocar la guitarra
en la que, en lugar de utilizar las puas para hacer vibrar las
cuerdas, se usan las puntas de los dedos para golpearla contra el
traste de la guitarra
tecnica_liricas_complejas.     % Letras complejas y elaboradas,
dignas de mención por su mensaje
tecnica_canciones_melodicas.   % Canciones con una fuerte
tendencia melódica
tecnica_voz_rimas.             % Musicalidad lírica basada en
rimas

```

Datos_pesos.pl

```

:-encoding(utf8).

% En este archivo se definen los pesos de los distintos géneros y se
mantiene
% separado del resto ya que es información muy amplia que podría
dificultar
% las partes lógicas de la aplicación.

% En cada ejecución limpiamos la base de conocimiento existente
:-retractall(pesos(_,_,_)).

% El heavy metal, también conocido como "Metal" a secas, existía
% antes de que el género se dividiese en infinidad de
especializaciones
% y subgéneros. Por lo general se caracteriza por sus solos de
% guitarra, riffs de velocidad media-alta, un bajo potente y
% vocales limpias -por lo general agudas- y voces de coros
:-anyadir pesos(heavy metal, [

```

```

[formacion_clasica           , 8],
[formacion_teclado          , -5],
[formacion_folk              , -8],
[formacion_orquestal        , -2],
[formacion_dj                , -20],
[formacion_sintetizador     , -20],
[formacion_alternativa      , -6],

[lirica_metal                , 10],
[lirica_religion             , -8],
[lirica_anti_religion        , -8],
[lirica_muerte               , 1],
[lirica_horror               , 1],
[lirica_guerra               , 4],
[lirica_historica            , 7],
[lirica_cristianismo         , -20],
[lirica_fantasia             , 5],
[lirica_filosofia            , 1],
[lirica_espiritualidad       , -4],
[lirica_folk_y_mitologia     , -8],
[lirica_mitologia_nordica    , -8],
[lirica_muerte_luto_y_desesperacion , 1],
[lirica_sangre_y_visceras    , -8],
[lirica_sociedad             , 5],
[lirica_introspeccion        , 5],

[tempo_30bpm                 , -8],
[tempo_60bpm                 , 8],
[tempo_90bpm                 , 6],
[tempo_120bpm                , -4],
[tempo_160bpm                , -8],
[tempo_200bpm                , -10],
[tempo_240bpm                , -15],
[tempo_superior              , -99],

[voz_limpia                  , 8],
[voz_aguda                   , 8],
[voz_grave                   , 3],
[voz_gritos_melodicos       , -2],
[voz_gritos                  , -2],
[voz_gutural                 , -6],
[voz_gutural_limpia         , -6],
[voz_aullidos                , -8],
[voz_death_growl            , -8],

[tecnica_solos_largos        , 10],
[tecnica_musica_tecnica      , -1],
[tecnica_double_kick_drumming , -10],
[tecnica_blast_beat         , -10],
[tecnica_tremolo_riff        , -8],
[tecnica_fuerte_distorsion   , 3],
[tecnica_shred_guitar        , 7],
[tecnica_tapping             , 7],
[tecnica_canciones_melodicas , 5],
[tecnica_voz_rimas           , -8]
]) .

```

```

% No es un género en sí mismo, pero el metal cristiano
% engloba a cualquier género que transmite mensajes
% cristianos. Téngase en cuenta que en el nacimiento
% del metal (años 70) éste era considerado como

```

```

% algo que atentaba contra la religión y se le asociaba
% con el diablo (algunos grupos son así, pero no todos)
% y el metal cristiano fue la respuesta a ese tipo de
% música. Por ello esta caracterización solo tiene en
% cuenta las vocales, ya que no hace referencia
% a un subgénero en concreto y es más bien
% una especie de "fallback"
:-anyadir_pesos(cristian_metal, [
    [lirica_cristianismo           , 99],
    [lirica_metal                 , -10],
    [lirica_religion              , -10],
    [lirica_anti_religion         , -99],
    [lirica_muerte               , -10],
    [lirica_horror               , -10],
    [lirica_guerra               , -10],
    [lirica_historica            , -10],
    [lirica_fantasia             , -10],
    [lirica_filosofia            , -10],
    [lirica_espiritualidad       , -10],
    [lirica_folk_y_mitologia     , -10],
    [lirica_mitologia_nordica    , -10],
    [lirica_muerte_luto_y_desesperacion , -10],
    [lirica_sangre_y_visceras    , -10],
    [lirica_sociedad            , -10],
    [lirica_introspeccion        , -10],
    [lirica_otra                 , -10],
    [tempo_60bpm                 , 5],
    [tecnica_solos_largos        , 10],
    [tecnica_fuerte_distorsion   , 3],
    [tecnica_shred_guitar        , 7],
    [tecnica_tapping            , 7],
    [tecnica_canciones_melodicas , 5],
    [tecnica_voz_rimas          , -8]
]).

% Subgénero extremo, también conocido como "speed metal"
% que se caracteriza por voces a gritos, en ocasiones melódicas,
% con una formación tradicional que toca a gran velocidad,
% mejorando la velocidad de los solos del heavy tradicional
% y tocando temas como la sociedad y la crítica al poder
% como abanderados de sus líricas, sin dejar de lado la
% muerte, el horror u otros temas controvertidos
:-anyadir_pesos(thrash_metal, [
    [formacion_clasica           , 8],
    [formacion_teclado          , -10],
    [formacion_folk             , -10],
    [formacion_orquestal        , -8],
    [formacion_dj               , -20],
    [formacion_sintetizador     , -20],
    [formacion_alternativa      , -10],

    [lirica_sociedad            , 10],
    [lirica_religion            , -8],
    [lirica_anti_religion       , -8],
    [lirica_muerte              , 6],
    [lirica_horror              , 6],
    [lirica_guerra              , 4],
    [lirica_historica           , -4],
    [lirica_cristianismo        , -10],
    [lirica_fantasia            , -8],
    [lirica_filosofia           , -8],

```

```

[lirica_espiritualidad          , -8],
[lirica_folk_y_mitologia        , -8],
[lirica_mitologia_nordica       , -10],
[lirica_muerte_luto_y_desesperacion , -1],
[lirica_sangre_y_visceras       , 3],
[lirica_introspeccion           , -4],

[tempo_30bpm                    , -15],
[tempo_60bpm                    , -10],
[tempo_90bpm                    , -8],
[tempo_120bpm                   , 4],
[tempo_160bpm                   , 8],
[tempo_200bpm                   , 4],
[tempo_240bpm                   , -4],
[tempo_superior                 , -8],

[voz_limpia                     , -4],
[voz_aguda                      , 3],
[voz_grave                      , 7],
[voz_gritos_melodicos          , 8],
[voz_gritos                     , 6],
[voz_gutural                   , -6],
[voz_gutural_limpia            , -6],
[voz_aullidos                  , -8],
[voz_death_growl               , -8],
[voz_otros                     , -3],

[tecnica_solos_largos           , 5],
[tecnica_musica_tecnica         , 1],
[tecnica_double_kick_drumming   , -8],
[tecnica_blast_beat            , 2],
[tecnica_tremolo_riff           , 6],
[tecnica_fuerte_distorsion      , 6],
[tecnica_shred_guitar           , 6],
[tecnica_tapping                , 6],
[tecnica_canciones_melodicas    , 1],
[tecnica_voz_rimas              , -8]
])).

% El black metal es un género muy agresivo que se basa en su
% agresividad,
% lírica comunmente anti-religiosa, vocales escalofriantes y un
% tremolo-picking muy rápido que define el veloz.
% Suelen tener instrumentos clásicos, innovando en ocasiones con
% elementos orquestales u teclados.
:-anyadir_pesos(black_metal, [
    [formacion_clasica          , 6],
    [formacion_teclado          , 4],
    [formacion_folk              , -6],
    [formacion_orquestal        , 1],
    [formacion_dj                , -20],
    [formacion_sintetizador      , -10],
    [formacion_alternativa       , -5],

    [lirica_metal                , -8],
    [lirica_religion              , 15],
    [lirica_anti_religion        , 15],
    [lirica_muerte               , 6],
    [lirica_horror               , 6],
    [lirica_guerra               , 4],
    [lirica_historica            , 4],

```

```

[lirica_cristianismo           , -99],
[lirica_fantasia               ,  3],
[lirica_filosofia             , -5],
[lirica_espiritualidad        ,  3],
[lirica_folk_y_mitologia      ,  5],
[lirica_mitologia_nordica     ,  5],
[lirica_muerte_luto_y_desesperacion ,  4],
[lirica_sangre_y_visceras     ,  2],
[lirica_sociedad              , -8],
[lirica_introspeccion         , -4],
[lirica_otra                  , -4],

[tempo_30bpm                  , -10],
[tempo_60bpm                  , -10],
[tempo_90bpm                  , -6],
[tempo_120bpm                 , -4],
[tempo_160bpm                 ,  4],
[tempo_200bpm                 ,  8],
[tempo_240bpm                 ,  4],
[tempo_superior               , -8],

[voz_limpia                   , -10],
[voz_aguda                    , -8],
[voz_grave                    , -8],
[voz_gritos_melodicos        ,  6],
[voz_gritos                   , 10],
[voz_gutural                  ,  2],
[voz_gutural_limpia          ,  2],
[voz_aullidos                 , 10],
[voz_death_growl              , -8],
[voz_otros                    , -6],

[tecnica_solos_largos         , -5],
[tecnica_musica_tecnica       , -3],
[tecnica_double_kick_drumming , -6],
[tecnica_blast_beat           ,  6],
[tecnica_tremolo_riff         ,  7],
[tecnica_fuerte_distorsion    ,  6],
[tecnica_shred_guitar         , -5],
[tecnica_tapping              , -5],
[tecnica_canciones_melodicas , -2],
[tecnica_voz_rimas            , -8]
]).

% El unblack metal es exactamente lo mismo que el black metal con una
% enorme diferencia: en lugar de criticar y atacar la religión,
% enaltecen
% el cristianismo con líricas que hable de dios, los santos y la
% mitología
% de diversas sectas de la iglesia católica
:-copiar_pesos(black_metal, unblack_metal).
:-anyadir_pesos(unblack_metal, [
    [lirica_cristianismo , 99],
    [lirica_anti_religion, -90]
]).

% El black metal sinfónico es una especialización del black metal
% que utiliza elementos orquestales en sus actuaciones.
% Dichos elementos pueden ser reales (órganos, orquestas) o bien

```

```

% sintetizadores a tal efecto. Las voces también pueden ser
% limpias.
:-copiar_pesos(black_metal, symphonic_black_metal).
:-anyadir_pesos(symphonic_black_metal, [
    [formacion_orquestal      , 10],
    [formacion_teclado        , 8],
    [formacion_sintetizador   , 4],
    [voz_limpia               , 4],
    [tecnica_canciones_melodicas, 5],
    [tempo_160bpm             , 6],
    [tempo_200bpm             , 4],
    [tempo_240bpm             , -2]
]).

%palm mute tremolo growling screams blast beat
%violencia religion ocultismo naturaleza mitologia
%politica ciencia ficcion actos extremos distorsion
:-anyadir_pesos(death_metal, [
    [formacion_clasica          , 6],
    [formacion_teclado          , -4],
    [formacion_folk             , -10],
    [formacion_orquestal        , -10],
    [formacion_dj               , -20],
    [formacion_sintetizador     , -10],
    [formacion_alternativa      , -8],

    [lirica_metal               , -3],
    [lirica_religion            , -3],
    [lirica_anti_religion       , 4],
    [lirica_muerte              , 10],
    [lirica_horror              , 8],
    [lirica_guerra              , 8],
    [lirica_historica           , -4],
    [lirica_cristianismo        , -10],
    [lirica_fantasia            , -5],
    [lirica_filosofia           , -5],
    [lirica_espiritualidad      , -3],
    [lirica_folk_y_mitologia    , 4],
    [lirica_mitologia_nordica   , 4],
    [lirica_muerte_luto_y_desesperacion , 4],
    [lirica_sangre_y_visceras   , 4],
    [lirica_sociedad            , 4],
    [lirica_introspeccion       , -5],
    [lirica_otra                , -5],

    [tempo_30bpm                , -10],
    [tempo_60bpm                , -10],
    [tempo_90bpm                , -10],
    [tempo_120bpm               , -6],
    [tempo_160bpm               , -4],
    [tempo_200bpm               , 4],
    [tempo_240bpm               , 10],
    [tempo_superior             , 8],

    [voz_limpia                 , -10],
    [voz_aguda                  , -8],
    [voz_grave                  , -4],
    [voz_gritos_melodicos       , -8],
    [voz_gritos                 , -3],
    [voz_qutural                , 10],

```

```

[voz_gutural_limpia          , 8],
[voz_aullidos                , -3],
[voz_death_growl            , 10],
[voz_otros                   , -6],

[tecnica_solos_largos        , -5],
[tecnica_musica_tecnica      , 2],
[tecnica_double_kick_drumming , 10],
[tecnica_blast_beat          , 10],
[tecnica_tremolo_riff         , 6],
[tecnica_fuerte_distorsion    , 6],
[tecnica_shred_guitar        , -5],
[tecnica_tapping             , -5],
[tecnica_canciones_melodicas , -5],
[tecnica_voz_rimas           , -8]
])).

% El death metal técnico junta la brutalidad y velocidad
% del death metal con el afán de mejorar la música a nivel
% técnico encontrado en otros géneros como el metal
% progresivo. Composiciones elaboradas, técnicas de
% guitarra refinadas y letras complejas son fáciles de encontrar
% en este subgénero
:-copiar_pesos(death_metal, technical_death_metal).
:-anyadir_pesos(technical_death_metal, [
    [formacion_alternativa      , 4],
    [voz_gutural_limpia         , 10],
    [tecnica_solos_largos        , 4],
    [tecnica_musica_tecnica      , 10],
    [tecnica_liricas_complejas   , 8],
    [tecnica_canciones_melodicas , 2],
    [tempo_200bpm                , 6],
    [tempo_240bpm                , 2]
])).

% Sugénero que toma como base el death metal añadiéndole
% arreglos melódicos con estrillo marcado, presencia de
% teclado y voces más limpios o formaciones de instrumentos
% poco habituales
:-copiar_pesos(death_metal, melodic_death_metal).
:-anyadir_pesos(melodic_death_metal, [
    [formacion_alternativa      , 7],
    [formacion_teclado           , 7],
    [voz_gutural_limpia         , 11],
    [tecnica_canciones_melodicas , 14]
])).

% El grindcore es uno de los subgéneros más difíciles de oír,
% incluso entre los amantes del metal, pues su velocidad,
% brutalidad y fuerza de las vocales lo hace un metal
% muy agresivo. Su temática lírica suele hacer referencia
% a actos extremos, sangre o política.
:-anyadir_pesos(grindcore, [
    [formacion_clasica           , 6],
    [formacion_teclado           , -10],
    [formacion_folk              , -10],
    [formacion_orquestal         , -10],
    [formacion_dj                , -20],
    [formacion_sintetizador      , -20],
    [formacion_alternativa       , -10],

```

```

[lirica_metal , -7],
[lirica_religion , -5],
[lirica_anti_religion , -5],
[lirica_muerte , 5],
[lirica_horror , 5],
[lirica_guerra , -5],
[lirica_historica , -5],
[lirica_cristianismo , -10],
[lirica_fantasia , -7],
[lirica_filosofia , -5],
[lirica_espiritualidad , -5],
[lirica_folk_y_mitologia , -8],
[lirica_mitologia_nordica , -8],
[lirica_muerte_luto_y_desesperacion , -3],
[lirica_sangre_y_visceras , 10],
[lirica_sociedad , 8],
[lirica_introspeccion , -6],
[lirica_otra , -6],

[tempo_30bpm , -10],
[tempo_60bpm , -10],
[tempo_90bpm , -10],
[tempo_120bpm , -10],
[tempo_160bpm , -10],
[tempo_200bpm , -6],
[tempo_240bpm , 8],
[tempo_superior , 15],
[tempo_variante , -5],

[voz_limpia , -10],
[voz_aguda , -10],
[voz_grave , -10],
[voz_gritos_melodicos , -10],
[voz_gritos , 4],
[voz_gutural , 10],
[voz_gutural_limpia , 2],
[voz_aullidos , 4],
[voz_death_growl , 10],
[voz_otros , -10],

[tecnica_solos_largos , -7],
[tecnica_musica_tecnica , -7],
[tecnica_double_kick_drumming , 10],
[tecnica_blast_beat , 10],
[tecnica_tremolo_riff , 4],
[tecnica_fuerte_distorsion , 10],
[tecnica_shred_guitar , -7],
[tecnica_tapping , -7],
[tecnica_canciones_melodicas , -10],
[tecnica_voz_rimas , -8],
]),

% Subgénero que combina el heavy metal con el
% rock progresivo, buscando sonidos técnicos y
% complejos dentro de la instrumentación
% habitual del heavy metal
:-anyadir_pesos(progressive_metal, [
    [formacion_clasica , 6],
    [formacion_teclado , 6],
    [formacion_folk , -5],
    [formacion_orquestal , -5],

```



```

[formacion_dj                      , -20],
[formacion_sintetizador            , -10],
[formacion_alternativa              , -5],

[lirica_metal                      , -4],
[lirica_religion                    , -7],
[lirica_anti_religion              , -8],
[lirica_muerte                      , -5],
[lirica_horror                     , -5],
[lirica_guerra                     , -5],
[lirica_historica                   , -5],
[lirica_cristianismo                , -10],
[lirica_fantasia                    , -5],
[lirica_filosofia                   , 8],
[lirica_espiritualidad              , 7],
[lirica_folk_y_mitologia            , -5],
[lirica_mitologia_nordica           , -8],
[lirica_muerte_luto_y_desesperacion , -5],
[lirica_sangre_y_visceras           , -8],
[lirica_sociedad                    , 7],
[lirica_introspeccion               , -8],
[lirica_otra                        , 3],

[tempo_30bpm                        , 2],
[tempo_60bpm                        , 6],
[tempo_90bpm                        , 6],
[tempo_120bpm                       , 2],
[tempo_variante                     , 15],
[tempo_160bpm                       , -5],
[tempo_200bpm                       , -5],
[tempo_240bpm                       , -5],
[tempo_superior                     , -5],

[voz_limpia                         , 6],
[voz_aguda                          , 6],
[voz_grave                          , 6],
[voz_gritos_melodicos               , 5],
[voz_gritos                         , 5],
[voz_gutural                        , 1],
[voz_gutural_limpia                 , 1],
[voz_aullidos                       , -6],
[voz_death_growl                    , -6],

[tecnica_solos_largos                , 7],
[tecnica_musica_tecnica              , 10],
[tecnica_double_kick_drumming        , -8],
[tecnica_blast_beat                  , -8],
[tecnica_tremolo_riff                 , 5],
[tecnica_fuerte_distorsion            , 4],
[tecnica_shred_guitar                , 10],
[tecnica_liricas_complejas           , 8],
[tecnica_tapping                     , 5],
[tecnica_canciones_melodicas         , 3],
[tecnica_voz_rimas                   , -8]
]]).

```

```

% El folk metal es un subgénero que se centra en la
% temática y composición de la banda, favoreciendo
% instrumentación folk (flautas, violines, laudes,
% gaitas, etc.) y una temática basada la mitología,
% orígenes e historia de la étnia o nacionalidad

```

```

% de la banda. Por ello los grupos folk
% pueden ser melódicos, extremos, rápidos, lentos...
:-anyadir_pesos(folk_metal, [
    [formacion_clasica           , 2],
    [formacion_teclado           , 6],
    [formacion_folk               , 15],
    [formacion_orquestal         , -4],
    [formacion_dj                 , -20],
    [formacion_sintetizador       , -10],
    [formacion_alternativa        , -10],

    [lirica_historica             , 8],
    [lirica_folk_y_mitologia       , 15],
    [lirica_mitologia_nordica     , 8],
    [lirica_fantasia              , 4],
    [lirica_metal                 , -10],
    [lirica_religion              , -10],
    [lirica_anti_religion         , -10],
    [lirica_muerte                , -10],
    [lirica_horror                , -10],
    [lirica_guerra                , -10],
    [lirica_cristianismo          , -10],
    [lirica_filosofia             , -10],
    [lirica_espiritualidad        , -10],
    [lirica_muerte_luto_y_desesperacion , -10],
    [lirica_sangre_y_visceras     , -10],
    [lirica_sociedad              , -10],
    [lirica_introspeccion         , -10],
    [lirica_otra                  , -10],
    [tecnica_voz_rimas            , -8]
]).

% El doom metal es un género caracterizado por ser lento,
% denso y atmosférico, haciendo hincapié en ello con tempos
% muy lentos y grandes distorsiones. Sus líricas
% suelen hacer referencia a sentimientos negativos
% relacionados con la muerte, la depresión, el suicidio
% o la introspección personal
:-anyadir_pesos(doom_metal, [
    [formacion_clasica           , 6],
    [formacion_teclado           , 6],
    [formacion_folk               , -4],
    [formacion_orquestal         , -4],
    [formacion_dj                 , -20],
    [formacion_sintetizador       , 2],
    [formacion_alternativa        , -10],

    [lirica_metal                 , -7],
    [lirica_religion              , -7],
    [lirica_anti_religion         , -7],
    [lirica_muerte                , 10],
    [lirica_horror                , 6],
    [lirica_guerra                , -7],
    [lirica_historica             , -7],
    [lirica_cristianismo          , -7],
    [lirica_fantasia              , -7],
    [lirica_filosofia             , -7],
    [lirica_espiritualidad        , -7],
    [lirica_folk_y_mitologia       , -7],
    [lirica_mitologia_nordica     , -7],
    [lirica_muerte_luto_y_desesperacion , 15]
]).

```

```

[lirica_sangre_y_visceras      , -7],
[lirica_sociedad              , -7],
[lirica_introspeccion         ,  4],
[lirica_otra                  ,  2],

[tempo_30bpm                  , 15],
[tempo_60bpm                  ,  4],
[tempo_90bpm                  , -7],
[tempo_120bpm                 , -7],
[tempo_160bpm                 , -7],
[tempo_200bpm                 , -7],
[tempo_240bpm                 , -7],
[tempo_superior               , -7],

[voz_limpia                   ,  6],
[voz_aguda                    , -5],
[voz_grave                    , -5],
[voz_gritos_melodicos        , -5],
[voz_gritos                   , -5],
[voz_gutural                  ,  6],
[voz_gutural_limpia          ,  6],
[voz_aullidos                 , -7],
[voz_death_growl              , -7],
[voz_otros                    , -7],

[tecnica_solos_largos         , -5],
[tecnica_musica_tecnica       , -5],
[tecnica_double_kick_drumming , -5],
[tecnica_blast_beat           , -5],
[tecnica_tremolo_riff         , -5],
[tecnica_fuerte_distorsion    ,  8],
[tecnica_shred_guitar         , -5],
[tecnica_tapping              , -5],
[tecnica_voz_rimas            , -8]
])).

% Subgénero del folk metal basado en la mitología nórdica
:-copiar_pesos(folk_metal, viking_metal).
:-anyadir_pesos(viking_metal, [
    [lirica_mitologia_nordica, 20]
])).

% clean, melodic, agudo, melodic lead guitar, fantasia,
% upbeat, fantasia, guerra, meurte
:-anyadir_pesos(power_metal, [
    [formacion_clasica      ,  6],
    [formacion_teclado      ,  6],
    [formacion_folk         ,  1],
    [formacion_orquestal    ,  2],
    [formacion_dj            , -20],
    [formacion_sintetizador , -10],
    [formacion_alternativa  , -10],

    [lirica_metal           , -8],
    [lirica_religion        , -8],
    [lirica_anti_religion   , -8],
    [lirica_muerte          , -8],
    [lirica_horror          , -8],
    [lirica_guerra          ,  4],
    [lirica_historica       , -8],
    [lirica_cristianismo    , -10],

```

```

[lirica_fantasia           , 10],
[lirica_filosofia         , -8],
[lirica_espiritualidad    , -8],
[lirica_folk_y_mitologia  , -8],
[lirica_mitologia_nordica , -8],
[lirica_muerte_luto_y_desesperacion , -8],
[lirica_sangre_y_visceras , -8],
[lirica_sociedad          , -8],
[lirica_introspeccion     , 8],
[lirica_otra              , -8],

[tempo_30bpm              , -5],
[tempo_60bpm              , -5],
[tempo_90bpm              , -5],
[tempo_120bpm             , 6],
[tempo_160bpm             , 8],
[tempo_200bpm             , 8],
[tempo_240bpm             , 4],
[tempo_superior           , -4],

[voz_limpia               , 8],
[voz_aguda                 , 8],
[voz_grave                 , -7],
[voz_gritos_melodicos     , 4],
[voz_gritos                , -7],
[voz_gutural              , -7],
[voz_gutural_limpia       , -7],
[voz_aullidos             , -7],
[voz_death_growl          , -7],
[voz_otros                 , -7],

[tecnica_solos_largos     , 10],
[tecnica_musica_tecnica   , 4],
[tecnica_double_kick_drumming , -8],
[tecnica_blast_beat       , -8],
[tecnica_tremolo_riff      , -8],
[tecnica_fuerte_distorsion , 2],
[tecnica_shred_guitar     , 10],
[tecnica_tapping          , 10],
[tecnica_canciones_melodicas , 6],
[tecnica_voz_rimas        , -8]
]]).

```

```

% El nu metal es considerado por muchos, incluido el alumno
% autor de esta herramienta, como algo que no debería
% pertenecer al metal ni tener su apelativo.
% El motivo es que este simplifica la esencia del metal,
% rebajando y reduciendo los clásicos solos de guitarra
% en aras de sustituirlos por otros elementos como
% vocales con rimas (traídas del rap o hip-hop),
% secuenciadores o mesas de DJ.
% - blastbeat +distort -solos -complejidad -ritmo
% +hip-hop +scream +growl +rapping +personal
:-anyadir_pesos(nu metal, [
    [formacion_clasica      , 2],
    [formacion_teclado     , -7],
    [formacion_folk         , -7],
    [formacion_orquestal    , -7],
    [formacion_dj           , 15],
    [formacion_sintetizador , 8],
    [formacion_alternativa  , -7],

```

```

[lirica_metal                , -7],
[lirica_religion             , -7],
[lirica_anti_religion        , -7],
[lirica_muerte               , -7],
[lirica_horror               , -7],
[lirica_guerra               , -7],
[lirica_historica            , -7],
[lirica_cristianismo         , -7],
[lirica_fantasia             , -7],
[lirica_filosofia            , -7],
[lirica_espiritualidad       , -7],
[lirica_folk_y_mitologia     , -7],
[lirica_mitologia_nordica    , -7],
[lirica_muerte_luto_y_desesperacion , -7],
[lirica_sangre_y_visceras    , -7],
[lirica_sociedad             , 4],
[lirica_introspeccion        , 8],
[lirica_otra                 , 2],

[voz_limpia                  , -7],
[voz_aguda                   , -7],
[voz_grave                   , -7],
[voz_gritos_melodicos       , -5],
[voz_gritos                  , 8],
[voz_gutural                 , 8],
[voz_gutural_limpia         , -2],
[voz_aullidos                , -5],
[voz_death_growl            , -1],
[voz_otros                   , 8],

[tecnica_solos_largos        , -10],
[tecnica_musica_tecnica      , -10],
[tecnica_double_kick_drumming , -10],
[tecnica_blast_beat         , -10],
[tecnica_tremolo_riff        , -5],
[tecnica_fuerte_distorsion   , 10],
[tecnica_shred_guitar        , -5],
[tecnica_tapping             , -5],
[tecnica_voz_rimas           , 15]
])).

% Combina el heavy metal y la música industrial,
% que hace fuerte uso de la distorsión, secuenciadores
% y sampling.
:-anyadir_pesos(industrial_metal, [
    [formacion_clasica        , -7],
    [formacion_teclado        , 2],
    [formacion_folk           , -7],
    [formacion_orquestal      , -7],
    [formacion_dj              , -4],
    [formacion_sintetizador    , 10],
    [formacion_alternativa     , 1],

    [lirica_metal             , 2],
    [lirica_religion           , 4],
    [lirica_anti_religion      , 4],
    [lirica_muerte             , 4],
    [lirica_horror             , -5],
    [lirica_guerra             , 4],
    [lirica_historica          , 4],

```

```

[lirica_cristianismo           , -10],
[lirica_fantasia               , -2],
[lirica_filosofia              , 5],
[lirica_espiritualidad         , 1],
[lirica_folk_y_mitologia       , -7],
[lirica_mitologia_nordica      , -7],
[lirica_muerte_luto_y_desesperacion , -2],
[lirica_sangre_y_visceras      , -5],
[lirica_sociedad               , 8],
[lirica_introspeccion          , 7],
[lirica_otra                   , 1],

[tempo_30bpm                   , -4],
[tempo_60bpm                   , 5],
[tempo_90bpm                   , 8],
[tempo_120bpm                  , 6],
[tempo_160bpm                  , -5],
[tempo_200bpm                  , -6],
[tempo_240bpm                  , -7],
[tempo_superior                , -8],

[voz_limpia                    , 2],
[voz_aguda                     , 2],
[voz_grave                     , 2],
[voz_gritos_melodicos         , 2],
[voz_gritos                    , 2],
[voz_gutural                   , 2],
[voz_gutural_limpia           , -7],
[voz_aullidos                  , -8],
[voz_death_growl               , -8],
[voz_otros                     , 2],

[tecnica_solos_largos          , -5],
[tecnica_musica_tecnica        , 1],
[tecnica_double_kick_drumming  , -7],
[tecnica_blast_beat            , -5],
[tecnica_tremolo_riff          , -5],
[tecnica_fuerte_distorsion     , 10],
[tecnica_shred_guitar          , -5],
[tecnica_tapping               , -5],
[tecnica_voz_rimas             , -8]
])).

% Una característica inexistente siempre tendrá peso 0 y devolverá
true.
:-assert(pesos(_ , _ , 0)).

```

Hechos.pl

```

:-encoding(utf8).

% Fichero en el que almacenamos la mayoría de hechos sin variables
% que relacionarán las distintas características con sus distintos
% nombres

% Definiciones de géneros musicales
genero(metal).
genero(heavy_metal).
genero(thrash_metal).

```

```
genero(black_metal).
genero(unblack_metal).
genero(symphonic_black_metal).
genero(cristian_metal).
genero(death_metal).
genero(technical_death_metal).
genero(melodic_death_metal).
genero(grindcore).
genero(progressive_metal).
genero(folk_metal).
genero(viking_metal).
genero(power_metal).
genero(doom_metal).
genero(nu_metal).
genero(industrial_metal).

% Formaciones de instrumentos
formacion(formacion_clasica).
formacion(formacion_teclado).
formacion(formacion_folk).
formacion(formacion_orquestal).
formacion(formacion_dj).
formacion(formacion_sintetizador).
formacion(formacion_alternativa).

% Temáticas líricas
lirica(lirica_metal).
lirica(lirica_religion).
lirica(lirica_anti_religion).
lirica(lirica_muerte).
lirica(lirica_horror).
lirica(lirica_guerra).
lirica(lirica_historica).
lirica(lirica_cristianismo).
lirica(lirica_fantasia).
lirica(lirica_filosofia).
lirica(lirica_espiritualidad).
lirica(lirica_folk_y_mitologia).
lirica(lirica_mitologia_nordica).
lirica(lirica_muerte_luto_y_desesperacion).
lirica(lirica_sangre_y_visceras).
lirica(lirica_sociedad).
lirica(lirica_introspeccion).
lirica(lirica_otra).

% Velocidades de interpretación
tempo(tempo_30bpm).
tempo(tempo_60bpm).
tempo(tempo_90bpm).
tempo(tempo_120bpm).
tempo(tempo_160bpm).
tempo(tempo_200bpm).
tempo(tempo_240bpm).
tempo(tempo_superior).
tempo(tempo_variante).

% Tipos de voz
voz(voz_limpia).
voz(voz_aguda).
voz(voz_grave).
voz(voz_gritos_melodicos).
```

```

voz(voz_gritos).
voz(voz_gutural).
voz(voz_gutural_limpia).
voz(voz_aullidos).
voz(voz_death_growl).
voz(voz_otros).

% Técnicas de ejecución
tecnica(tecnica_solos_largos).
tecnica(tecnica_musica_tecnica).
tecnica(tecnica_double_kick_drumming).
tecnica(tecnica_blast_beat).
tecnica(tecnica_tremolo_riff).
tecnica(tecnica_fuerte_distorsion).
tecnica(tecnica_shred_guitar).
tecnica(tecnica_tapping).
tecnica(tecnica_liricas_complejas).
tecnica(tecnica_canciones_melodicas).
tecnica(tecnica_voz_rimas).

% Predicado utilizado para determinar si un valor es una
% característica de la banda, entendiendo como característica
% una lírica, formación instrumental, forma de tocar, etc.
caracteristica(X):-
    lirica(X);
    formacion(X);
    tempo(X);
    voz(X);
    tecnica(X).

% Verifica si un conjunto de características se considera válido
% Se considerará válido si:
% - Es un conjunto (sin repetidos)
% - Todos sus elementos son características

% Caso base de recursión. El 1 es porque es el predicado que
% usa recursión. El predicado sin el 1 es el caso inicial
% y hace más comprobaciones
caracteristicas_validas([], 1).

% Función de recursión donde comprobamos que la característica
% que toque sea válida
caracteristicas_validas([H|T], 1):-
    caracteristica(H),
    caracteristicas_validas(T, 1).

% Función inicial, verificamos que la lista sea un conjunto
% e inicializamos la recursión
caracteristicas_validas(Lista):-
    conjunto(Lista),
    caracteristicas_validas(Lista, 1).

```

Datos_bandas.pl

```

:-encoding(utf8).

% En este fichero se definen las propiedades y predicados relacionados
% con
% las bandas, así como varias constantes basadas en bandas
% reales
% para que puedan ser usadas para evaluar en la herramienta
%
```



```

% Las bandas están compuestas por una lista que engloba las
características
% más relevantes que contiene. En principio será una característica de
cada,
% si bien no sería descabellado que pudiese tener repeticiones (p.ej.:
una banda
% que tuviese a la vez vocales guturales graves y agudas).
%
% La base de conocimiento no limita esto, si bien no creará bandas
automáticamente
% con repeticiones.

% Varias bandas predefinidas basadas en bandas reales

% heavy metal
:-anyadir_banda(iron_maiden, [
    formacion_clasica, lirica_metal, tempo_120bpm,
    voz_limpia
]).
:-anyadir_banda(judas_priest, [
    formacion_clasica, lirica_sociedad, tempo_90bpm,
    voz_aguda
]).

% thrash
:-anyadir_banda(slayer, [
    formacion_clasica, lirica_sociedad, tempo_160bpm,
    voz_gritos_melodicos, tecnica_fuerte_distorsion
]).
:-anyadir_banda(metallica, [
    formacion_clasica, lirica_sociedad, tempo_160bpm,
    voz_gritos, tecnica_shred_guitar
]).

% black
:-anyadir_banda(satyricon, [
    formacion_clasica, lirica_folk_y_mitologia, tempo_200bpm
]).
:-anyadir_banda(emperor, [
    formacion_teclado, lirica_anti_religion, tempo_240bpm,
    tecnica_tremolo_riff
]).

% symphonic
:-anyadir_banda(dimmu_borgir, [
    formacion_orquestal, lirica_anti_religion, tempo_200bpm,
    tecnica_canciones_melodicas,
    voz_gutural_limpia
]).

% unblack
:-anyadir_banda(antestor, [
    formacion_clasica, lirica_cristianismo,
    tempo_200bpm, voz_death_growl, tecnica_blast_beat,
    tecnica_double_kick_drumming
]).

%death
:-anyadir_banda(amon_amarth, [
    formacion_clasica, lirica_mitologia_nordica,
    tempo_240bpm, voz_gutural_limpia, tecnica_blast_beat

```

```
)).  
:-anyadir_banda(dethklok, [  
    formacion_clasica, lirica_sangre_y_visceras,  
    tempo_240bpm, voz_gutural, tecnica_solos_largos,  
    tecnica_shred_guitar  
])).  
  
%technical_death_metal  
:-anyadir_banda(gojira, [  
    formacion_clasica, lirica_otra,  
    tempo_240bpm, voz_gutural_limpia, tecnica_musica_tecnica,  
    tecnica_solos_largos  
])).  
  
%power  
:-anyadir_banda(rhapsody, [  
    lirica_fantasia, formacion_teclado, tempo_200bpm,  
    voz_aguda, tecnica_solos_largos  
])).  
:-anyadir_banda(blind_guardian, [  
    lirica_fantasia, formacion_clasica, tempo_160bpm,  
    voz_aguda, tecnica_shred_guitar  
])).  
  
%cristian_metal  
:-anyadir_banda(bloodgod, [  
    formacion_clasica, lirica_cristianismo, tempo_60bpm,  
    voz_limpia, tecnica_tapping, tecnica_shred_guitar,  
    tecnica_musica_tecnica  
])).  
  
%melodic_death_metal  
:-anyadir_banda(carass, [  
    formacion_clasica, lirica_sangre_y_visceras,  
    tempo_240bpm, voz_gutural_limpia, tecnica_solos_largos,  
    tecnica_canciones_melodicas  
])).  
:-anyadir_banda(at_the_gates, [  
    formacion_clasica, lirica_otra,  
    tempo_240bpm, voz_gutural_limpia, tecnica_solos_largos,  
    tecnica_canciones_melodicas  
])).  
  
%grindcore  
:-anyadir_banda(bolt_thrower, [  
    formacion_clasica, lirica_otra,  
    tempo_superior, voz_death_growl, tecnica_blast_beat,  
    tecnica_double_kick_drumming  
])).  
:-anyadir_banda(holocausto_canibal, [  
    formacion_clasica, lirica_sangre_y_visceras,  
    tempo_superior, tecnica_double_kick_drumming, voz_death_growl  
])).  
  
%progressive_metal  
:-anyadir_banda(amorphis, [  
    formacion_teclado, lirica_folk_y_mitologia,  
    tempo_variante, tecnica_shred_guitar, voz_limpia  
])).  
  
%folk_metal
```

```

:-anyadir_banda(turisas, [
    formacion_folk, lirica_folk_y_mitologia,
    tempo_160bpm, tecnica_canciones_melodicas, voz_gutural_limpia
])).

%viking_metal
:-anyadir_banda(finsterforst, [
    formacion_folk, lirica_mitologia_nordica,
    tempo_160bpm, tecnica_double_kick_drumming
])).

%doom_metal
:-anyadir_banda(katatonia, [
    formacion_clasica, lirica_muerte_luto_y_desesperacion,
    tempo_30bpm, voz_gutural, tecnica_fuerte_distorsion
])).

%nu_metal
% N.A: A pesar de lo que dije del nu_metal en otro fichero, a éstos
los salvo
:-anyadir_banda(korn, [
    formacion_clasica, lirica_introspeccion,
    tempo_60bpm, voz_gritos, tecnica_fuerte_distorsion,
    tecnica_voz_rimas
])).

%industrial_metal
:-anyadir_banda(victor_love, [
    formacion_sintetizador, lirica_introspeccion,
    tempo_90bpm, voz_gritos, tecnica_fuerte_distorsion
])).

```

Bandas.pl

```

:-encoding(utf8).

% En este fichero se definen las propiedades y predicados relacionados
% con las bandas.
%
% Conceptualmente hablando una banda es una asociación entre un nombre
% y un grupo de características del género.
%
% Por ejemplo, la banda Amon Amarth está caracterizada por
% una formación instrumental clásica, acompañada de voces guturales
% limpias, blast beating y una temática lírica que bebe de la
mitología
% nórdica.
%
% Este ejemplo anterior se representaría así:
% ?- banda(amon_amarth, [
%     formacion_clasica, lirica_mitologia_nordica,
%     voz_gutural_limpia, tecnica_blast_beat
% ]).
%
% El sistema será capaz de generar bandas aleatorias asignándoles
% características válidas. Dichas bandas posteriormente podrán ser
% clasificadas.
%

```

```

% Adicionalmente, el sistema ya incluirá varias bandas predefinidas
% existentes en el mundo real (ver datos_bandas.pl).

% Empezamos por definir un predicado dinámico que será utilizado
% para almacenar la lista de bandas de la aplicación.
% Los argumento serán los descritos en la cabecera de
% este fichero: (NOMBRE_BANDA, LISTA_CARACTERISTICAS)
dynamic(bandas/2).

% Determina si una banda con un nombre pasado como parámetro ya
% existe.
banda(Nombre):- bandas(Nombre,_).

% Añade una banda a la base de conocimientos
anyadir_banda(Nombre, Caracteristicas):-]).
    banda_valida(Nombre, Caracteristicas),
    assert(bandas(Nombre, Caracteristicas)).

% Predicado que define si una banda es válida.
% Una banda se considera válida si:
% - El nombre no existe
% - Sus características son válidas
banda_valida(Nombre, Lista):-
    \+ banda(Nombre),
    caracteristicas_validas(Lista).

% Genera un conjunto de características aleatorias
generar_caracteristicas(Caracteristicas):-
    % Formación aleatoria
    findall(A, formacion(A), Formaciones),
    elemento_aleatorio(Formaciones, Formacion),

    % Lírica aleatoria
    findall(B, lirica(B), Liricas),
    elemento_aleatorio(Liricas, Lirica),

    % Tempo aleatorio
    findall(C, tempo(C), Tempos),
    elemento_aleatorio(Tempos, Tempo),

    % Voces aleatorias
    findall(D, voz(D), Voces),
    elemento_aleatorio(Voces, Voz),

    % Técnicas aleatorias
    findall(E, tecnica(E), Tecnicas),
    elemento_aleatorio(Tecnicas, Tecnica),

    % Guardamos características
    Caracteristicas = [Formacion,Lirica,Voz,Tecnica,Tempo].

% Dado un nombre de banda que no exista en la base de conocimientos,
% genera un grupo de características aleatorias y se las asocia a la
% banda
generar_banda(Nombre, Caracteristicas):-
    \+ banda(Nombre),
    generar_caracteristicas(Caracteristicas),
    anyadir_banda(Nombre, Caracteristicas).

% Para tener mensajes de error verbose...

```

```

generar_banda(Nombre, _):-
    banda(Nombre),
    writeallnl(["Ya existe una banda con el nombre '", Nombre, "'."]),
    false.

generar_banda(Nombre):-
    generar_banda(Nombre, _),
    writeallnl(["Se ha generado la banda '", Nombre, "'"]).

```

Usuario.pl

```

:-encoding(utf8).

% Lista de predicados destinados a actuar de "frontend" de la base
% de conocimientos para que un usuario pueda cuestionarla sin
% conocer los detalles de su implementación

% Muestra el mensaje de bienvenida de la aplicación, que explica
% qué se puede hacer con ella
ayuda:-
    writenl(' Base de conocimientos del metal cargada y lista para ser
usada. '),
    writenl(' Los siguientes predicados pueden usarse para explorar la
base de conocimientos: '),
    nl,

    writenl(' ===== AYUDA ===== '),
    writenl(' - glosario                : para una breve explicación
del glosario usado en la base de conocimientos. '),
    nl,

    writenl(' ===== CONSULTAS GENERALES ===== '),
    writenl(' - mostrar_generos          : para ver los géneros con los
que trabaja la herramienta. '),
    writenl(' - mostrar_caracteristicas : para ver las características
que pueden tener las bandas. '),
    writenl(' - mostrar_bandas          : para ver que bandas de metal
existen en la aplicacion. '),
    nl,

    writenl(' ===== INFORMACIÓN ===== '),
    writenl(' - info_banda(X)            : muestra información de la
banda pasada como argumento. '),
    writenl(' - pesos_banda(X)          : muestra los pesos de una
banda para todos los distintos sub-géneros. '),
    nl,

    writenl(' ===== CREACIÓN / ELIMINACIÓN ===== '),
    writenl(' - generar_banda(X)        : crea una banda de
características aleatorias y le asigna el nombre X. '),
    writenl(' - generar_banda_aleatoria : crea una banda de título y
característiacs aleatorias. '),
    writenl(' - borrar_banda(X)         : elimina la banda de nombre X
de la base de conocimientos. '),
    nl,

    writenl(' ===== MISCELÁNEA ===== '),

```

```

    writenl(' - buscar_bandas(Genero) : muestra todas las bandas de
la base de conocimientos que sean de un género pasado como
argumento. '),
    writenl(' - generar_nombre(Genero) : genera un nombre de banda,
aceptando como argumento un subgénero del metal para el que queremos
generarla. '),
    writenl(' - ayuda : para volver a ver esta
pantalla. '),

    nl, writenl(" ~ They can't stop us, let'em try ~"),
    writenl(" ~ For Heavy Metal we will die! ~"),
    nl
.

% Muestra un breve glosario que puede servir para que un usuario que
% no tenga conocimientos de música pueda saber qué significan algunas
% de las palabras usadas en el mundo de la base de conocimientos
glosario:-
    writenl(" - banda : Entiéndase como 'banda' a un grupo de
música que ejecuta una musica basada en el metal."),
    writenl(" - característica: Los factores diferenciadores de cada
banda (p.ej.: potentes voces, riffs escalofriantes o una velocidad
endiablada)."),
    writenl(" - sub-género : El heavy metal se divide en un sinfín
de sub-géneros cada uno de los cuales tiene un grupo de
características que los identifican. Un subgénero es cada una de esas
especialiaciones."),
    writenl(" - peso : Las características llevan asociadas
un valor (o peso) que los relaciona con cada uno de los sub-géneros.
Cuanto más alto sea el peso, más significativa es la importancia de
esa característica para el género. El género con mayor peso para una
banda definirá la pertenencia de ésta al mismo."),
    writenl(" - riff : Una pequeña composición de guitarra.
Una canción está formada de muchos riffs (repetidos o no) seguidos."),
    writenl(" - tempo : Conceptualmente (aunque con matices)
es la velocidad de ejecución de la música. Cuanto más alto el tempo,
más rápida la música").

% Muestra todos los géneros y subgéneros
mostrar_generos:-
    writenl("Listado de géneros en la base de conocimientos:"),
    findall(X, genero(X), Z),
    lista(Z).

% Muestra todas las posibles características de las bandas
mostrar_caracteristicas:-
    writenl("Listado de características en la base de coocimientos:"),
    findall(X, caracteristica(X), Z),
    lista(Z).

% Muestra por pantalla todas las bandas creadas
mostrar_bandas:-
    writenl("Listado de bandas en la base de conocimientos:"),
    findall(X, banda(X), Z),
    lista(Z).

% Muestra información de una banda pasada como parámetro
info_banda(Banda):-
    \+ bandas(Banda, _),
    writeallnl(["La banda", Banda, "no existe en la base de
conocimientos"])).

```

```

info_banda(Banda):-
    writeallnl(["=== Información de la banda '", Banda, "' ==="]),
    writeallnl([" - Nombre: ", Banda]),
    dame_genero_banda(Banda, Genero),
    bandas(Banda, Caracteristicas),
    writeallnl([" - Género: ", Genero]),
    nl, writenl("Lista de características de la banda:"),
    lista(Caracteristicas).

% Genera una banda de nombre y características aleatorias
generar_banda_aleatoria:-
    generar_caracteristicas(Caracteristicas),
    dame_genero_caracteristicas(Caracteristicas, Genero),
    generar_nombre(Genero, Titulo),
    anyadir_banda(Titulo, Caracteristicas),
    writenl("Se ha generado aleatoriamente la siguiente banda:"),
    nl, writeallnl([Titulo, "(", Genero, ")"]), nl,
    writenl("Sus principales características son las siguientes:"),
    lista(Caracteristicas).

% Borra una banda de la base de conocimientos
borrar_banda(Banda):-
    \+ bandas(Banda, _),
    writeallnl(["No existe ninguna banda con el nombre '", Banda,
    "'"]).

borrar_banda(Banda):-
    retract(bandas(Banda, _));
    writeallnl("Se ha borrado la banda con éxito.").

% Genera un nombre y lo muestra por pantalla
generar_nombre(Genero):-
    \+ genero(Genero),
    writeallnl(["'", Genero, "' no parece ser un sub-género válido."]).

generar_nombre(Genero):-
    generar_nombre(Genero, X),
    writenl(X).

% Muestra por pantalla los pesos para cada género de la banda
% pasada como parámetro
pesos_banda(_, _, []).
pesos_banda(Nombre, Caracteristicas, [Genero|T]):-
    dame_peso_genero_banda(Caracteristicas, Genero, X),
    writeallnl([
        ' - ', Nombre, 'tiene un peso de ', X,
        'para el género ', Genero
    ]),
    pesos_banda(Nombre, Caracteristicas, T).

pesos_banda(Banda):-
    \+ banda(Banda),
    writeallnl(["'", Banda, "' no parece ser una Banda existente."]).

pesos_banda(Banda):-
    bandas(Banda, Z),
    findall(X, genero(X), Generos),
    pesos_banda(Banda, Z, Generos).

```

```
% Muestra todas las bandas de un género
buscar_bandas(Genero):-
    \+ genero(Genero),
    writeallnl(["",Genero,"" no parece ser un sub-género válido.]).

buscar_bandas(Genero):-
    findall(X, dame_genero_banda(X, Genero), Z),
    len(Z, Y),
    Y \= 0,
    writeallnl(["Listado de bandas pertenecientes al género ", Genero,
"en la base de conocimientos:"]),
    lista(Z).

buscar_bandas(_):-
    writenl("Sin coincidencias").
```

Pesos.pl

```
:-encoding(utf8).

% Librería de predicados relacionados con los pesos de los
% distintos elementos que engloban el mundo del metal.
%
% En esta base de conocimientos las bandas están compuestas
% de características que las defines. Los pesos permiten relacionar
% los géneros y las características con un valor numérico -el peso-.
%
% Cuanto mayor sea el peso, más afín será esa característica para
% un determinado género.
%
% Esto permite, una vez tenemos una banda, medir las características
% intrínsecas de la misma y conseguir un árbol de pesos donde uno
% de los géneros tendrá mayor puntuación que el resto, definiendo
% así la banda.
%
% Cuanto mayor sea un peso, más influyente es ese elemento
% para definir a una banda. Determinados elementos pueden
% tener un peso negativo, definiendo así restricciones para
% las bandas. P.ej.: "una banda de black metal jamás tendría
% líricas badadas en el cristianismo".
%
% Cabe destacar que en el mundo del metal hay muchos subgéneros que
% son muy similares a sus géneros padres, variando únicamente en
% un par de características. En la base de conocimientos esto será
% representado copiando los pesos del género padre y añadiendo las
% diferencias.
%
% Un peso inexistente para un género tendrá un valor de 0,
% (ver predicado pesos(_,_,0) más abajo)

% Los pesos se agrupan tal que peso(Genero,Caracteristica,Valor)
% P.ej.: si asumimos que la lírica basada en mitología nórdica
% debería otorgar un peso de 90 hacia el "viking metal" podríamos
% definirlo conceptualmente como:
% peso(viking_metal, lirica_mitologia_nordica, 90)
:-dynamic(pesos/3).

% Añade un peso a nuestros hechos dinámicos, eliminando
% cualquiera anterior que pudiese existir para ese
```



```

% género y característica
anyadir_peso(Genero, Caracteristica, Peso):-
    genero(Genero),
    caracteristica(Caracteristica),
    !,
    (retract(pesos(Genero, Caracteristica, _));true),
    assertz(pesos(Genero, Caracteristica, Peso)).

% Función recursiva para añadir varios elementos con un mismo peso
% a un mismo género. Usado por conveniencia en estos ficheros
% para facilitar la lectura y comprensión del código.
anyadir_pesos(_, []).
anyadir_pesos(Genero, [[Caracteristica, Peso]|T]):-
    anyadir_peso(Genero, Caracteristica, Peso),
    anyadir_pesos(Genero, T).

% Usado para copiar los pesos asociados a un género en otro.
% Esto se usa para crear los subgéneros casi iguales de los
% géneros padres, o para inicializar géneros parecidos.
% En la cabecera del fichero se explica esta aproximación.
copiar_pesos(_, _, []).

copiar_pesos(GeneroDesde, GeneroHasta, [Caracteristica|T]):-
    pesos(GeneroDesde, Caracteristica, Valor),
    anyadir_peso(GeneroHasta, Caracteristica, Valor),
    copiar_pesos(GeneroDesde, GeneroHasta, T).

copiar_pesos(GeneroDesde, GeneroHasta, [_|T]):-
    copiar_pesos(GeneroDesde, GeneroHasta, T).

copiar_pesos(GeneroDesde, GeneroHasta):-
    findall(X, caracteristica(X), Z),
    copiar_pesos(GeneroDesde, GeneroHasta, Z).

% Dado una banda y un género, devuelve el peso total asociado a ese
género
dame_peso_genero_banda([], _, A, A).
dame_peso_genero_banda([C|T], Genero, Acc, Total):-
    pesos(Genero, C, Peso),
    Peso \= 0,
    !,
    AccNew is Acc + Peso,
    dame_peso_genero_banda(T, Genero, AccNew, Total).
.
dame_peso_genero_banda([], T, Genero, Acc, Total):-
    dame_peso_genero_banda(T, Genero, Acc, Total).
.
dame_peso_genero_banda(Banda, Genero, Total):-
    dame_peso_genero_banda(Banda, Genero, 0, Total).

% dame_genero_banda es un predicado que, dado una banda, nos devolverá
cual es el
% género al que pertenece, entendiendo como esto, el género cuyo peso
sea
% más alto después de calcular todas las diferentes características de
la
% banda.
%
% El predicado se basará en recursión y será metamórfico para ofrecer
% al usuario un wrapper que no requiera especificar todos los
parámetros

```

```
% y permita pregunta en base a una banda

% El primer predicado es el caso base de la recursión
dame_genero_banda(_, [], _, A, A).

% El segundo predicado extrae la cabeza de la lista de géneros y
verifica
% el acumulado contra el peso de ese género. Si es superior, guarda el
% nombre del género y continúa iterando.
%
% El corte de este predicado es un "corte rojo" y, sin él, no
funcionaría
% correctamente.
dame_genero_banda(Banda, [Genero|T], AccPeso, _, Resultado):-
    dame_peso_genero_banda(Banda, Genero, PesoGenero),
    PesoGenero > AccPeso,
    !,
    dame_genero_banda(Banda, T, PesoGenero, Genero, Resultado).

% Este predicado se da cuando el anterior falla, es decir, cuando
% el peso del género que estamos calculando es inferior al ya
% calculado. La única finalidad de éste es continuar la recursividad.
dame_genero_banda(Banda, [_|T], AccPeso, AccGenero, Resultado):-
    dame_genero_banda(Banda, T, AccPeso, AccGenero, Resultado).

% Este es un predicado/2 que toma como parámetros el nombre de una
banda
% ya definida con anterioridad y devuelve el género al que pertenece.
dame_genero_caracteristicas([Caracteristica|T], Genero):-
    findall(X, genero(X), ListaGeneros),
    dame_genero_banda([Caracteristica|T], ListaGeneros, 0, 0, Genero).

% Este es un predicado/2 que toma como parámetros el nombre de una
banda
% ya definida con anterioridad y devuelve el género al que pertenece.
dame_genero_banda(Banda, Genero):-
    findall(X, genero(X), ListaGeneros),
    bandas(Banda, Z),

    dame_genero_banda(Z, ListaGeneros, 0, 0, Genero).
```

Titulos.pl

```
:-encoding(utf8).

% Este fichero define reglas y predicados para generar títulos de
forma
% aleatoria valiéndonos de las capacidades de prolog para generar
reglas
% lingüísticas.
%
% Los títulos estarán basadas en átomos (verbos, adjetivos, etc.) que
dependerán
% del sub género de metal.
%
% Puesto que la escena de este género de musica es mayoritariamente
% anglosajona los títulos se generarán en inglés para otorgar
% mayor credibilidad.

% Definimos varias constantes
```

```

titulo.          % Un título completo
sustantivo.      % Un sustantivo
nombre.         % Un nombre propio p.ej.: "Lemmy Killmeister"
adjetivo.        % Adjetivo
articulo.        % Un artículo
preposicion.     % Una preposición
conjuncion.      % Una conjunción

% Predicado dinámico que guardará los diccionarios
% Los argumentos son GeneroMetal, TipoPalabra y palabra
% p.ej.: black_metal, adjetivo, 'Unholy'
:-dynamic(diccionario/3).

% Predicado para añadir varias palabras al diccionario
% Caso base:
anyadir_diccionario(_, _, []).

% Recursión:
anyadir_diccionario(Genero, TipoPalabra, [H|T]):-
    % Añadimos
    anyadir_diccionario(Genero, TipoPalabra, H),

    % Recursión...
    anyadir_diccionario(Genero, TipoPalabra, T).

% Caso individual que añade:
anyadir_diccionario(Genero, TipoPalabra, Texto):-
    genero(Genero),
    palabra(TipoPalabra),
    assert(diccionario(Genero, TipoPalabra, Texto)).

% Determina si algo es considerado una palabra.
% Una palabra es algo de lo siguiente:
% - Un sustantivo
% - Un adjetivo
% - Un delimitador
% La forma más rápida de comprobarlo es sencillamente ver
% si está en el diccionario
palabra(X):-

    contiene([adjetivo, sustantivo, preposicion, conjuncion, articulo, nombre],
X).

tipo_palabra(Genero, X, Z):-
    diccionario(Genero, X, Z);
    diccionario(metal, X, Z).

% Genera un nombre de banda aleatorio para un género basándonos
% en los diccionarios relacionados con el mismo

% Reglas para generación de título
% Puesto que hay varias combinaciones separamos los predicados

% Un adjetivo seguido de un sustantivo.
titulo(Genero, T):-
    tipo_palabra(Genero, adjetivo, A),
    tipo_palabra(Genero, sustantivo, S),
    T = [A, S].

% Artículo seguido de sustantivo

```

```

titulo(Genero, T):-
    tipo_palabra(Genero, articulo, A),
    tipo_palabra(Genero, sustantivo, S),
    T = [A,S].

% Adjetivo + palabra + sustantivo
%titulo(Genero,T):-
%    tipo_palabra(Genero, adjetivo, A1),
%    tipo_palabra(Genero, articulo, A2),
%    tipo_palabra(Genero, sustantivo, S),
%    T = [A1,A2,S].

% Sustantivo + conjuncion + sustantivo
titulo(Genero,T):-
    tipo_palabra(Genero, sustantivo, S1),
    tipo_palabra(Genero, conjuncion, C),
    tipo_palabra(Genero, sustantivo, S2),
    S1\=S2,
    T = [S1,C,S2].

% Sustantivo + preposicion + sustantivo
titulo(Genero,T):-
    tipo_palabra(Genero, sustantivo, S),
    tipo_palabra(Genero, preposicion, C),
    tipo_palabra(Genero, sustantivo, A),
    S\=A,
    T = [S, C, A].

% Posesivo (nombre's nombre)
titulo(Genero,T):-
    tipo_palabra(Genero, sustantivo, S),
    tipo_palabra(Genero, sustantivo, S2),
    S \= S2,
    string_concat(S, "'s", S1),
    T = [S1, S2].

% Posesivo (nombre's adjetivo nombre)
%titulo(Genero,T):-
%    tipo_palabra(Genero, sustantivo, S),
%    tipo_palabra(Genero, sustantivo, S2),
%    tipo_palabra(Genero, adjetivo, A),
%    S \= S2,
%    string_concat(S, "'s", S1),
%    T = [S1, A, S2].

% Devuelve un título aleatorio generado dinámicamente
% para un género dado, se garantiza que el título no existirá
% en la lista de bandas
generar_nombre(Genero,T):-
    findall(X, titulo(Genero, X), ListaTitulos),
    elemento_aleatorio(ListaTitulos, Z),
    \+ banda(Z),
    implode(Z, ' ', T).

:-encoding(utf8).

% Diccionario de palabras, adjetivos, etc. disponible para la
generación de títulos
/*

```

```

anyadir_diccionario(Genero, articulo , []).
anyadir_diccionario(Genero, preposicion , []).

anyadir_diccionario(Genero, sustantivo , []).
anyadir_diccionario(Genero, nombre , []).
anyadir_diccionario(Genero, adjetivo , []).

*/

% Estos serán siempre iguales, por eso los ponemos en "metal"
:-anyadir_diccionario(metal, preposicion, [of]).
:-anyadir_diccionario(metal, articulo , [the]).
:-anyadir_diccionario(metal, conjuncion , [and]).

% Heavy metal (también los genéricos)
:-anyadir_diccionario(heavy_metal, sustantivo, [
    aero, angel, witch, anvil, armaggedon,
    axe, battle, machine,
    earth, thunder, lightning, metal, grave,
    gun, reaper, heaven, load, tide, claw, lord, maiden,
    priest, lord, king, rider, steel, forge, motor,
    night
]).
:-anyadir_diccionario(heavy_metal, adjetivo, [
    burning, crimson, oud, neverending, noisy, roaring,
    impetuous, red, black, crimson, atomic
]).

% Thrash metal
:-anyadir_diccionario(thrash_metal, sustantivo , [
    reign, dust, crush, annihilator, anthrax,
    destruction, epidemic, bomb, missile,
    sphere, hammer, angel, majesty, death,
    sacrifice, head
]).
:-anyadir_diccionario(thrash_metal, adjetivo , [
    destruction, savage, necro, deadly, nuclear,
    possessed, enraged, gamma
]).

% Black metal
:-anyadir_diccionario(black_metal, sustantivo , [
    angel, demon, blood, raven, satan, christ,
    asmodeus, judas, frost, winter,
    throne, fortress, goat, hell, dammnation,
    heresy
]).
:-anyadir_diccionario(black_metal, adjetivo , [
    unholy, black, dark, demonic, bloody, filthy,
    carpathian, rotted
]).

% Symphonic black metal
:-anyadir_diccionario(symphonic_black_metal, sustantivo , [
    majesty, lord, king, demon, succubi,
    cradle
]).
:-anyadir_diccionario(symphonic_black_metal, adjetivo , [
    unholy, unsacred, daemonic, burning, frozen,
    lusting
]).

```

```
% Unblack metal
:-anyadir_diccionario(unblack_metal, sustantivo , [
    angel, christ, church, pope, prayers,
    inquisition, god
]).
:-anyadir_diccionario(unblack_metal, adjetivo , [
    ruthless, brutal, sacred, fiery
]).

% Christian metal
:-anyadir_diccionario(cristian_metal, sustantivo , [
    christ, jesus, god, church, pope
]).
:-anyadir_diccionario(cristian_metal, adjetivo , [
    holy, immortal, faithful, eternal
]).

% Death metal
:-anyadir_diccionario(death_metal, sustantivo , [
    flesh, sorrow, torment, alchemist, corpse,
    doom, enemy, atrocity, autopsy, massacre,
    gut, ghoul, pestilence, knight, suffocation
]).
:-anyadir_diccionario(death_metal, adjetivo , [
    acidic, afflicted, amoral, armoured, destroyer,
    deadly, broken, crushed, rotten, cephalic
]).

% Technical death metal
:-anyadir_diccionario(technical_death_metal, sustantivo , [
    flesh, sorrow, torment, alchemist, corpse,
    doom, enemy, atrocity, autopsy, massacre,
    gut, ghoul, pestilence, knight, suffocation
]).
:-anyadir_diccionario(technical_death_metal, adjetivo , [
    acidic, afflicted, amoral, armoured, destroyer,
    deadly, broken, crushed, rotten, cephalic
]).

% Melodic death metal
:-anyadir_diccionario(melodic_death_metal, sustantivo , [
    flesh, sorrow, torment, alchemist, corpse,
    doom, enemy, atrocity, autopsy, massacre,
    gut, ghoul, pestilence, knight, suffocation
]).
:-anyadir_diccionario(melodic_death_metal, adjetivo , [
    acidic, afflicted, amoral, armoured, destroyer,
    deadly, broken, crushed, rotten, cephalic
]).

% Grindcore (estos van a ser explícitos)
:-anyadir_diccionario(grindcore, sustantivo , [
    pig, fetus, truth, remains, corpse, pain,
    nosebleed, abortion, 'death wound', cunt,
    flesh, god, cacophony
]).
:-anyadir_diccionario(grindcore, adjetivo , [
    aborted, agoraphobic, anal, thrower, driller, cephalic,
    bloody, disembowed, damaged, dying, destroyer,
    requirgitated
```

```
)).  
  
% Progressive metal  
:-anyadir_diccionario(progressive_metal, sustantivo , [  
    forever, animals, sevenfold, baron, baroness,  
    illusion, mage, sorrow, moon, rythm, edge  
]).  
:-anyadir_diccionario(progressive_metal, adjetivo , [  
    distorted, dream, iced, galactic, fiery, stolen,  
    orphaned  
]).  
  
% Folk metal  
:-anyadir_diccionario(folk_metal, sustantivo , [  
    frost, land, wizard, legend, tale,  
    kingdom, hymn  
]).  
:-anyadir_diccionario(folk_metal, adjetivo , [  
    celtic, gallaeic, galician, "köörpä", "kivimetsän",  
    chthonic  
]).  
  
% Viking metal  
:-anyadir_diccionario(viking_metal, sustantivo , [  
    battle, legend, warriors, maidens, wisdom, rites  
]).  
:-anyadir_diccionario(viking_metal, adjetivo , [  
    odin, thor, loki, tyr, valhalla, asgard,  
    ragnarok  
]).  
  
% Power metal  
:-anyadir_diccionario(power_metal, sustantivo , [  
    dragon, sword, kingdom, sauron, fantasy,  
    guardian, watcher, beholder, orc, fire,  
    flame, princess, knight  
]).  
:-anyadir_diccionario(power_metal, adjetivo , [  
    blind, burning, black, dark, divine, epic,  
    legendary, annointed, forgotten  
]).  
  
% Doom metal  
:-anyadir_diccionario(doom_metal, sustantivo , [  
    funeral, tendencies, fate, bride, maiden,  
    catharsis, candle, misery, despair, suicide  
]).  
:-anyadir_diccionario(doom_metal, adjetivo , [  
    suicidal, dying, mournful, terrified, dying,  
    silent  
]).  
  
% Nu metal  
:-anyadir_diccionario(nu_metal, sustantivo , [  
    rhymes, rythms, park, moves, stones, fists,  
    theory, monkey, town, chamber, cell, pool,  
    factory  
]).
```

```

:-anyadir_diccionario(nu_metal, adjetivo , [
    busting, shouting, raging, edgy, dry, vanilla,
    alien, apex
])).

% Industrial metal
:-anyadir_diccionario(industrial_metal, sustantivo , [
    factory, sun, riot, robot, chain, gear
])).
:-anyadir_diccionario(industrial_metal, adjetivo , [
    automated, noisy, mechanical, alien, exploding,
    electric
])).

```

Titulos_datos.pl

```

:-encoding(utf8).

% Diccionario de palabras, adjetivos, etc. disponible para la
generación de títulos
/*
anyadir_diccionario(Genero, articulo , []).
anyadir_diccionario(Genero, preposicion , []).

anyadir_diccionario(Genero, sustantivo , []).
anyadir_diccionario(Genero, nombre , []).
anyadir_diccionario(Genero, adjetivo , []).

*/

% Estos serán siempre iguales, por eso los ponemos en "metal"
:-anyadir_diccionario(metal, preposicion, [of]).
:-anyadir_diccionario(metal, articulo , [the]).
:-anyadir_diccionario(metal, conjuncion , [and]).

% Heavy metal (también los genéricos)
:-anyadir_diccionario(heavy_metal, sustantivo, [
    aero, angel, witch, anvil, armaggedon,
    axe, battle, machine,
    earth, thunder, lightning, metal, grave,
    gun, reaper, heaven, load, tide, claw, lord, maiden,
    priest, lord, king, rider, steel, forge, motor,
    night
])).
:-anyadir_diccionario(heavy_metal, adjetivo, [
    burning, crimson, oud, neverending, noisy, roaring,
    impetuous, red, black, crimson, atomic
])).

% Thrash metal
:-anyadir_diccionario(thrash_metal, sustantivo , [
    reign, dust, crush, annihilator, anthrax,
    destruction, epidemic, bomb, missile,
    sphere, hammer, angel, majesty, death,
    sacrifice, head
])).
:-anyadir_diccionario(thrash_metal, adjetivo , [
    destruction, savage, necro, deadly, nuclear,
    possessed, enraged, gamma
])).

```



```
% Black metal
:-anyadir_diccionario(black_metal, sustantivo , [
    angel, demon, blood, raven, satan, christ,
    asmodeus, judas, frost, winter,
    throne, fortress, goat, hell, dammnation,
    heresy
]).
:-anyadir_diccionario(black_metal, adjetivo , [
    unholy, black, dark, demonic, bloody, filthy,
    carpathian, rotted
]).

% Symphonic black metal
:-anyadir_diccionario(symphonic_black_metal, sustantivo , [
    majesty, lord, king, demon, succubi,
    cradle
]).
:-anyadir_diccionario(symphonic_black_metal, adjetivo , [
    unholy, unsacred, daemonic, burning, frozen,
    lusting
]).

% Unblack metal
:-anyadir_diccionario(unblack_metal, sustantivo , [
    angel, christ, church, pope, prayers,
    inquisition, god
]).
:-anyadir_diccionario(unblack_metal, adjetivo , [
    ruthless, brutal, sacred, fiery
]).

% Christian metal
:-anyadir_diccionario(cristian_metal, sustantivo , [
    christ, jesus, god, church, pope
]).
:-anyadir_diccionario(cristian_metal, adjetivo , [
    holy, immortal, faithful, eternal
]).

% Death metal
:-anyadir_diccionario(death_metal, sustantivo , [
    flesh, sorrow, torment, alchemist, corpse,
    doom, enemy, atrocity, autopsy, massacre,
    gut, ghoul, pestilence, knight, suffocation
]).
:-anyadir_diccionario(death_metal, adjetivo , [
    acidic, afflicted, amoral, armoured, destroyer,
    deadly, broken, crushed, rotten, cephalic
]).

% Technical death metal
:-anyadir_diccionario(technical_death_metal, sustantivo , [
    flesh, sorrow, torment, alchemist, corpse,
    doom, enemy, atrocity, autopsy, massacre,
    gut, ghoul, pestilence, knight, suffocation
]).
:-anyadir_diccionario(technical_death_metal, adjetivo , [
    acidic, afflicted, amoral, armoured, destroyer,
    deadly, broken, crushed, rotten, cephalic
]).
```

```
% Melodic death metal
:-anyadir_diccionario(melodic_death_metal, sustantivo , [
    flesh, sorrow, torment, alchemist, corpse,
    doom, enemy, atrocity, autopsy, massacre,
    gut, ghoul, pestilence, knight, suffocation
]).
:-anyadir_diccionario(melodic_death_metal, adjetivo , [
    acidic, afflicted, amoral, armoured, destroyer,
    deadly, broken, crushed, rotten, cephalic
]).

% Grindcore (estos van a ser explícitos)
:-anyadir_diccionario(grindcore, sustantivo , [
    pig, fetus, truth, remains, corpse, pain,
    nosebleed, abortion, 'death wound', cunt,
    flesh, god, cacophony
]).
:-anyadir_diccionario(grindcore, adjetivo , [
    aborted, agoraphobic, anal, thrower, driller, cephalic,
    bloody, disembowed, damaged, dying, destroyer,
    regurgitated
]).

% Progressive metal
:-anyadir_diccionario(progressive_metal, sustantivo , [
    forever, animals, sevenfold, baron, baroness,
    illusion, mage, sorrow, moon, rythm, edge
]).
:-anyadir_diccionario(progressive_metal, adjetivo , [
    distorted, dream, iced, galactic, fiery, stolen,
    orphaned
]).

% Folk metal
:-anyadir_diccionario(folk_metal, sustantivo , [
    frost, land, wizard, legend, tale,
    kingdom, hymn
]).
:-anyadir_diccionario(folk_metal, adjetivo , [
    celtic, gallaeic, galician, "köörpi", "kivimetsän",
    chthonic
]).

% Viking metal
:-anyadir_diccionario(viking_metal, sustantivo , [
    battle, legend, warriors, maidens, wisdom, rites
]).
:-anyadir_diccionario(viking_metal, adjetivo , [
    odin, thor, loki, tyr, valhalla, asgard,
    ragnarok
]).

% Power metal
:-anyadir_diccionario(power_metal, sustantivo , [
    dragon, sword, kingdom, sauron, fantasy,
    guardian, watcher, beholder, orc, fire,
    flame, princess, knight
]).
```

```
: -anyadir_diccionario(power_metal, adjetivo , [
    blind, burning, black, dark, divine, epic,
    legendary, annointed, forgotten
]).

% Doom metal
: -anyadir_diccionario(doom_metal, sustantivo , [
    funeral, tendencies, fate, bride, maiden,
    catharsis, candle, misery, despair, suicide
]).
: -anyadir_diccionario(doom_metal, adjetivo , [
    suicidal, dying, mournful, terrified, dying,
    silent
]).

% Nu metal
: -anyadir_diccionario(nu_metal, sustantivo , [
    rhymes, rythms, park, moves, stones, fists,
    theory, monkey, town, chamber, cell, pool,
    factory
]).
: -anyadir_diccionario(nu_metal, adjetivo , [
    busting, shouting, raging, edgy, dry, vanilla,
    alien, apex
]).

% Industrial metal
: -anyadir_diccionario(industrial_metal, sustantivo , [
    factory, sun, riot, robot, chain, gear
]).
: -anyadir_diccionario(industrial_metal, adjetivo , [
    automated, noisy, mechanical, alien, exploding,
    electric
]).
```

Caso de prueba

Inicio de la aplicación

El único punto de inicio de la aplicación es el archivo *inicio.pl*. Al ejecutarlo, sin necesidad de teclear nada, la aplicación nos recibirá con una pantalla de inicio (que podremos mostrar de nuevo escribiendo el predicado *ayuda.*).

```
Base de conocimientos del metal cargada y lista para ser usada.
Los siguientes predicados pueden usarse para explorar la base de
conocimientos:

===== AYUDA =====
- glosario                : para una breve explicación del glosario
usado en la base de conocimientos.

===== CONSULTAS GENERALES =====
- mostrar_generos         : para ver los géneros con los que trabaja
la herramienta.
- mostrar_caracteristicas : para ver las características que pueden
tener las bandas.
- mostrar_bandas         : para ver que bandas de metal existen en
la aplicación.

===== INFORMACIÓN =====
- info_banda (X)          : muestra información de la banda pasada
como argumento.
- pesos_banda (X)         : muestra los pesos de una banda para todos
los distintos sub-géneros.

===== CREACIÓN / ELIMINACIÓN =====
- generar_banda (X)       : crea una banda de características
aleatorias y le asigna el nombre X.
- generar_banda_aleatoria : crea una banda de título y
características aleatorias.
- borrar_banda (X)        : elimina la banda de nombre X de la base
de conocimientos.

===== MISCELÁNEA =====
- buscar_bandas (Genero)  : muestra todas las bandas de la base de
conocimientos que sean de un género pasado como argumento.
- generar_nombre (Genero) : genera un nombre de banda, aceptando como
argumento un subgénero del metal para el que queremos generarla.
- ayuda                   : para volver a ver esta pantalla.

~ They can't stop us, let'em try ~
~ For Heavy Metal we will die! ~
```

Consulta de géneros existentes

Siguiendo las instrucciones que nos muestra la pantalla de ayuda, usamos el predicado *mostrar_generos.* para ver por pantalla los géneros que conoce la herramienta.

```
?- mostrar_generos.
Listado de géneros en la base de conocimientos:
- metal
- heavy_metal
- thrash_metal
- black_metal
- unblack_metal
```

```

- symphonic_black_metal
- cristian_metal
- death_metal
- technical_death_metal
- melodic_death_metal
- grindcore
- progressive_metal
- folk_metal
- viking_metal
- power_metal
- doom_metal
- nu_metal
- industrial_metal
true

```

El predicado **genero(X)**. nos ofrece una forma alternativa de explorar los géneros.

```

?- genero(X).
X = metal ;
X = heavy_metal ;
X = thrash_metal ;
X = black_metal ;
X = unblack_metal ;
X = symphonic_black_metal ;
X = cristian_metal ;
X = death_metal ;
X = technical_death_metal ;
X = melodic_death_metal ;
X = grindcore ;
X = progressive_metal ;
X = folk_metal ;
X = viking_metal ;
X = power_metal ;
X = doom_metal ;
X = nu_metal ;
X = industrial_metal.

```

Consulta de características

De igual forma que los géneros, podemos mostrar las características que la aplicación considera válidas con el predicado **mostrar_características**.

Recordemos que las características son las peculiaridades intrínsecas y diferenciadoras de cada banda.

```

?- mostrar_caracteristicas.
Listado de características en la base de coocimientos:
- lirica_metal
- lirica_religion
- lirica_anti_religion
- lirica_muerte
- lirica_horror
- lirica_guerra
- lirica_historica
- lirica_cristianismo
- lirica_fantasia
- lirica_filosofia
- lirica_espiritualidad

```

```

- lirica_folk_y_mitologia
- lirica_mitologia_nordica
- lirica_muerte_luto_y_desesperacion
- lirica_sangre_y_visceras
- lirica_sociedad
- lirica_introspeccion
- lirica_otra
- formacion_clasica
- formacion_teclado
- formacion_folk
- formacion_orquestal
- formacion_dj
- formacion_sintetizador
- formacion_alternativa
- tempo_30bpm
- tempo_60bpm
- tempo_90bpm
- tempo_120bpm
- tempo_160bpm
- tempo_200bpm
- tempo_240bpm
- tempo_superior
- tempo_variante
- voz_limpia
- voz_aguda
- voz_grave
- voz_gritos_melodicos
- voz_gritos
- voz_gutural
- voz_gutural_limpia
- voz_aullidos
- voz_death_growl
- voz_otros
- tecnica_solos_largos
- tecnica_musica_tecnica
- tecnica_double_kick_drumming
- tecnica_blast_beat
- tecnica_tremolo_riff
- tecnica_fuerte_distorsion
- tecnica_shred_guitar
- tecnica_tapping
- tecnica_liricas_complejas
- tecnica_canciones_melodicas
- tecnica_voz_rimas
true.

```

El predicado **característica(X)** consigue lo mismo. Por la longitud de su salida por pantalla, éste no se ilustra.

Consulta de bandas

Aunque se pueden crear nuevas bandas, la base de conocimientos viene con conocimiento de una serie de las mismas predefinidas basadas en sus homónimos de la vida real.

El predicado **mostrar_bandas**. Permite ver, en todo momento, las bandas que conoce la aplicación, sean predefinidas o generadas dinámicamente.

```

?- mostrar_bandas.
Listado de bandas en la base de conocimientos:
- iron_maiden
- judas_priest

```

```

- slayer
- metallica
- satyricon
- emperor
- dimmu_borgir
- antestor
- amon_amarth
- dethklok
- gojira
- rhapsody
- blind_guardian
- bloodgod
- carcass
- at_the_gates
- bolt_thrower
- holocausto_canibal
- amorphis
- turisas
- finsterforst
- katatonia
- korn
- victor_love
true.

```

El predicado **banda(X)** puede usarse para obtener la misma salida, o comprobar si una banda existe.

Mostrar la información de una banda

El predicado **info_banda(X)**. permite obtener la información detallada de todas las bandas (si especificamos una variable) o de una banda en concreto (si pasamos un nombre de banda. La información mostrada es el nombre, características de la banda y el género calculado de la misma acorde a los pesos de sus características.

```

?- info_banda(slayer).
=== Información de la banda ' slayer ' ===
- Nombre: slayer
- Género: thrash_metal

Lista de características de la banda:
- formacion_clasica
- lirica_sociedad
- tempo_160bpm
- voz_gritos_melodicos
- tecnica_fuerte_distorsion
true .

?- info_banda("banda_inexistente").
La banda banda_inexistente no existe en la base de conocimientos
true .

```

Mostrar los pesos de una banda

El predicado **pesos_banda(X)** nos permitirá ver los pesos de cada banda para los distintos subgéneros. Recordemos que el peso más alto es el que define el sub género de la banda. **X** puede ser el nombre de la banda a comprobar o una variable, en cuyo caso se mostrarán todos los pesos para todas las bandas.

El predicado no permite mostrar los pesos de bandas inexistentes.

```

?- pesos_banda(dimmu_borgir).

```

```

- dimmu_borgir tiene un peso de 0 para el género metal
- dimmu_borgir tiene un peso de -21 para el género heavy_metal
- dimmu_borgir tiene un peso de -17 para el género thrash_metal
- dimmu_borgir tiene un peso de 24 para el género black_metal
- dimmu_borgir tiene un peso de -81 para el género unblack_metal
- dimmu_borgir tiene un peso de 36 para el género
symphonic_black_metal
- dimmu_borgir tiene un peso de -94 para el género cristian_metal
- dimmu_borgir tiene un peso de 1 para el género death_metal
- dimmu_borgir tiene un peso de 12 para el género
technical_death_metal
- dimmu_borgir tiene un peso de 23 para el género
melodic_death_metal
- dimmu_borgir tiene un peso de -29 para el género grindcore
- dimmu_borgir tiene un peso de -14 para el género
progressive_metal
- dimmu_borgir tiene un peso de -14 para el género folk_metal
- dimmu_borgir tiene un peso de -14 para el género viking_metal
- dimmu_borgir tiene un peso de 1 para el género power_metal
- dimmu_borgir tiene un peso de -12 para el género doom_metal
- dimmu_borgir tiene un peso de -16 para el género nu_metal
- dimmu_borgir tiene un peso de -16 para el género
industrial_metal
true .

?- pesos_banda(inexistente).
' inexistente ' no parece ser una Banda existente.
true .

```

Nótese en el ejemplo como una misma banda puede tener características de varios géneros.

En este caso *dimmu_borgir* tiene elementos característicos de los géneros *melodic_death_metal*, *technical_death_metal*, *black_metal* y *symphonic_black_metal*, siendo éste último el que tiene el peso mayor y define el género de la banda.

Generación de una banda con título especificado y características procedurales

La herramienta permite crear bandas dinámicamente de forma procedural tomando como parámetro un nombre. La banda creada será guardada en el conjunto de bandas de la aplicación. Para ello podemos usar el predicado **generar_banda(X)**, donde **X** es el nombre a asignar.

El predicado no permite crear bandas que ya existen en la base de conocimientos.

En el caso de prueba siguiente generamos una banda llamada *banda de prueba 1*. Usando el predicado **info_banda(X)**, podemos comprobar como el sistema le asignó características aleatorias que la clasifican como una banda del género *death_metal*.

```

?- generar_banda("banda de prueba 1").
Se ha generado la banda ' banda de prueba 1 '
true .

?- info_banda("banda de prueba 1").
=== Información de la banda ' banda de prueba 1 ' ===
- Nombre:  banda de prueba 1
- Género:  death_metal

Lista de características de la banda:
- formacion clasica

```



```
- lirica_anti_religion
- voz_death_growl
- tecnica_blast_beat
- tempo_120bpm
true
```

Tras ejecutar este predicado podemos comprobar -usando el predicado **mostrar_bandas**- que ésta se ha añadido a la base de conocimientos en la última posición:

```
?- mostrar_bandas.
Listado de bandas en la base de conocimientos:
- iron_maiden
- judas_priest
- slayer
- metallica
- satyricon
- emperor
- dimmu_borgir
- antestor
- amon_amarth
- dethklok
- gojira
- rhapsody
- blind_guardian
- bloodgod
- carcass
- at_the_gates
- bolt_thrower
- holocausto_canibal
- amorphis
- turisas
- finsterforst
- katatonia
- korn
- victor_love
- banda de prueba 1
true.
```

Generación de una banda con título y características procedurales

El predicado **generar_banda_aleatoria**. crea una banda con características y nombre procedural, usando para ella las reglas lingüísticas definidas en la base de conocimientos para cada género.

Primero el sistema genera las características, acorde a ellas calcula su género y posteriormente genera un título válido para ese género.

A continuación, mostramos como creamos dos bandas que se añaden a la base de conocimientos.

```
?- generar_banda_aleatoria.
Se ha generado aleatoriamente la siguiente banda:

crushed doom ( technical_death_metal )

Sus principales características son las siguientes:
- formacion_alternativa
- lirica_religion
```

```

- voz_gutural_limpia
- tecnica_solos_largos
- tempo_120bpm
true

?- generar_banda_aleatoria.
Se ha generado aleatoriamente la siguiente banda:

maiden of armaggedon ( heavy_metal )

Sus principales características son las siguientes:
- formacion_alternativa
- lirica_metal
- voz_gritos_melodicos
- tecnica_canciones_melodicas
- tempo_90bpm
true

```

El predicado **mostrar_bandas**. nos permite corroborar que ambas se han añadido a la base de conocimientos.

```

?- mostrar_bandas.
Listado de bandas en la base de conocimientos:
- iron_maiden
- judas_priest
- slayer
- metallica
- satyricon
- emperor
- dimmu_borgir
- antestor
- amon_amarth
- dethklok
- gojira
- rhapsody
- blind_guardian
- bloodgod
- carcass
- at_the_gates
- bolt_thrower
- holocausto_canibal
- amorphis
- turisas
- finsterforst
- katatonia
- korn
- victor_love
- banda de prueba 1
- crushed doom
- maiden of armaggedon
true.

```

Podemos comprobar que las características y géneros de las bandas persisten usando el predicado **info_banda(X)**.

```

?- info_banda("maiden of armaggedon").
=== Información de la banda 'maiden of armaggedon' ===
- Nombre: maiden of armaggedon
- Género: heavy_metal

Lista de características de la banda:

```

```
- formacion_alternativa
- lirica_metal
- voz_gritos_melodicos
- tecnica_canciones_melodicas
- tempo_90bpm
true
```

Eliminación de bandas de la base de conocimientos

El predicado ***borrar_banda(X)***. nos permite eliminar la banda de nombre ***X*** de la base de conocimientos. Si ***X*** no se especifica, se irán eliminando bandas de forma secuencial en cada iteración. El predicado no permite borrar bandas inexistentes.

Nótese en la prueba como borramos la banda *Maiden of armaggedon* dos veces, resultando la segunda en un mensaje de que ésta ya no existe.

```
?- borrar_banda("maiden of armaggedon").
true .

?- borrar_banda("maiden of armaggedon").
No existe ninguna banda con el nombre ' maiden of armaggedon '
true .
```

Podemos corroborar que está eliminada usando el predicado ***mostrar_bandas***.

```
?- mostrar_bandas.
Listado de bandas en la base de conocimientos:
- iron_maiden
- judas_priest
- slayer
- metallica
- satyricon
- emperor
- dimmu_borgir
- antestor
- amon_amarth
- dethklok
- gojira
- rhapsody
- blind_guardian
- bloodgod
- carcass
- at_the_gates
- bolt_thrower
- holocausto_canibal
- amorphis
- turisas
- finsterforst
- katatonia
- korn
- victor_love
- banda de prueba 1
- crushed doom
true.
```

Buscar bandas por género

El predicado ***buscar_bandas(X)***. permite mostrar por pantalla todas las bandas existentes en la base de conocimientos para un género dado. Si el género no existe, se mostrará un error.

Nótese que la prueba muestra una de las bandas creadas en ejemplos anteriores (*crushed doom*) en el subgénero que le corresponde (*technical_death_metal*).

```
?- buscar_bandas(technical_death_metal).  
Listado de bandas pertenecientes al género technical_death_metal en  
la base de conocimientos:  
- dethklok  
- gojira  
- crushed doom  
true .  
  
?- buscar_bandas(black_metal).  
Listado de bandas pertenecientes al género black_metal en la base de  
conocimientos:  
- satyricon  
- emperor  
true .  
  
?- buscar_bandas(genero_inexistente).  
'genero_inexistente' no parece ser un sub-género válido.  
true .
```

Generación de nombres para subgéneros

El predicado **generar_nombre(X)**. nos permite generar un nombre arbitrario para un sub género especificado en el argumento **X**.

Nótese que la base de conocimientos asigna títulos diferentes a los distintos títulos, como se puede ver a continuación.

El predicado no permite recibir géneros inexistentes.

```
?- generar_nombre(X).  
atrocitiy and knight  
true .  
  
?- generar_nombre(black_metal).  
heresy's demon  
true .  
  
?- generar_nombre(black_metal).  
judas's raven  
true .  
  
?- generar_nombre(black_metal).  
angel's raven  
true .  
  
?- generar_nombre(heavy_metal).  
noisy lightning  
true .  
  
?- generar_nombre(cristian_metal).  
faithful church  
true .  
  
?- generar_nombre(nu_metal).  
dry town  
true .  
  
?- generar_nombre(nu_metal).
```

```

town of monkey
true .

?- generar_nombre(viking_metal).
wisdom and battle
true

?- generar_nombre(viking_metal).
ragnarok warriors
true .

?- generar_nombre(power_metal).
the guardian
true .

?- generar_nombre(power_metal).
princess and fantasy
true .

?- generar_nombre(inexistente).
' inexistente ' no parece ser un sub-género válido
true .

```

Comprobación de todos los géneros y bandas

El predicado **dame_genero_banda(X, Y)**. permite obtener el género (variable **Y**) de una banda pasada como parámetro (**X**).

También puede ejecutarse sin argumentos (mostrará todas las bandas con sus subgéneros) o solo con el argumento del subgénero (mostrará todas las bandas de ese subgénero).

A continuación, se muestran las distintas ejecuciones.

```

?- dame_genero_banda(iron_maiden,Y) .
Y = heavy_metal .

?- dame_genero_banda(X,heavy_metal) .
X = iron_maiden ;
X = judas_priest .

?- dame_genero_banda(X,Y) .
X = iron_maiden,
Y = heavy_metal ;
X = judas_priest,
Y = heavy_metal ;
X = slayer,
Y = thrash_metal ;
X = metallica,
Y = thrash_metal ;
X = satyricon,
Y = black_metal ;
X = emperor,
Y = black_metal ;
X = dimmu_borgir,
Y = symphonic_black_metal ;
X = antestor,
Y = unblack_metal ;
X = amon_amarth,
Y = melodic_death_metal ;

```

```
X = dethklok,  
Y = technical_death_metal ;  
X = gojira,  
Y = technical_death_metal ;  
X = rhapsody,  
Y = power_metal ;  
X = blind_guardian,  
Y = power_metal ;  
X = bloodgod,  
Y = cristian_metal ;  
X = carcass,  
Y = melodic_death_metal ;  
X = at_the_gates,  
Y = melodic_death_metal ;  
X = bolt_thrower,  
Y = grindcore ;  
X = holocausto_canibal,  
Y = grindcore ;  
X = amorphis,  
Y = progressive_metal ;  
X = turisas,  
Y = folk_metal ;  
X = finsterforst,  
Y = viking_metal ;  
X = katatonia,  
Y = doom_metal ;  
X = korn,  
Y = nu_metal ;  
X = victor_love,  
Y = industrial_metal ;  
X = "banda de prueba 1",  
Y = death_metal ;  
X = "crushed doom",  
Y = technical_death_metal ;  
false.
```

Dificultades encontradas

La principal dificultad encontrada en la actividad fue sin duda el mero hecho de aprender a pensar en términos de un lenguaje lógico.

Acostumbrado a un lenguaje procedimental, en donde el programador ha de especificar claramente qué quiere hacer, el hecho de tener que definir una base de conocimientos por la que prolog sería capaz de moverse *sin más* me fue algo ajeno.

Mis primeros intentos de crear una base de conocimientos intentaban, sin éxito, crear métodos y funciones con los conceptos de *pasar parámetros*, *recibir valores de salida*, *iterar sobre elementos*, etc. cuando realmente debería saber que las variables aquí no existen, solo los conjuntos de hechos y reglas existentes.

Cuando por fin comprendí que una creación de regla bastaba para que prolog *supiese* lo que es un concepto, entendí mejor el paradigma lógico y, como consecuencia, fui capaz de comenzar a crear la base de conocimientos.

En comparación, otras dificultades fueron la de la implementación en sí y el desarrollo de la base de conocimientos. La primera venía dada porque desconocía el lenguaje y la segunda porque no sabía cómo implementar una base de conocimientos de estas características, donde

los subgéneros vienen definidos por un conjunto de características que pueden ser heterogéneas, no por conjuntos fijos dados. No obstante, estas dos dificultades no fueron tan complicadas de superar como la inicial.

