

Query Sparql:

Le query sono state realizzate come se fossero realizzate da un assistente virtuale in cerca di risposte alle domande fornite in chat dall'utente.

1) **Esiste un cantante rap che ha vinto l' oscar? Se si, quale?**

Seleziona dall'ontologia tutti gli individui che appartengono al genere rap che hanno ricevuto un oscar e ne restituisce i nomi.

```
PREFIX : <http://www.semanticweb.org/sorre/ontologies/2020/1/Musicista#>
```

```
SELECT ?rapper  
WHERE{  
    ?rapper :hasGenre :Rap  
    FILTER EXISTS { ?rapper :receive :Oscar .}  
}
```

rapper

Eminem

2) **Quali cantanti hanno collaborato con Eminem?**

Seleziona tutte le entità che appartengono alla classe "singer" che sono collegate ad Eminem tramite la relazione di collaborazione.

```
PREFIX : <http://www.semanticweb.org/sorre/ontologies/2020/1/Musicista#>
```

```
SELECT ?Singer  
WHERE {  
    ?Singer :collaborateWith :Eminem  
}
```

Singer

Rihanna

3) In che traccia hanno collaborato Eminem e Rihanna?

Seleziona le entità della classe "Songe" in che sono state create da entrambi gli artisti sovracitati

```
PREFIX : <http://www.semanticweb.org/sorre/ontologies/2020/1/Musicista#>
```

```
SELECT ?song  
WHERE {  
  ?song :madeBy :Eminem.  
  ?song :madeBy :Rihanna  
}
```

song
LoveTheWayYouLie

4) Di che album fa parte quella canzone?

Seleziona l'album in cui questa particolare canzone è contenuta

```
PREFIX : <http://www.semanticweb.org/sorre/ontologies/2020/1/Musicista#>
```

```
SELECT ?album  
WHERE {  
  :LoveTheWayYouLie :isContainedIn ?album  
}
```

album
Recovery

5) In che anno è uscito Recovery?

Richiede l'anno di pubblicazione di questo album particolare album

PREFIX : <http://www.semanticweb.org/sorre/ontologies/2020/1/Musicista#>

```
SELECT ?year
WHERE {
  :Recovery :yearsOfPublication ?year
}
```

year

"2010"^{^^}<http://www.w3.org/2001/XMLSchema#integer>

6) Elenco di tutti gli album di Eminem con rispettiva data di uscita?

Elenca tutti gli album pubblicati dall'artista, con rispettivo anno di pubblicazione e ordinati cronologicamente.

PREFIX : <http://www.semanticweb.org/sorre/ontologies/2020/1/Musicista#>

```
SELECT ?album ?year
WHERE
{
  :Eminem :releaseAlbum ?album.
  ?album :yearsOfPublication ?year
}
order by asc(?year)
```

album	year
Infinite	"1996" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
TheSlimShadyLP	"1999" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
TheMarshallMathersLP	"2000" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
TheEminemShow	"2002" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
Encore	"2004" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
Relapse	"2009" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
Recovery	"2010" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
TheMarshallMathersLP2	"2013" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
Revival	"2017" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
Kamikaze	"2018" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>
MusicToBeMurderedBy	"2020" ^{^^} <http://www.w3.org/2001/XMLSchema#integer>

7) Quanti anni ha Eminem?

Restituisce l'età dell'entità Eminem

```
PREFIX : <http://www.semanticweb.org/sorre/ontologies/2020/1/Musicista#>
```

```
SELECT ?age
WHERE
{
  :Eminem :age ?age
}
```

age

"47"^^<http://www.w3.org/2001/XMLSchema#integer>

Lista di query in formato testuale:

PREFIX: <http://www.semanticweb.org/sorre/ontologies/2020/1/Musicista#>

```
1) SELECT ?rapper
WHERE {
  ?rapper :hasGenre :Rap
  FILTER EXISTS { ?rapper :receive :Oscar.}
}
```

```
2) SELECT ?Singer
WHERE {
  ?Singer :collaborateWith :Eminem
}
```

```
3) SELECT ?song
WHERE {
  ?song :madeBy :Eminem.
  ?song :madeBy :Rihanna
}
```

```
4) SELECT ?album
WHERE {
  :LoveTheWayYouLie :isContainedIn ?album
}
```

```
5) SELECT ?year
WHERE {
  :Recovery :yearsOfPublication ?year
}
```

```
6) SELECT ?album ?year
WHERE
{
  :Eminem :releaseAlbum ?album.
  ?album :yearsOfPublication ?year
}
order by asc(?year)
```

```
6) SELECT ?age
WHERE
{
  :Eminem :age ?age
}
```

Regole SWRL

1) **CreateRapSong**: A rapper write lyrics following the rithm of the beat
`mu:make(?Musicistas, ?Musicistam) -> mu:writeSong(?Musicistas, ?Musicistam)`

2) **Producer**: A Beatmaker is the artist who produce the beat
`mu:Artist(?Musicistaa) ^ mu:produce(?Musicistaa, ?Musicistab) -> mu:BeatMaker(?Musicistab)`

3) **RapSong**: A Track is composed by a lyrics and a sound
`mu:Track(?Musicistax) -> mu:Lyrics(?Musicistax) ^ mu:Beat(?Musicistax)`

4) **isRapper**: Rule that label as Rapper a singer who do rap
`mu:Singer(?x) ^ mu:hasGenre(?x, mu:Rap) -> mu:Rapper(?x)`

Operatore built in:

5) **Featuring**: A collaboration between two artists
`differentFrom(?art1, ?art2) ^ mu:publishSong(?art1, ?t) ^ mu:publishSong(?art2, ?t) -> mu:collaborateWith(?art1, ?art2)`