

PROTOKOLL v 0.9

Det här är första versionen av protokollet för databasen vid Hultsfredsprojektet. Vi har baserat protokollet på de krav vi har fått in från grupperna hittills. Läs och begrundna.

Protokollet är uppdelat i personer och artister, och för dessa finns det kommandon och attribut. Med kommandon kan man ställa förfrågningar från databasen utifrån specifika attribut. Man kan även förändra attributvärden och som det är nu får ALLA databasanvändare förändra ALLA attribut. Bra eller dåligt?

Tanken är att information om personer och artister hårdkodas in i databasen från start och att man sedan bara förändrar vissa attribut, t.ex. en person får ytterligare en favoritartist eller skaffar sig en ny vän.

OBS! Det krävs att någon specificerar vilka genrer som skall finnas. T.ex. SKA, ROCK, POP, SOUL m.m. och att någon specificerar i vilken ordning dessa anges för en persons musikpreferens. Det krävs även att någon specificerar vilka artister/grupper som finns "att välja på" och vad respektive artist har för måttvärde 1-100 för respektive genre.

Vi har inte fått några krav från simuleringsgruppen och skulle ytterligare behöva diskutera med dem för att se hur simuleringen av personer och deras positioner skall lösas.

En stjärna * innebär att antalet är valbart. Exempelvis antalet attribut.

För personattributen så är de i svart gemensamma för "The Game" och Visualiseringsgrupperna, attribut i **rött** är specifika för "The Game" och i **blått** är specifika för Visualiseringsgrupperna.

PERSON

PERSONKOMMANDON

PERSON_LIST <ATTR>=value[& <ATTR>=value]*

Förklaring: Returnerar samtliga personer det/de angivna attributet/attributen.

Returnerar: Lista med ett eller flera ID-nr: [id,...] eller SERVER_ERROR <ERROR_CODE>

Exempel: PERSON_LIST NAME='courtney'
>>[6332467816,8739016528,7290654189]
PERSON_LIST LENGTH=130 & PULSE>180
>>[7785209812]

PERSON_RAND <ATTR>=value[& <ATTR>=value]*

Förklaring: Returnerar en slumpmässig person med det/de angivna attributet/attributen.

Returnerar: Ett ID-nr: [id] eller SERVER_ERROR <ERROR_CODE>

Exempel: PERSON_RAND NAME='göran'
>>[2132334512]
PERSON_RAND LENGTH=130 & FAVOURITES=[hives]
>>[]

PERSON_GET <ATTR>[, <ATTR>]* FROM id

Förklaring: Returnerar värdet för det angivna attributet för personen med angivet idnr.

Returnerar: Ett/flera attributvärden: [value,...] eller SERVER_ERROR <ERROR_CODE>

Exempel: PERSON_GET NAME, AGE FROM 1234567891
>>['Bengt-Erik', 32]
PERSON_GET FAVOURITES, LENGTH FROM 4567891234
>>[['Metallica', 'Slayer', 'Dr. Alban'],183]

PERSON_SET <ATTR> TO value FOR id

Förklaring: Sätter värdet för det angivna attributet för personen med angivet idnr.

Returnerar: SUCCESS eller SERVER_ERROR <ERROR_CODE>

Exempel: PERSON_SET NAME TO 'nisse' FOR 1234567891
>>SUCCESS
PERSON_SET GENREPREFS TO [10,20,30,40] FOR 4567891234
>>SERVER_ERROR 119

PERSONATTRIBUT (<ATTR>)

ID

Förklaring: Personens ID-nr (10 siffror). Unikt.

Typ: int

Exempel: ID=1234560078

NAME

Förklaring: Namn på personen. Om efternamn önskas kan detta anges.

Typ: string

Exempel: NAME='nisse'
NAME='nisse andersson'

POSITION

Förklaring: Personens aktuella position.

Typ: tuple (x,y) med heltal: [int,int]

Exempel: POSITION=[5,89]

STATUS

Förklaring: Simulerad eller "verklig" aktör?

Typ: int, 0 för simulerad, 1 för "verklig"

Exempel: STATUS=1

AGE

Förklaring: Personens ålder.

Typ: int

Exempel: AGE=23

GENDER

Förklaring: Personens kön.

Typ: int, 0 för man, 1 för kvinna

Exempel: GENDER=1

HAIR

Förklaring: Personens hårfärg.

Typ: string

Exempel: HAIR='blonde'

GAMER

Förklaring: Med i "the game"?

Typ: int, 0 för inte med i "the game", 1 för med i "the game"

Exempel: GAMER=1

POINTS

Förklaring: Poäng i "the game", alltid noll poäng för aktörer med GAMER=0

Typ: int

Exempel: POINTS=129

RFID

Förklaring: ID-nr för den RFID-tag som personen har. Finns ej för simulerade.

Typ: string
Exempel: RFID='0101a84c8f'

LENGTH

Förklaring: Personens längd i cm.
Typ: heltal
Exempel: LENGTH=123

PULSE

Förklaring: Personens puls.
Typ: int
Exempel: PULSE=160
Kommentar: Simuleringsapplikationen ansvarar för att uppdatera pulsen.

FRIENDS

Förklaring: Personens vänner.
Typ: lista med int (id): [id,...]
Exempel: FRIENDS=[6332467816,8739016528,7290654189]
Kommentar: Kan vara en eller flera id (eller i värsta fall tomma listan).

GENREPREFS

Förklaring: Hur mycket personens musikpreferens stämmer överens med respektive genre. Hur många genres finns det? I vilken ordning?
Typ: lista med heltal: [int,int,int,int,...]
Exempel: GENREPREFS=[10,30,45,60,...]
Kommentar: Varje position i listan motsvarar en genre, dessa positionsmappningar måste användas konsekvent av alla grupper som använder GENREPREFS.

FAVOURITES

Förklaring: Personens favoritartist/favoritartister.
Typ: lista med strängar: [string,string,...]
Exempel: FAVOURITES=[donnas,metallica,hives,...]
Kommentar: Artister som sätts med PERSON_SET måste finnas med i databasen.

SCHEDULED (deprecated)

Förklaring: De/den artist/artister som personens planerar att se.
Typ: lista med strängar: [string,string,...]
Exempel: SCHEDULED=[donnas,hives,...]
Kommentar: Artister som sätts med PERSON_SET måste finnas med i databasen.

ARTIST

ARTISTKOMMANDON

ARTIST_LIST <ATTR>=value [& <ATTR>=value] *

Förklaring: Returnerar samtliga artister med det/de angivna attributet/attributen.

Returnerar: Lista med en eller flera artister: [artist,...] eller SERVER_ERROR <ERROR_CODE>

Exempel:
ARTIST_LIST GENRE_SKA>30
>>['skatalites','liberator','rancid']
ARTIST_LIST GENRE_POP>70 & GENRE_PUNK>20
>>['blink-182']

ARTIST_RAND <ATTR>=value [& <ATTR>=value] *

Förklaring: Returnerar en slumpmässig artist med det/de angivna attributet/attributen.

Returnerar: En artist: [artist] eller SERVER_ERROR <ERROR_CODE>

Exempel:
ARTIST_RAND GENRE_ROCK=30
>>['mando diao']
ARTIST_RAND GENRE_SOUL>50 & GENRE_JAZZ>60
>>['brand new heavies']

ARTIST_GET <ATTR>[, <ATTR>]* FROM artist

Förklaring: Returnerar värdet för angivet attribut och angiven artist

Returnerar: Ett attributvärde: [value] eller SERVER_ERROR <ERROR_CODE>

Exempel:
ARTIST_GET GENRE_ROCK FROM 'donnas'
>>[80]
ARTIST_GET INFO FROM 'fu manchu'
>>['örongodis för vuxna']

ARTIST_SET <ATTR> TO value FOR artist

Förklaring: Sätter värdet för angivet attribut och angiven artist

Returnerar: SUCCESS eller SERVER_ERROR <ERROR_CODE>

Exempel:
ARTIST_SET GENRE_HIPHOP=50 FOR 'fjärde världen'
>>SUCCESS
ARTIST_SET INFO='pop från götet' FOR 'håkan hellström'
>>SERVER_ERROR 110

ARTISTATTRIBUT

ARTIST

Förklaring: Artistens namn. Unikt.

Typ: string

Exempel: ARTIST='moneybrother'
ARTIST='kristofer åström'

GENRE_X

Förklaring: Artistens måttlighet för denna genre. Hur många genres X (SKA, ROCK, POP m.m.) finns det?

Typ: int

Exempel: GENRE_ELECTRONICA=10
GENRE_GARAGE=50

INFO

Förklaring: Kort information om artisten

Typ: string

Exempel: INFO='spelar skjortan av dig'

FELKODER (ERROR_CODE)

100

Förklaring: Syntaxfel i förfrågan

Fler felkoder kommer snart...