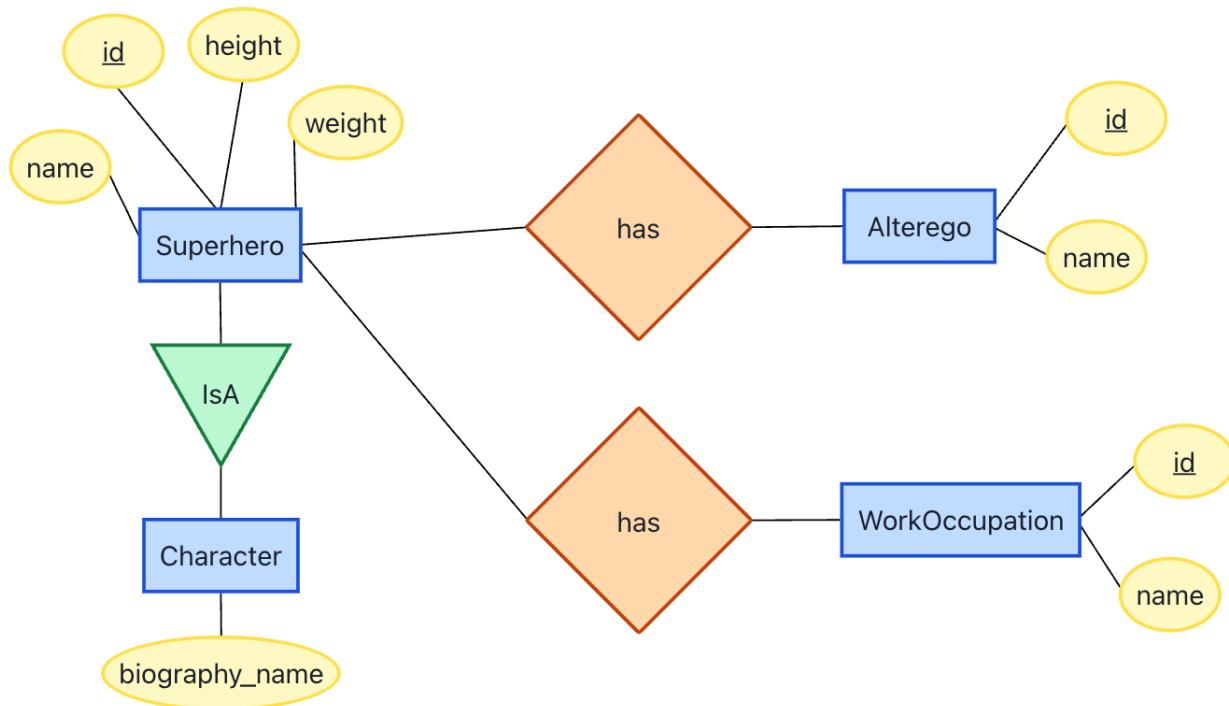


Laboratorio 5 - LDD y Acceso Programático

Considere el siguiente modelo entidad relacion que corresponde a una base de datos de superheroes:



En este laboratorio usted deberá crear las tablas necesarias correspondientes al modelo entidad relación y luego importar los datos a partir de un archivo csv.

La creación de tablas e importación deberá ser hecha en el servidor del curso en el esquema **superheroes**.

- Para conectarse al servidor directamente, usted puede usar una terminal en Linux/Mac/Windows o Putty en Windows. El servidor está hosteado en `cc3201.dcc.uchile.cl` y escucha en el puerto 240. Su usuario es `cc3201` y la contraseña se publicó en u-cursos en Material Docente. El comando que deberían ejecutar en la terminal para conectarse al servidor desde Linux/Mac es:

```
ssh -p 240 cc3201@cc3201.dcc.uchile.cl
```

Para conectarse a la base de datos basta con ejecutar el comando `psql cc3201`. Para cerrar la sesión utilice el comando `\q`.

- Para conectarse al servidor programáticamente, puede crear una conexión en python como se describe a continuación:

```
conn = psycopg2.connect(host="cc3201.dcc.uchile.cl",
    database="cc3201",
    user="cc3201",
    password="j'<3_cc3201", port="5440")
```

Usted deberá entregar un archivo de texto (`.txt`) y un archivo python (`.py`) con el desarrollo de los ejercicios que siguen.

- P1.** 30 PUNTOS Cree las tablas necesarias de modo que satisfagan el modelo entidad relación en el schema **superheroes**, agregando las restricciones de llave primaria y foránea según corresponda. Tenga en cuenta que el nombre de cada tabla debe tener un prefijo único para cada grupo, como por ejemplo **equiporocket_tablita**.

Hints:

- El campo id lo puede crear de tipo **serial**.
- Sugerimos traducir la herencia (isA) usando dos tablas (para **character** y **superhero**).

- P2.** 60 PUNTOS Crea y corra un archivo python para poblar la base de datos creada en el punto anterior a partir del archivo de datos csv provisto en `ucursos`. Este archivo tiene varios errores e inconsistencias, por lo que le sugerimos leer las siguientes consideraciones:

- Le recomendamos hacer la tarea en python. Para instalar la librería necesaria para conectarse a la base de datos utilice `pip3` ejecutando `pip3 install psycopg2-binary`
- Suponga un Superhero es Character si tiene datos en la columna `biography_full—name`.
- Algunos strings pueden venir con comillas dobles extras (e.g. `adventurer`, `"agent"`, `mechanic`)

Por lo tanto le sugerimos seguir los siguientes pasos.

- (a) Lea el archivo. Por cada personaje (fila):
- i. Inserte el superhero, obteniendo su id.
 - ii. Inserte el character usando el id del punto anterior.
 - iii. Para cada alter ego (asuma que están separados por `“,”` o `“;”`)
 - A. Elimine espacios blancos al comienzo y al final, y comillas dobles.
 - B. Busque si ya existe el alter ego para ese superhéroe. Si no existe, insertelo.
 - iv. Para cada ocupación/oficio (asuma que estan separadas por `“,”` o `“;”`)

- A. Elimine espacios blancos al comienzo y al final, y comillas dobles.
- B. Seleccione el id de la ocupación dado el nombre de esta. Si no existe, créala. Note que un mismo oficio puede venir en distintas combinaciones de mayúsculas y/o minúsculas en distintas filas.
- C. Busque si ya existe un elemento en la tabla intermedia entre superhéroe y ocupación. Si no existe insertela.

(b) Haga “commit” de sus cambios.

P3. 10 PUNTOS Entregue las consultas sql que responden las siguientes preguntas

- (a) **Pregunta:** Calcule los nombres y peso (en kilos) de los 10 superheroes más pesados.
- (b) **Pregunta:** Calcule los nombres, nombres biográficos, y altura (en cm) de los 10 characters más altos.
- (c) **Pregunta:** Calcule el nombre de los 3 trabajos más populares junto a la cantidad de superheroes asociados a ese trabajo.