



001

# MIGRACIONES & ASOCIACIONES

INGENIERÍA DE SOFTWARE



José Antonio Castro  
Felipe Fuentes

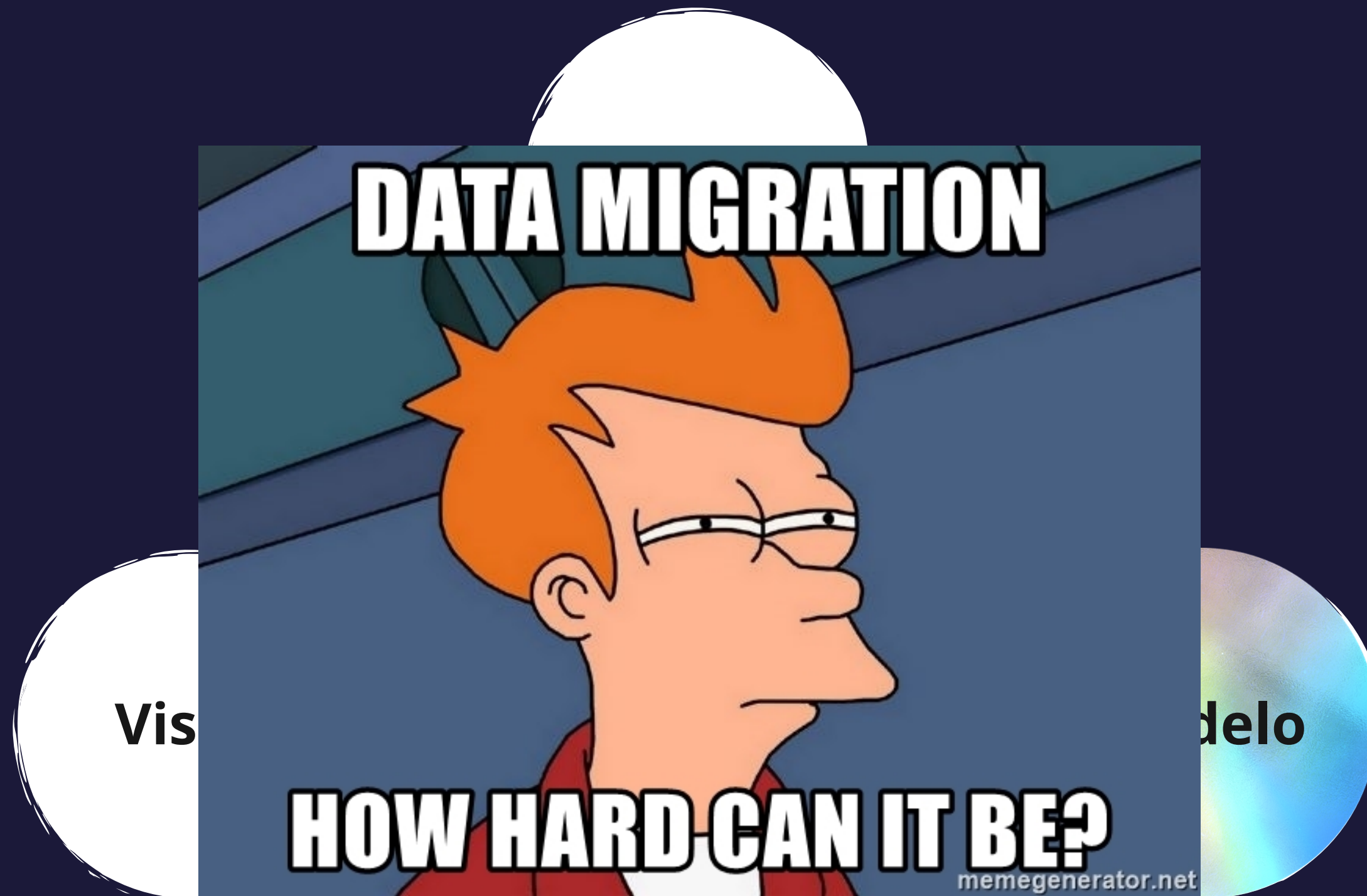
```
graph TD; Vista((Vista)) --> Controlador((Controlador)); Controlador --> Modelo((Modelo)); Modelo --> Vista;
```

**Controlador**

**Vista**

**Modelo**







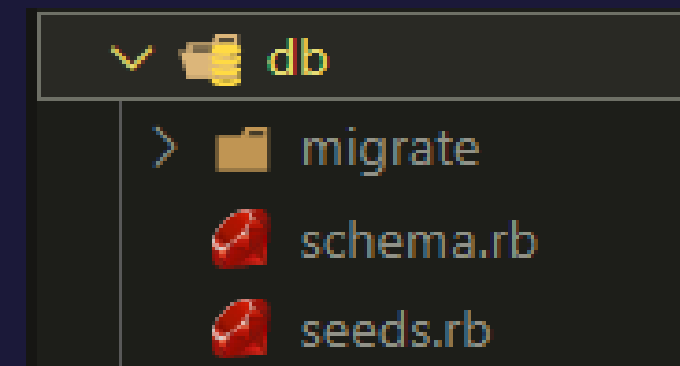
# MIGRACIONES

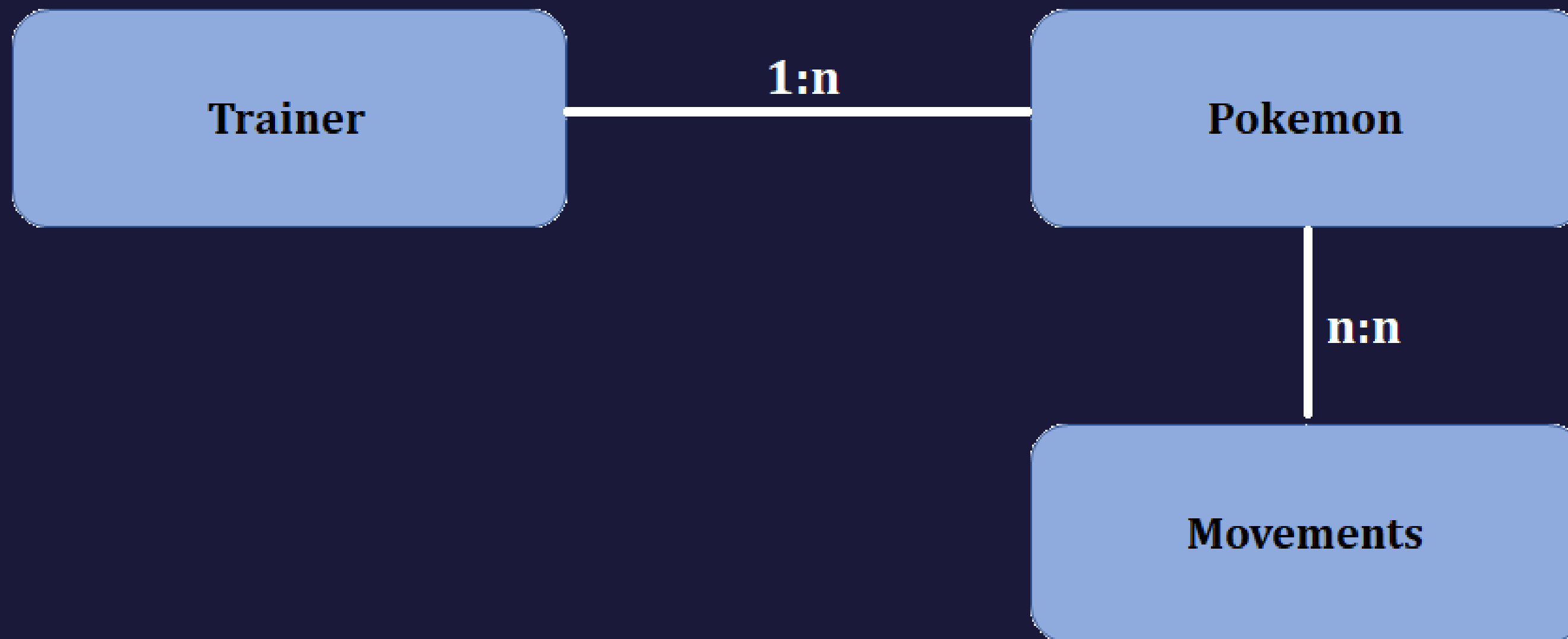
## ¿QUÉ SON?

- CÓDIGO QUE PERMITE CREAR Y ALTERAR LA BASE DE DATOS.

- COMANDOS QUE EVITAN OPERAR CON COMANDOS SQL.

- CREAN ARCHIVOS DE MIGRACIÓN EN "DB/MIGRATE".





# Crear un modelo



```
rails g model Nombre_modelo atributo1:tipo_atributo1 atributo2:tipo_atributo2
```

# Tipos de migración



# Crear

```
rails g model Pokemon name:string kind:string trainerID:integer
```

# Agregar

```
rails g migration AddStatsToPokemons attack:integer defense:integer speed:float
```

Nombre    Tabla en plural

# Eliminar

```
rails g migration RemoveSpeedFromPokemons speed:float
```



# ¿Qué es Rake?

ES UN CONSTRUCTOR DE RAIL QUE  
PERMITE MANJEAR EL MODELO

Desde Rails 5 en adelante se permite llamar a  
'rails' y que este incluya 'rake'

A terminal window with a dark background and three colored window control buttons (red, yellow, green) at the top left. It displays a list of rake tasks in a monospaced font, with the 'db:seed' task highlighted by a vertical cursor.


```
rake db:drop
rake db:create
rake db:migrate
rake db:rollback
rake db:seed
```

```
baelfire@LAPTOP-EGDM0IU0: ~$ rake -T
rake about # List versions...
rake active_storage:install # Copy over the...
rake app:template # Applies the t...
rake app:update # Update config...
rake assets:clean[keep] # Remove old co...
rake assets:clobber # Remove compil...
rake assets:environment # Load asset co...
rake assets:precompile # Compile all t...
rake cache_digests:dependencies # Lookup first-...
rake cache_digests:nested_dependencies # Lookup nested...
rake db:create # Creates the d...
rake db:drop # Drops the dat...
rake db:environment:set # Set the Envir...
rake db:fixtures:load # Loads fixture...
rake db:migrate # Migrate the d...
rake db:migrate:status # Display statu...
rake db:rollback # Rolls the sch...
rake db:schema:cache:clear # Clears a db/s...
rake db:schema:cache:dump # Creates a db/...
rake db:schema:dump # Creates a db/...
rake db:schema:load # Loads a schem...
rake db:seed # Loads the see...
rake db:setup # Creates the d...
rake db:structure:dump # Dumps the dat...
rake db:structure:load # Recreates the...
rake db:version # Retrieves the...
rake dev:cache # Toggle develo...
rake initializers # Print out all...
rake log:clear # Truncates all...
rake middleware # Prints out yo...
rake notes # Enumerate all...
rake notes:custom # Enumerate a c...
rake restart # Restart app b...
rake routes # Print out all...
rake secret # Generate a cr...
rake spec # Run all specs...
rake spec:helpers # Run the code ...
rake spec:models # Run the code ...
rake spec:requests # Run the code ...
rake spec:views # Run the code ...
rake stats # Report code s...
rake test # Runs all test...
rake test:db # Run tests qui...
rake test:system # Run system te...
rake time:zones[country_or_offset] # List all time...
```

# Seeds en Heroku



```
# en branch main
git add .
git commit -m "Commit a Heroku"
heroku login
git push heroku main
heroku pg:reset # resetea la base de datos de heroku (seguir instrucciones del comando)
heroku run rake db:migrate # hace las migraciones en heroku
heroku run rake db:seed # puebla la base de datos
```



# ASOCIACIONES ¿QUÉ SON?

- ES LA FORMA POR LA QUE  
HACEMOS LAS RELACIONES  
ENTRE LOS MODELOS DE  
ACTIVE RECORDS

# belongs\_to

```
class Book < ApplicationRecord
  belongs_to :author
end
```



```
class Book < ApplicationRecord
  belongs_to :author
end
```

# has\_one



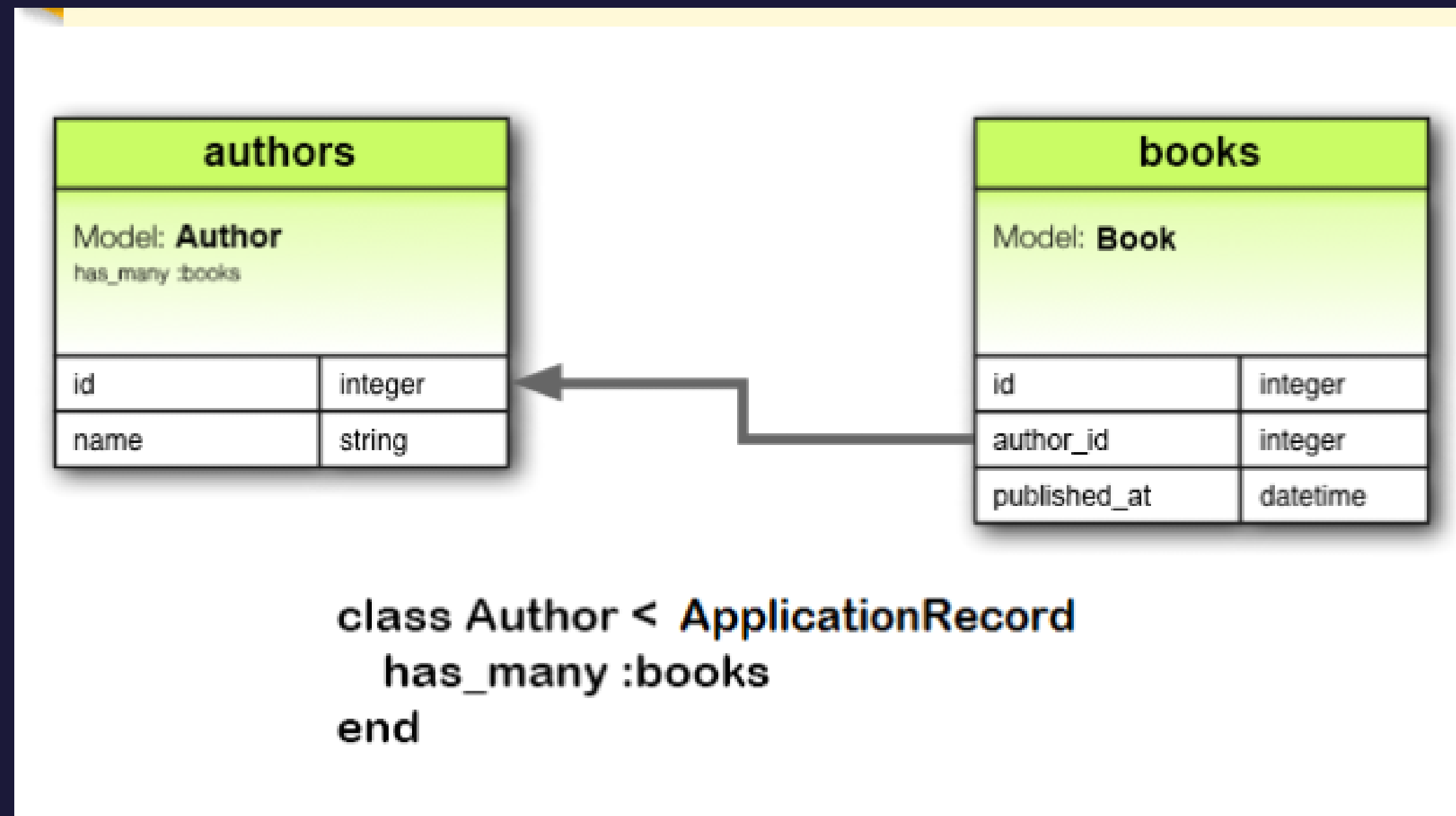
```
class Supplier < ApplicationRecord
  has_one :account
end
```

suppliers	
Model: <b>Supplier</b> has_one :account	
id	integer
name	string

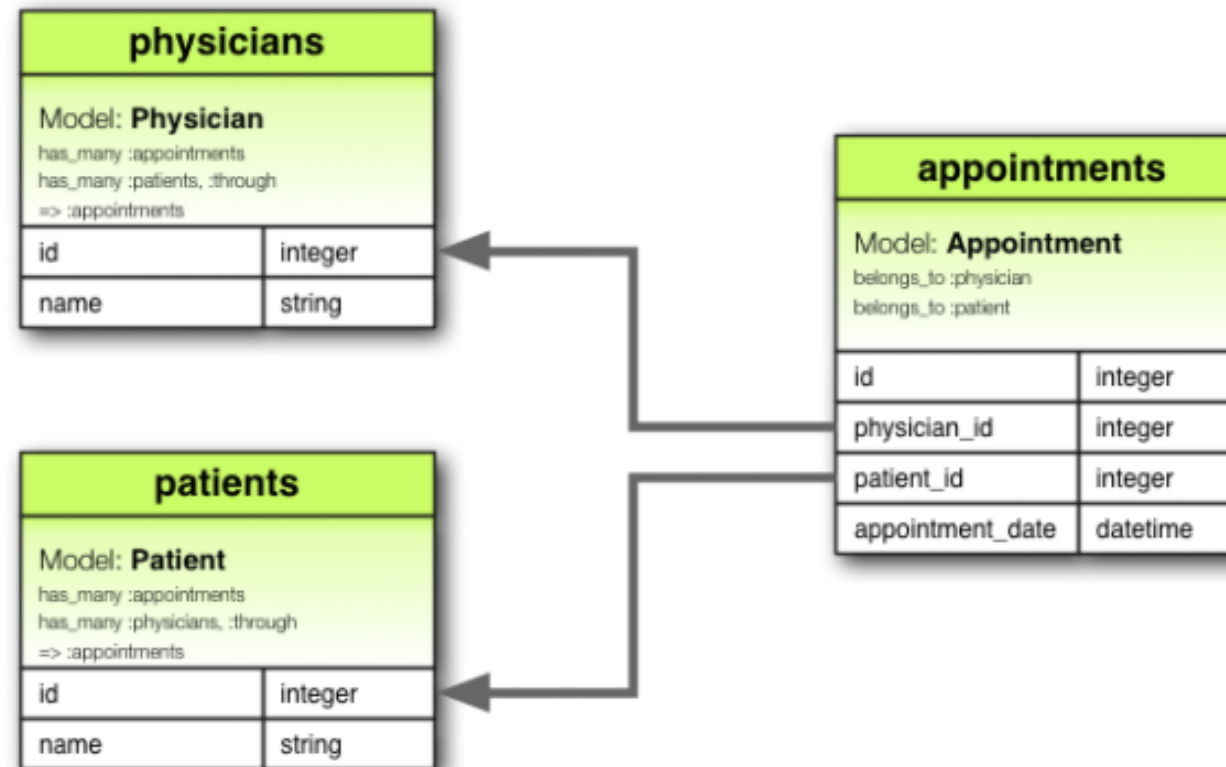
accounts	
Model: <b>Account</b>	
id	integer
supplier_id	integer
account_number	string

```
class Supplier < ApplicationRecord
  has_one :account
end
```

# has\_many



# has\_many: through

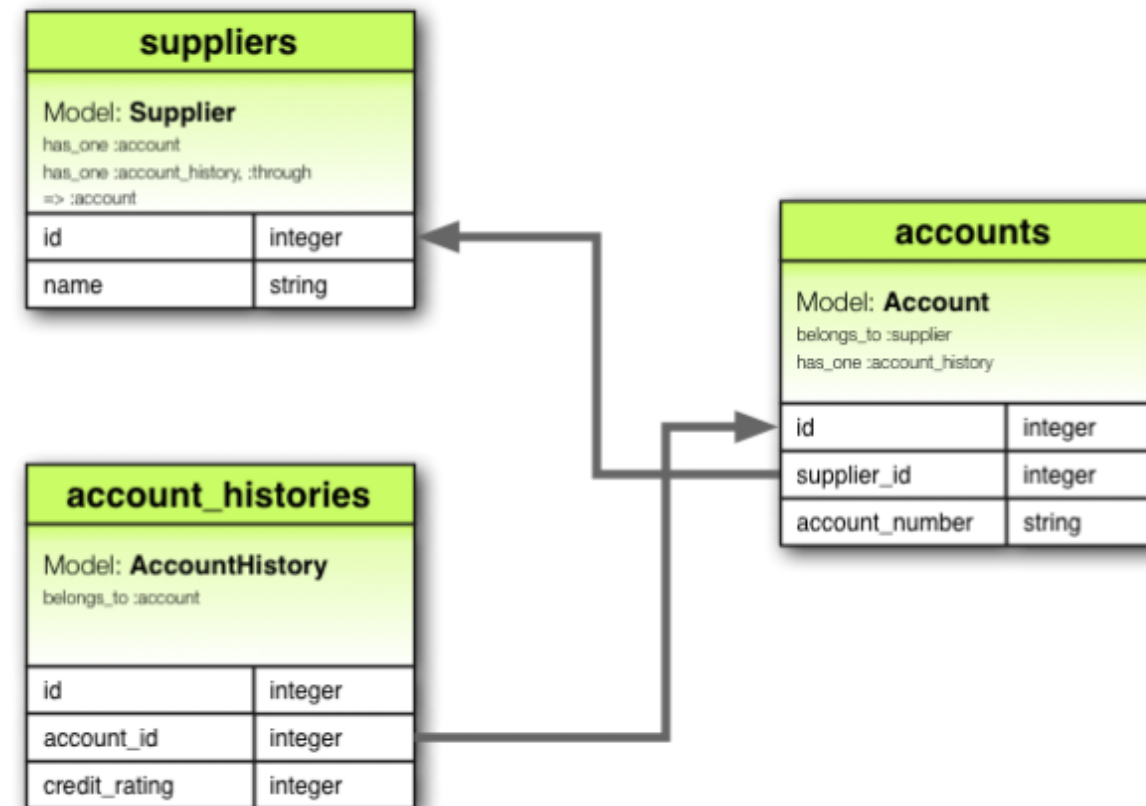


```
class Physician < ApplicationRecord
  has_many :appointments
  has_many :patients, :through => :appointments
end

class Appointment < ApplicationRecord
  belongs_to :physician
  belongs_to :patient
end

class Patient < ApplicationRecord
  has_many :appointments
  has_many :physicians, :through => :appointments
end
```

# has\_one: through



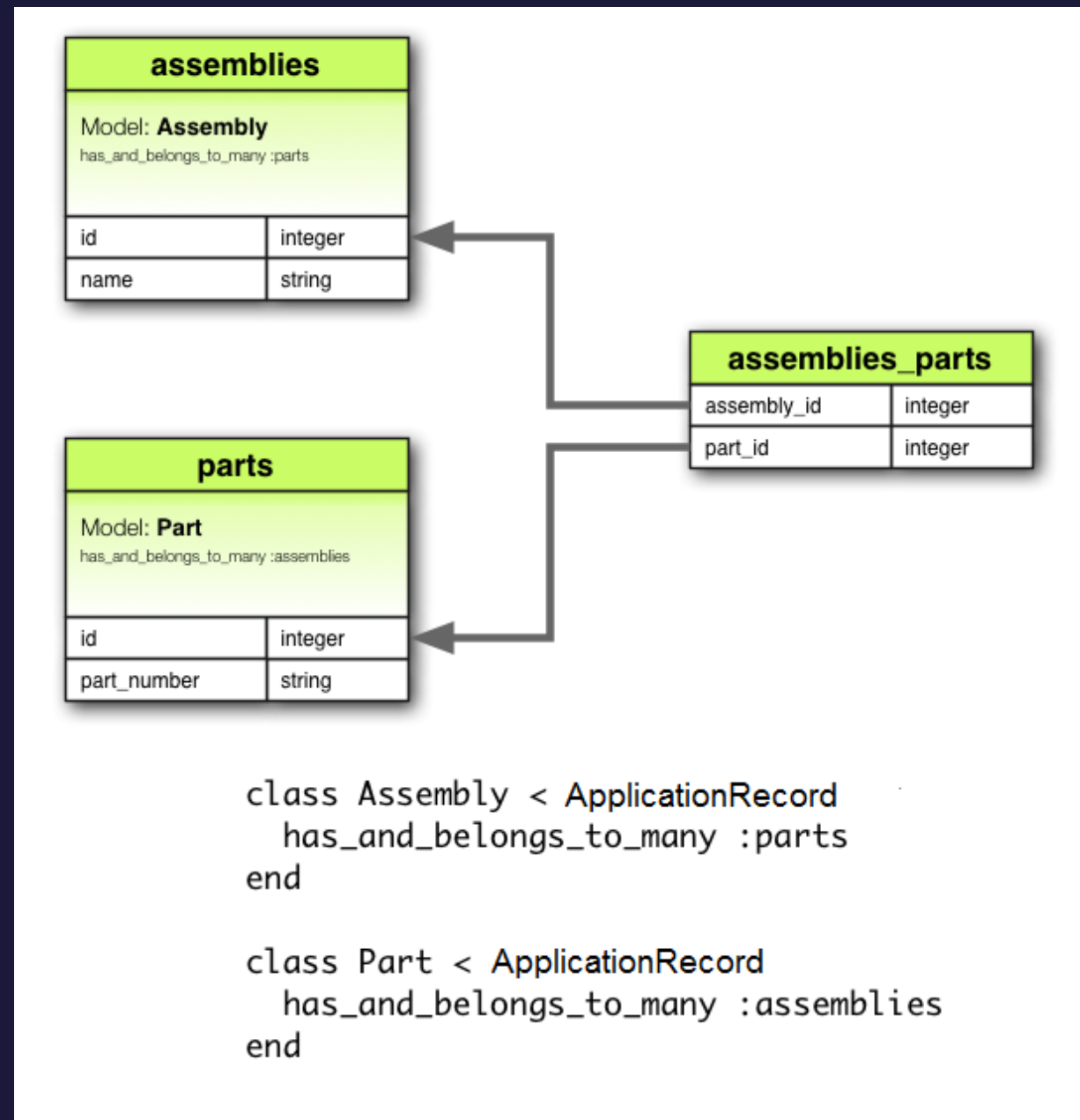
```
class Supplier < ApplicationRecord
  has_one :account
  has_one :account_history, :through => :account
end
```

```
class Account < ApplicationRecord
  belongs_to :supplier
  has_one :account_history
end
```

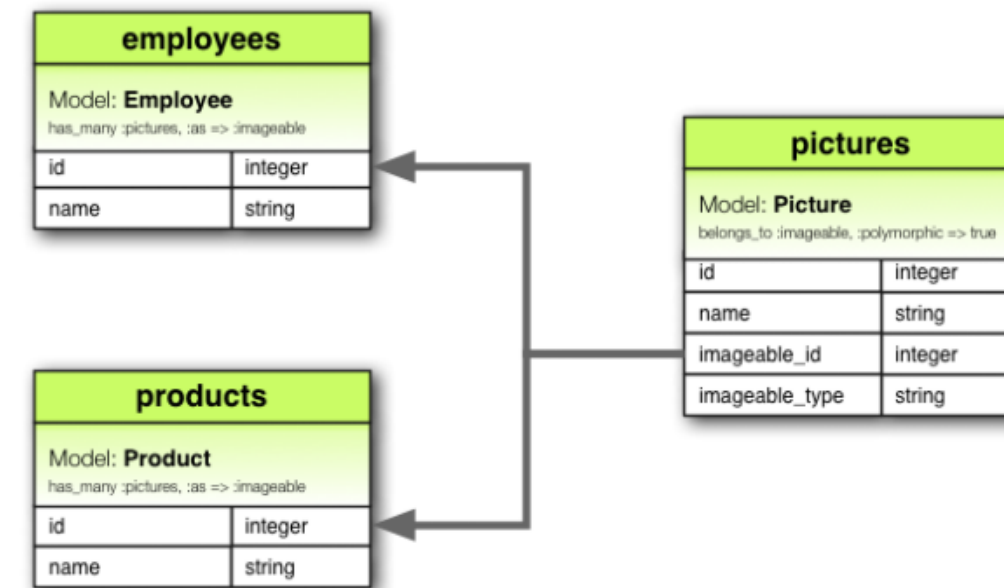
```
class AccountHistory < ApplicationRecord
  belongs_to :account
end
```



# has\_and\_belongs\_to\_many



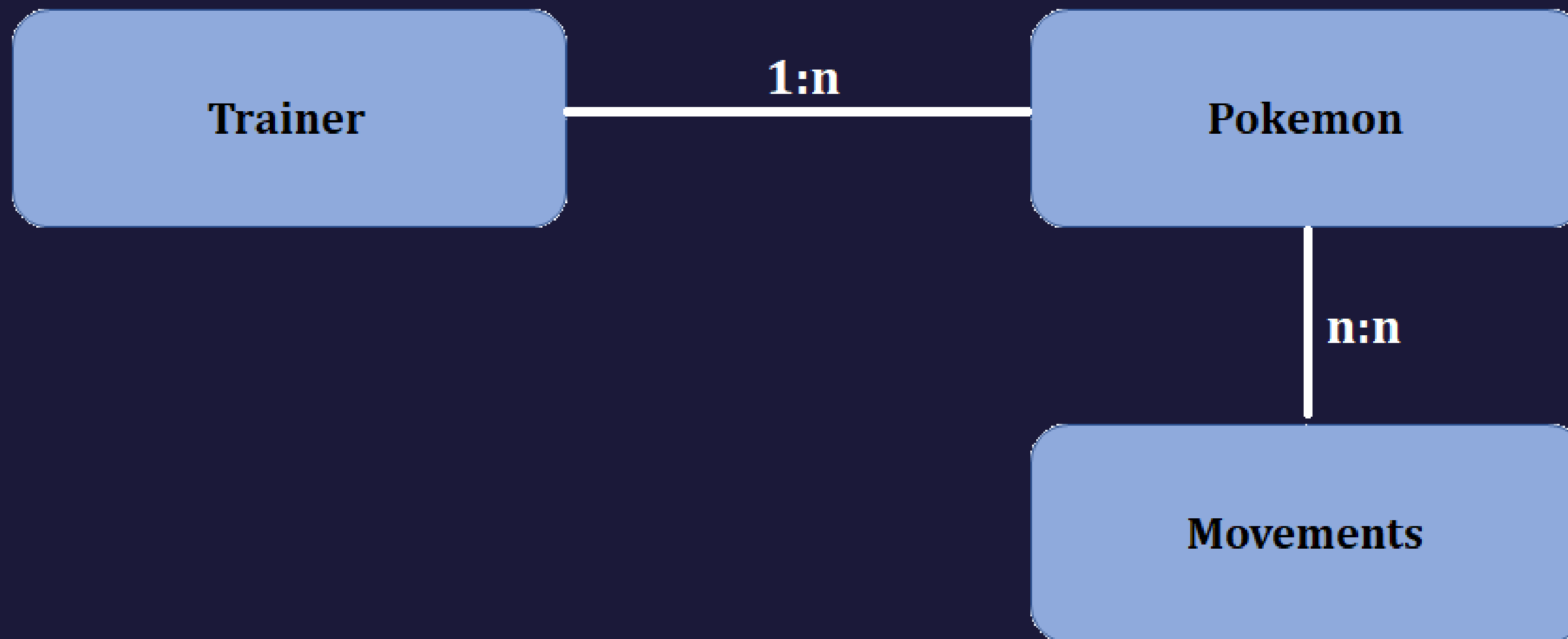
# polymorphic



```
class Picture < ApplicationRecord
  belongs_to :imageable, :polymorphic => true
end
```

```
class Employee < ApplicationRecord
  has_many :pictures, :as => :imageable
end
```

```
class Product < ApplicationRecord
  has_many :pictures, :as => :imageable
end
```



**¡Importante! al hacer asociaciones se debe crear la columna asociada a la key (id) del modelo al cual se le quiere hacer la asociación. (al usar belongs\_to)**



```
# Podemos crear el modelo con la asociación  
rails g model Pokemon name:string kind:string trainer:references  
  
# O podemos agregarsela posteriormente  
rails g migration AddStatsToPokemons attack:integer defense:integer speed:float
```

**has\_and\_belongs\_to\_many**

**rails g migration CreateJoinTablePokemonsMovimientos pokemons movimientos**



```
class Magazine < ApplicationRecord
  has_many :ads
end

class Ad < ApplicationRecord
  belongs_to :magazine
end
```

[Copy](#)

Nested routes allow you to capture this relationship in your routing. In this case, you could include this route declaration:



```
resources :magazines do
  resources :ads
end
```

[Copy](#)

In addition to the routes for magazines, this declaration will also route ads to an `AdsController`.

**Cuando entiendes migraciones y asociaciones**



**Jamás había sentido un poder como este**