Model and Database Relationships

Active Record:

- Represents models and their data
- Represents associations between these models
- Represents inheritance hierarchies through related models
- Validates models before they get persisted to the database
- Performs database operations in an object-oriented fashion

One to One: One to one relationship is where one record in a table/model is only associated with one and only one record in another table/model. For example for our student and portfolio tables/models it is a one to one relationship where one student can only have one portfolio.

To set this up in Ruby on Rails when you scaffold/ create the tables you will need to add a reference to the Student model/table in the portfolio model/table.
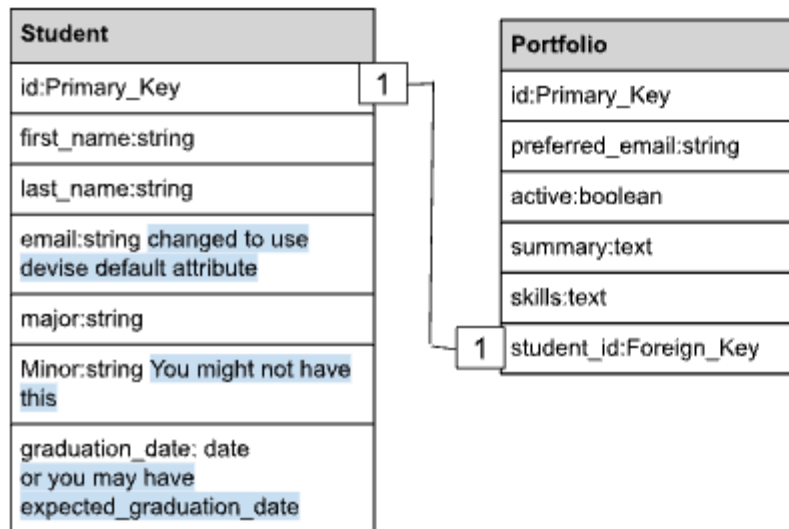
rails generate scaffold Portfolio email:text summary:text skills:text student:references

Your Portfolio model will look like this. The belongs to is an Active record association.

```
class Portfolio < ApplicationRecord
  belongs_to :student
end
```

Also in the students model we need to set up Active Record associations; this makes it so that when you delete a student record the portfolio is also deleted:

```
has_one :portfolio, dependent: :destroy
```

One to Many: In one to many relationships a record in one table can be associated with zero or more records in another table. For example the Projects model has a one to many relationship with the Students model. Where a student can have zero or more projects listed in their portfolio.

To set this up in Ruby on Rails when you scaffold/ create the tables you will need to add a reference to the Student model/table in the Projects model/table.

rails generate scaffold Projects title:text description:text student:references

Project Model:

class Project < ApplicationRecord
 belongs_to :student
end

In your Student model you will need this code to set up Active record association:

has_many :project