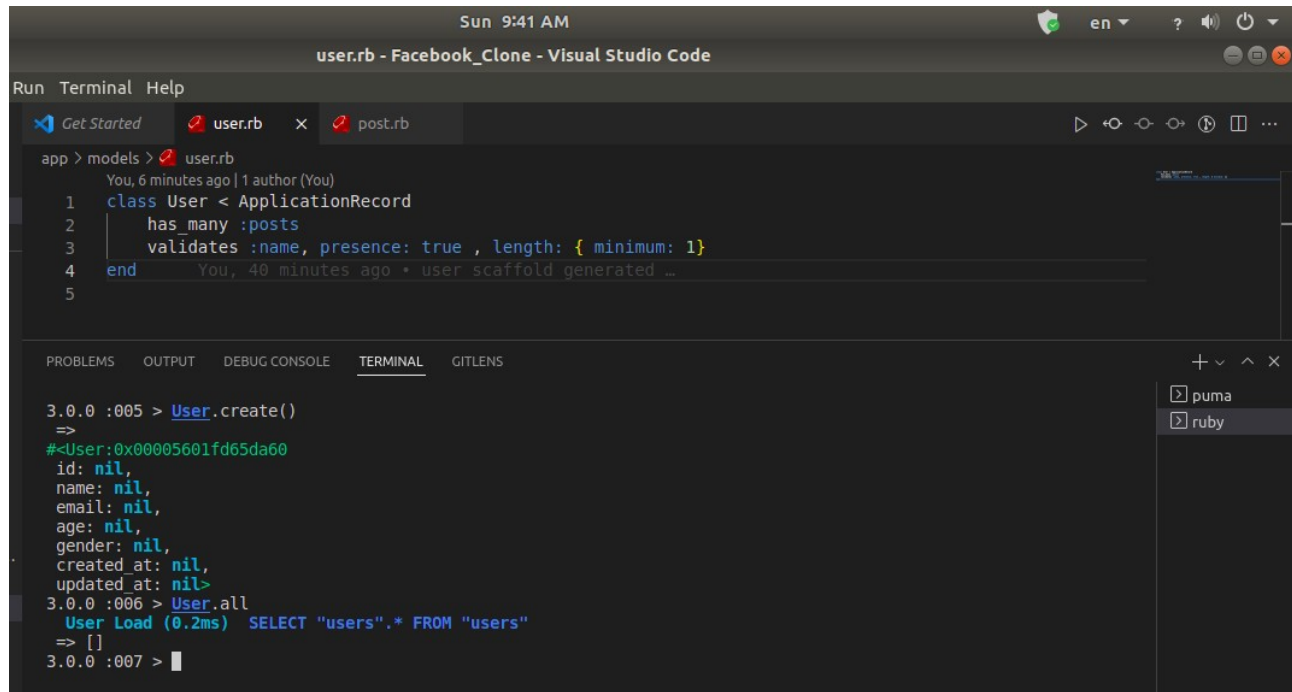


Creating empty user yet there is validation that user must have name

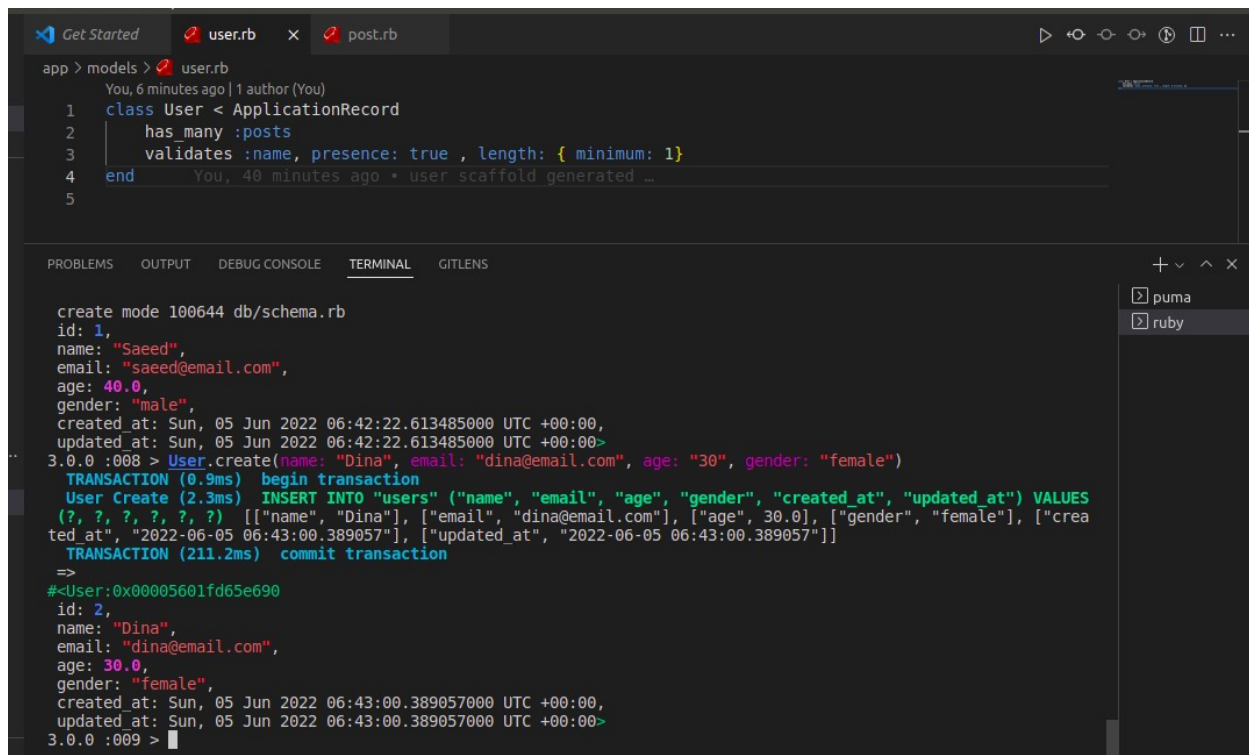


The screenshot shows a Visual Studio Code editor with a file named `user.rb` open. The code defines a `User` class that inherits from `ApplicationRecord`. It has a `has_many :posts` association and a `validates :name, presence: true, length: { minimum: 1}` validation. The terminal output shows the command `3.0.0 :005 > User.create()` being executed, which results in an empty user object with `id: nil`, `name: nil`, `email: nil`, `age: nil`, and `gender: nil`. The terminal also shows the command `3.0.0 :006 > User.all` being executed, which returns an empty array `[]`.

```
app > models > user.rb
You, 6 minutes ago | 1 author (You)
1 class User < ApplicationRecord
2   has_many :posts
3   validates :name, presence: true, length: { minimum: 1}
4 end
5

3.0.0 :005 > User.create()
=>
#<User:0x0005601fd65da60
 id: nil,
 name: nil,
 email: nil,
 age: nil,
 gender: nil,
 created_at: nil,
 updated_at: nil>
3.0.0 :006 > User.all
User Load (0.2ms) SELECT "users".* FROM "users"
=> []
3.0.0 :007 >
```

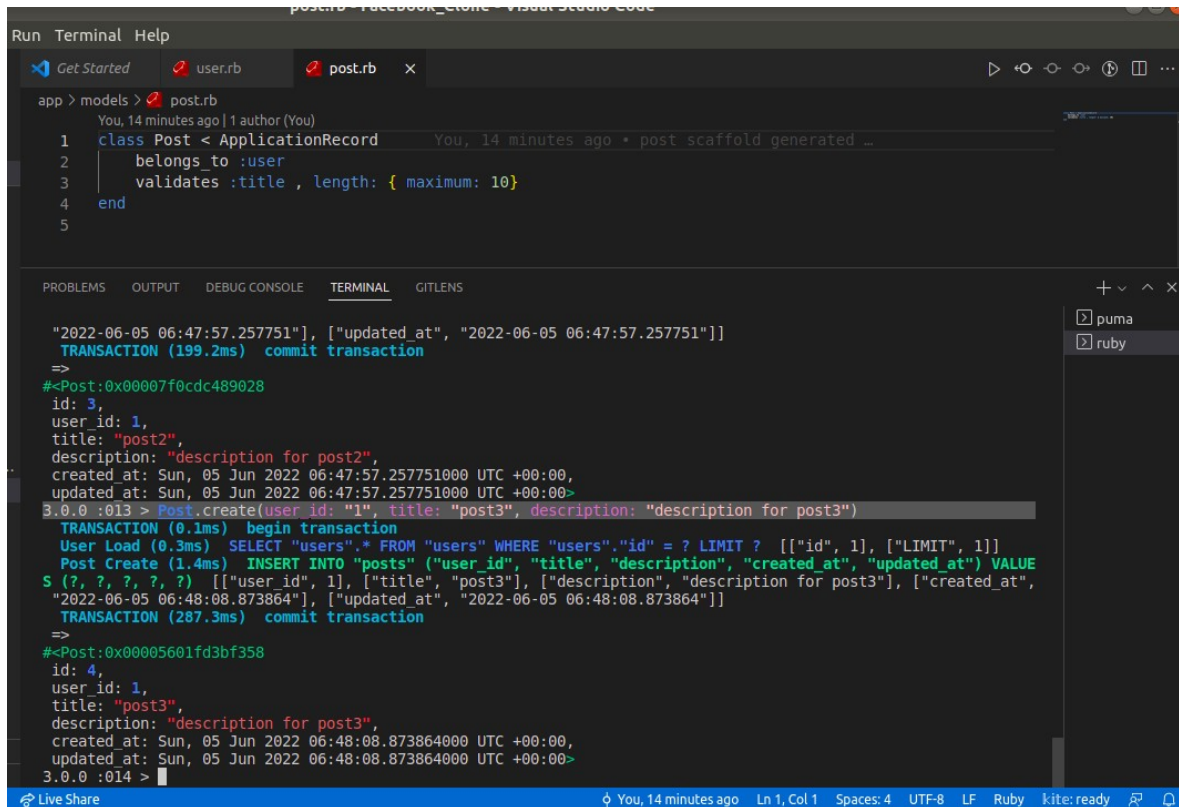
Creating 2 users with correct information



The screenshot shows the same Visual Studio Code editor with the `user.rb` file. The terminal output shows the command `create mode 100644 db/schema.rb` being executed. Then, the command `3.0.0 :008 > User.create(name: "Dina", email: "dina@email.com", age: "30", gender: "female")` is executed, which results in a new user object with `id: 2`, `name: "Dina"`, `email: "dina@email.com"`, `age: 30.0`, and `gender: "female"`. The terminal also shows the command `3.0.0 :009 >` being executed.

```
create mode 100644 db/schema.rb
id: 1,
 name: "Saeed",
 email: "saeed@email.com",
 age: 40.0,
 gender: "male",
 created_at: Sun, 05 Jun 2022 06:42:22.613485000 UTC +00:00,
 updated_at: Sun, 05 Jun 2022 06:42:22.613485000 UTC +00:00>
3.0.0 :008 > User.create(name: "Dina", email: "dina@email.com", age: "30", gender: "female")
TRANSACTION (0.9ms) begin transaction
User Create (2.3ms) INSERT INTO "users" ("name", "email", "age", "gender", "created_at", "updated_at") VALUES
(?, ?, ?, ?, ?, ?) [["name", "Dina"], ["email", "dina@email.com"], ["age", 30.0], ["gender", "female"], ["crea
ted_at", "2022-06-05 06:43:00.389057"], ["updated_at", "2022-06-05 06:43:00.389057"]]
TRANSACTION (211.2ms) commit transaction
=>
#<User:0x0005601fd65e690
 id: 2,
 name: "Dina",
 email: "dina@email.com",
 age: 30.0,
 gender: "female",
 created_at: Sun, 05 Jun 2022 06:43:00.389057000 UTC +00:00,
 updated_at: Sun, 05 Jun 2022 06:43:00.389057000 UTC +00:00>
3.0.0 :009 >
```

Creating posts

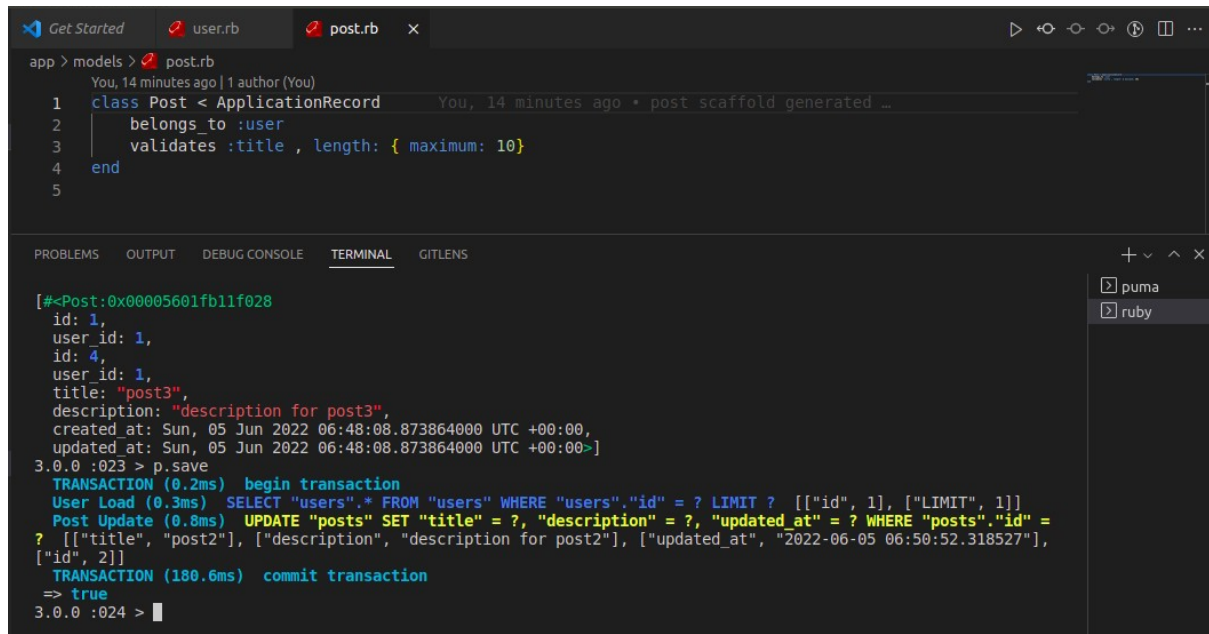


The screenshot shows the Visual Studio Code editor with a file named `post.rb` open. The code defines a `Post` model that belongs to a `user` and has a `title` validation with a maximum length of 10. Below the code, the terminal window shows the output of the Rails console. It displays the creation of a new post with the title "post2" and a description "description for post2". The terminal output includes the transaction details, the object representation of the post, and the successful completion of the `Post.create` method call.

```
app > models > post.rb
You, 14 minutes ago | 1 author (You)
1 class Post < ApplicationRecord
2   belongs_to :user
3   validates :title, length: { maximum: 10}
4 end
5

"2022-06-05 06:47:57.257751", [{"updated_at", "2022-06-05 06:47:57.257751"}]
TRANSACTION (199.2ms) commit transaction
=>
#<Post:0x00007f0cdc489028
 id: 3,
 user_id: 1,
 title: "post2",
 description: "description for post2",
 created_at: Sun, 05 Jun 2022 06:47:57.257751000 UTC +00:00,
 updated_at: Sun, 05 Jun 2022 06:47:57.257751000 UTC +00:00>
3.0.0 :013 > Post.create(user_id: "1", title: "post3", description: "description for post3")
TRANSACTION (0.1ms) begin transaction
User Load (0.3ms) SELECT "users".* FROM "users" WHERE "users"."id" = ? LIMIT ? [{"id", 1}, {"LIMIT", 1}]
Post Create (1.4ms) INSERT INTO "posts" ("user_id", "title", "description", "created_at", "updated_at") VALUE
S (?, ?, ?, ?, ?) [{"user_id", 1}, {"title", "post3"}, {"description", "description for post3"}, {"created_at",
"2022-06-05 06:48:08.873864"}, {"updated_at", "2022-06-05 06:48:08.873864"}]
TRANSACTION (287.3ms) commit transaction
=>
#<Post:0x00005601fd3bf358
 id: 4,
 user_id: 1,
 title: "post3",
 description: "description for post3",
 created_at: Sun, 05 Jun 2022 06:48:08.873864000 UTC +00:00,
 updated_at: Sun, 05 Jun 2022 06:48:08.873864000 UTC +00:00>
3.0.0 :014 >
```

Changing post content



The screenshot shows the Visual Studio Code editor with the same `post.rb` file. The terminal window now shows the output of the Rails console for updating a post. It displays the object representation of the post with the title "post3" and a description "description for post3". The terminal output includes the transaction details, the object representation of the post, and the successful completion of the `p.save` method call.

```
app > models > post.rb
You, 14 minutes ago | 1 author (You)
1 class Post < ApplicationRecord
2   belongs_to :user
3   validates :title, length: { maximum: 10}
4 end
5

[ #<Post:0x00005601fb11f028
 id: 1,
 user_id: 1,
 id: 4,
 user_id: 1,
 title: "post3",
 description: "description for post3",
 created_at: Sun, 05 Jun 2022 06:48:08.873864000 UTC +00:00,
 updated_at: Sun, 05 Jun 2022 06:48:08.873864000 UTC +00:00> ]
3.0.0 :023 > p.save
TRANSACTION (0.2ms) begin transaction
User Load (0.3ms) SELECT "users".* FROM "users" WHERE "users"."id" = ? LIMIT ? [{"id", 1}, {"LIMIT", 1}]
Post Update (0.8ms) UPDATE "posts" SET "title" = ?, "description" = ?, "updated_at" = ? WHERE "posts"."id" =
? [{"title", "post2"}, {"description", "description for post2"}, {"updated_at", "2022-06-05 06:50:52.318527"},
{"id", 2}]
TRANSACTION (180.6ms) commit transaction
=> true
3.0.0 :024 >
```

Creating long post which contradicts with validation

```
Get Started  user.rb  post.rb  x
app > models > post.rb
You, 14 minutes ago | 1 author (You)
1 class Post < ApplicationRecord
2   belongs_to :user
3   validates :title, length: { maximum: 10}
4 end
5

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  GITLENS

=>
#<User:0x00005601fd65e690
id: 2,
name: "Dina",
email: "dina@email.com",
age: 30.0,
gender: "female",
created at: Sun, 05 Jun 2022 06:43:00.389057000 UTC +00:00,
updated at: Sun, 05 Jun 2022 06:43:00.389057000 UTC +00:00>
3.0.0 :009 > Post.create(user_id: "1", title: "post1: This Title is longer than 10 characters", description: "de
scription for post1")
TRANSACTION (0.2ms) begin transaction
User Load (0.6ms) SELECT "users".* FROM "users" WHERE "users"."id" = ? LIMIT ? [["id", 1], ["LIMIT", 1]]
TRANSACTION (0.2ms) rollback transaction
=>
#<Post:0x00007f0cdc7f6648
id: nil,
user id: 1,
title: "post1: This Title is longer than 10 characters",
description: "description for post1",
created at: nil,
updated at: nil>
3.0.0 :010 >
```

Deleting user

```
app > models > post.rb
You, 14 minutes ago | 1 author (You)
1 class Post < ApplicationRecord
2   belongs_to :user
3   validates :title, length: { maximum: 10}
4 end
5

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  GITLENS

3.0.0 :045 > user_dina
=>
#<User:0x00005601fd334438
id: 3,
name: "Dina",
email: "Dina@email.com",
age: 30.0,
gender: "Female",
created at: Sun, 05 Jun 2022 07:13:42.852801000 UTC +00:00,
updated at: Sun, 05 Jun 2022 07:13:42.852801000 UTC +00:00>
3.0.0 :046 > user_dina.destroy
TRANSACTION (0.2ms) begin transaction
User Destroy (0.8ms) DELETE FROM "users" WHERE "users"."id" = ? [["id", 3]]
TRANSACTION (201.8ms) commit transaction
=>
#<User:0x00005601fd334438
id: 3,
name: "Dina",
email: "Dina@email.com",
age: 30.0,
gender: "Female",
created at: Sun, 05 Jun 2022 07:13:42.852801000 UTC +00:00,
updated at: Sun, 05 Jun 2022 07:13:42.852801000 UTC +00:00>
3.0.0 :047 >
```

Live Share You, 14 minutes ago Ln 1, Col 1 Spaces: 4 UTF-8 LF Ruby k:ite: initializing

Deleting all posts

The screenshot shows a code editor with a file named `post.rb` and a terminal window. The code defines a `Post` class that belongs to a `user` and has a validation for the title length. The terminal shows the command `Post.destroy_all` being executed, which results in four database transactions deleting posts with IDs 1, 2, 3, and 4. The terminal also displays the details of the first two posts being deleted.

```
app > models > post.rb
You, 14 minutes ago | 1 author (You)
1 class Post < ApplicationRecord
2   belongs_to :user
3   validates :title, length: { maximum: 10}
4 end
5

3.0.0 :047 > Post.destroy_all
Post Load (0.4ms) SELECT "posts".* FROM "posts"
TRANSACTION (0.2ms) begin transaction
Post Destroy (1.0ms) DELETE FROM "posts" WHERE "posts"."id" = ? [{"id", 1}]
TRANSACTION (194.9ms) commit transaction
TRANSACTION (0.2ms) begin transaction
Post Destroy (0.7ms) DELETE FROM "posts" WHERE "posts"."id" = ? [{"id", 2}]
TRANSACTION (204.0ms) commit transaction
TRANSACTION (0.2ms) begin transaction
Post Destroy (0.7ms) DELETE FROM "posts" WHERE "posts"."id" = ? [{"id", 3}]
TRANSACTION (215.0ms) commit transaction
TRANSACTION (0.1ms) begin transaction
Post Destroy (0.6ms) DELETE FROM "posts" WHERE "posts"."id" = ? [{"id", 4}]
TRANSACTION (215.5ms) commit transaction
=>
[#<Post:0x00007f0cdc38b900
 id: 1,
 user_id: 1,
 title: "post1",
 description: "description for post1",
 created_at: Sun, 05 Jun 2022 06:47:17.682554000 UTC +00:00,
 updated_at: Sun, 05 Jun 2022 06:47:17.682554000 UTC +00:00>,
 #<Post:0x00007f0cdc38b838
 id: 2,
 user_id: 1,
 title: "post2",
```