

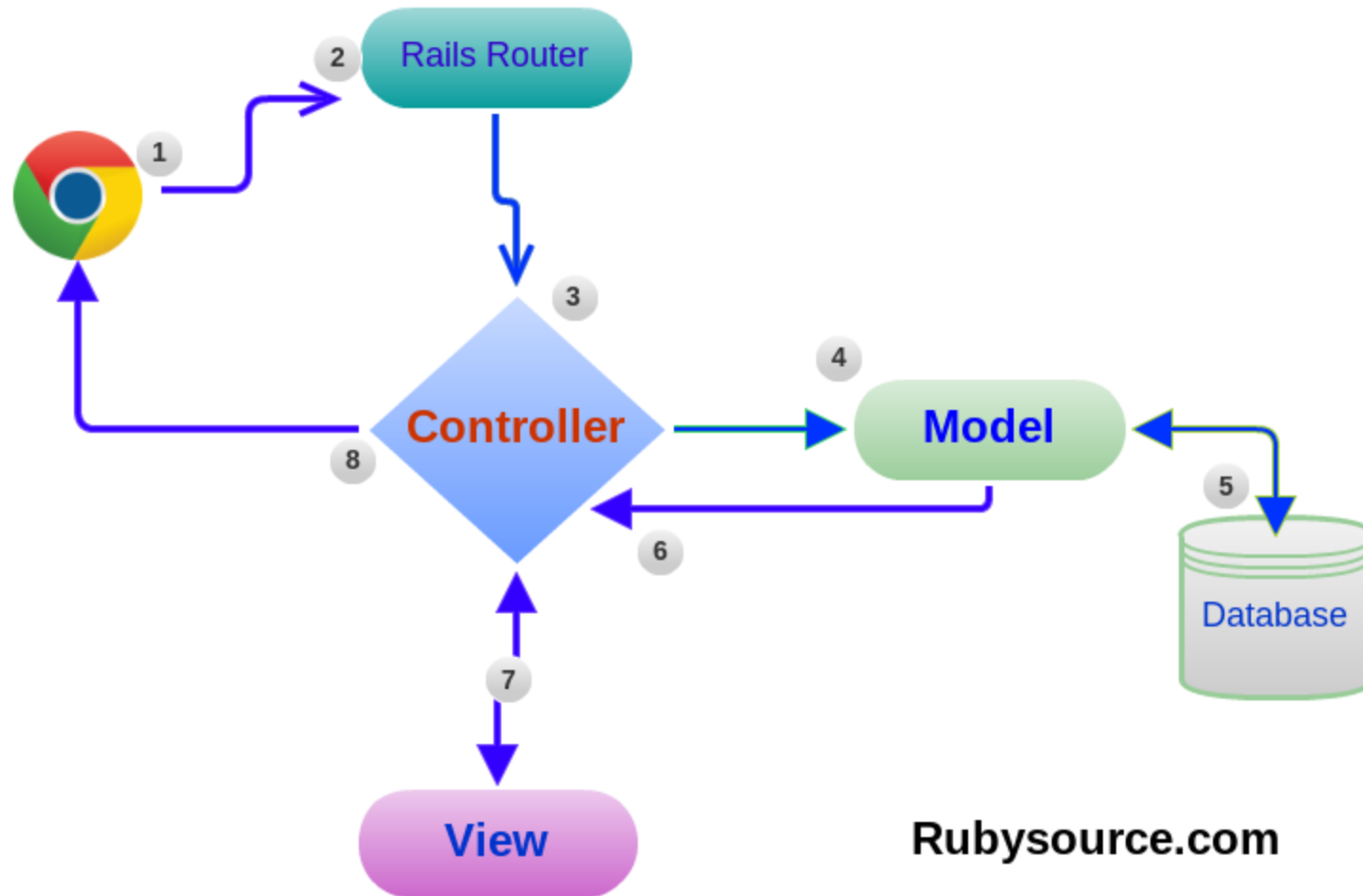
# Модели

## Хекслет: Rails-программист

# Программа урока

- Что такое модели в Rails и для чего они нужны
- Как связаны код и модель
- Как и зачем валидировать модель
- Как создавать, изменять, и удалять записи в БД, используя модели

# Model-View-Controller



Rubysource.com

# ActiveRecord

- Представление моделей и их данных
- Представление связей между моделями
- Представление иерархий наследования с помощью связанных моделей
- Валидация моделей до того, как они станут персистентными в базе данных
- Выполнение операций с базой данных в объектно-ориентированном стиле

# ActiveRecord

```
class User < ApplicationRecord
  validates :first_name, presence: true
  validates_inclusion_of :role, in: %w{ customer admin superuser }
  validates_length_of :last_name, maximum: 30, message: "less than 30 if you don't mind"

  attribute :is_active, :boolean, default: -> { false }

  scope :admins, -> {where(role: 'admin')}
  has_many :orders
  belongs_to :city
end
```

# ActiveRecord: Связь класса модели с таблицей

Class	Table name
Article	articles
LineItem	line_items
Deer	deers
Mouse	mice
Person	people

# Генерация моделей

```
bin/rails g model User first_name last_name is_active:boolean
```

```
invoke  active_record
create  db/migrate/20211202144414_create_users.rb
create  app/models/user.rb
invoke  test_unit
create  test/models/user_test.rb
create  test/fixtures/users.yml
```

```
class User < ApplicationRecord
end
```

# Миграции

```
class CreateUsers < ActiveRecord::Migration[6.1]
  def change
    create_table :users do |t|
      t.string :first_name
      t.string :last_name
      t.boolean :is_active
      t.timestamps
    end
  end
end
```

```
bin/rails g migration create_users first_name last_name is_admin:boolean is_active:boolean
```

```
bin/rails db:migrate
```



# Cxema

## db/schema.rb

```
ActiveRecord::Schema.define(version: 2021_12_02_144414) do
  create_table "users", force: :cascade do |t|
    t.string "first_name"
    t.string "last_name"
    t.boolean "is_active"
    t.datetime "created_at", precision: 6, null: false
    t.datetime "updated_at", precision: 6, null: false
  end
end
```

# Database tasks

```
rake db:create  
rake db:drop  
rake db:migrate  
rake db:prepare  
rake db:reset  
rake db:rollback STEP=3  
rake db:schema:dump  
rake db:schema:load  
rake db:seed  
rake db:setup
```

Полный список:

```
rake -T 'db'
```

# ActiveRecord: Create

```
admin = User.create(first_name: 'John', last_name: 'Doe', role: 'admin')
```

```
user = User.new(first_name: 'Tom', last_name: 'Smith', role: 'customer')  
user.save
```

# ActiveRecord: Errors

```
user = User.new(last_name: 'Smith', role: 'customer')

user.save

user.validate! # => ["cannot be nil"]
user.valid? # false
user.invalid? # true
user.errors # Error object
user.errors.messages # {"first_name" => "cannot be nil"}
user.errors.full_messages # ["first_name cannot be nil"]
```

# ActiveRecord: Read

```
users = User.all
user = User.first
user = User.find(1)
user = User.find_by(id: 1)
user = User.last
users = User.where(name: 'John', is_admin: true).order(created_at: :desc)
```

# ActiveRecord: Delete

```
user = User.find_by(name: 'John')  
user.destroy
```

```
User.destroy_all
```