

# Laravel Eager Loading

## Laravel Eager Loading

### ### How Eager Loading Works

Eager loading works by modifying the SQL queries generated by Laravel's Eloquent ORM to include the related models in a single query, using JOINS or subqueries. This reduces the number of queries executed against the database, thereby improving performance.

### ### Why Use Eager Loading

1. **Performance Improvement**: Eager loading reduces the number of queries executed. This is especially important for applications with large datasets or complex relationships, as it minimizes the overhead of multiple database round trips.
2. **Avoiding the N+1 Query Problem**: By loading related models in a single query, eager loading prevents the N+1 query problem, where N queries are executed for fetching related models for each instance of the main model.
3. **Simpler Code**: Eager loading can make the code simpler and more readable, as it handles fetching related models automatically.

### ### Eager Loading Methods

1. **with()**: The `with()` method is the most common way to perform eager loading. It accepts one or more relationship names to be loaded.

```
```php
```

```
$posts = Post::with('comments')->get();
```

## Laravel Eager Loading

```
...
```

2. **load()**: The `load()` method is used to eager load relationships on an already retrieved model instance.

```
```php
$posts = Post::all();
$posts->load('comments');
...

```

3. **loadMissing()**: The `loadMissing()` method is similar to `load()`, but it only loads the relationships if they are not already loaded.

```
```php
$posts = Post::all();
$posts->loadMissing('comments');
...

```

### ### Example

Consider two models, `Post` and `Comment`, with a one-to-many relationship:

```
```php
class Post extends Model {
    public function comments() {
        return $this->hasMany(Comment::class);
    }
}

```

## Laravel Eager Loading

```
}
```

```
class Comment extends Model {  
    public function post() {  
        return $this->belongsTo(Post::class);  
    }  
}  
...  
...  
...
```

#### Without Eager Loading:

```
```php
```

```
$posts = Post::all(); // Fetch all posts
```

```
foreach ($posts as $post) {  
    $comments = $post->comments; // Fetch comments for each post (N+1 queries)  
}  
...  
...  
...
```

#### With Eager Loading:

```
```php
```

```
$posts = Post::with('comments')->get(); // Fetch posts and their comments in two queries
```

```
...  
...  
...
```

#### Using load() Method:

```
```php
```

## Laravel Eager Loading

```
$posts = Post::all();  
  
$posts->load('comments'); // Load comments relationship on already retrieved posts  
  
...
```

### ### Summary

- **How It Works**: Eager loading modifies SQL queries to include related models, reducing the number of queries.
- **Why Use It**: To improve performance, avoid the N+1 query problem, and simplify code.
- **Methods**: ``with()``, ``load()``, and ``loadMissing()`` are used to perform eager loading in Laravel.