

Nom: Cognoms:

Enunciat de la pràctica

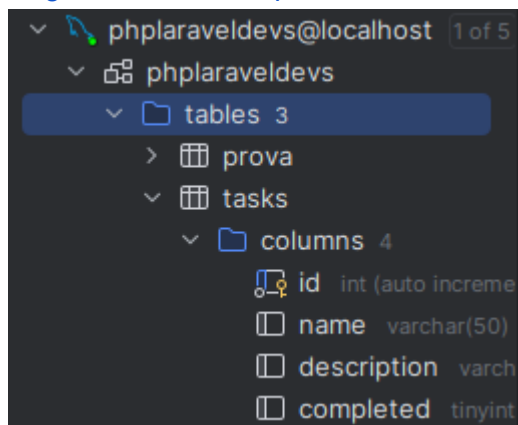
- Crear migració a la taula tasks. Ha de tenir els camps (id, name, description i completed)
- Crear factory
- Crear seed
- Crear inserts a partir del seed
- Crear controlador de tasks
- Crear model de tasks
- Crear ruta a routes/web.php i afegir la nova ruta al navegador al fitxer resources/views/components/layout.blade.php
- Crear la vista tasks.blade.php

Resolució de la pràctica

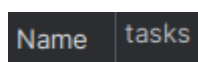
- Crear migració a la taula tasks . Ha de tenir els camps (id, name, description i completed)
- Per crear la taula primer clico a la icona de la dreta de Database.



Faig clic dret a la carpeta “tables” de la base de dades phplaraveldevs i clico New i Table.



Escric “tasks” com a nom de taula.



Escriu la definició de la taula. Conté els camps id, name, description i completed,

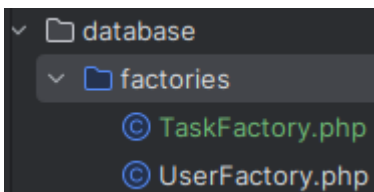
```
CREATE TABLE tasks (  
    id INT AUTO_INCREMENT PRIMARY KEY ,  
    name VARCHAR(50),  
    description VARCHAR(100),  
    completed TINYINT(1)  
);|
```

Clico OK.



- Crear factory

Creo TaskFactory a partir de UserFactory. Faig clic dret a UserFactory i copio. Faig clic dret a factories i pego.



Contingut de la classe TaskFactory.

```
class TaskFactory extends Factory  
{  
    /**  
     * Define the model's default state.  
     *  
     * @return array<string, mixed>  
     */  
    no usages new *  
    public function definition(): array  
    {  
        return [  
            'name' => $this->faker->name(),  
            'description' => $this->faker->unique()->safeEmail(),  
            'completed' => $this->faker->randomElement([0, 1]),  
        ];  
    }  
}
```

- Craer model de tasks

Creo el model de Task.

```
→ laravelApp git:(main) X php artisan make:model Task  
INFO Model [app/Models/Task.php] created successfully.
```

Li poso el següent contingut.

```
class Task extends Model  
{  
    use HasFactory;  
  
    no usages  
    protected $fillable = [  
        'name',  
        'description',  
        'completed'  
    ];  
}
```

- Crear seed

Creo el seeder per a la taula tasks.

```
→ laravelApp git:(main) X php artisan make:seeder TaskSeeder  
INFO Seeder [database/seeds/TaskSeeder.php] created successfully.
```

Li poso aquest contingut. Encara falta crear el model Task.

```
class TasksTableSeeder extends Seeder  
{  
    /**  
     * Run the database seeds.  
     */  
    public function run(): void  
    {  
        \App\Models\Task::factory(count: 10)->create();  
    }  
}
```

- Crear inserts a partir del seed

He hagut d'afegir "\$table->timestamps();" a la migració.

```
public function up(): void
{
    Schema::create( table: 'tasks', function (Blueprint $table) {
        $table->id();
        $table->string( column: 'name');
        $table->string( column: 'description');
        $table->boolean( column: 'completed');
        $table->timestamps();
    });
}
```

Executo el TaskSeeder.

php artisan db:seed --class=TaskSeeder

```
→ laravelApp git:(main) X php artisan db:seed --class=TaskSeeder

INFO Seeding database.
```

Comprovo que s'han creat els inserts.

	id	name	description	completed
1	1	Quinton Hoppe	savion24@example.com	1
2	2	Jacquelyn Champlin	vledner@example.net	1
3	3	Elias Murray	silas.larkin@example.com	0
4	4	Khalil Kertzmann	josianne67@example.com	1
5	5	Nelson Abshire	danyka.effertz@example.org	1
6	6	Merritt Flatley	constantin83@example.org	1
7	7	Dr. Candelario Bartell	miguel59@example.org	0
8	8	Berenice Cummings	travis99@example.org	0
9	9	Barry Jacobs	dorothea43@example.net	1
10	10	Malika Koelpin	jarrod46@example.org	0

- Crear controlador de tasks

Creo el controlador.

php artisan make:controller TaskController

```
→ laravelApp git:(main) X php artisan make:controller TaskController

INFO Controller [app/Http/Controllers/TaskController.php] created successfully.
```

- Crear ruta a routes/web.php i afegir la nova ruta al navegador al fitxer resources/views/components/layout.blade.php

Poso la ruta de tasks baix de la de users a web.php.

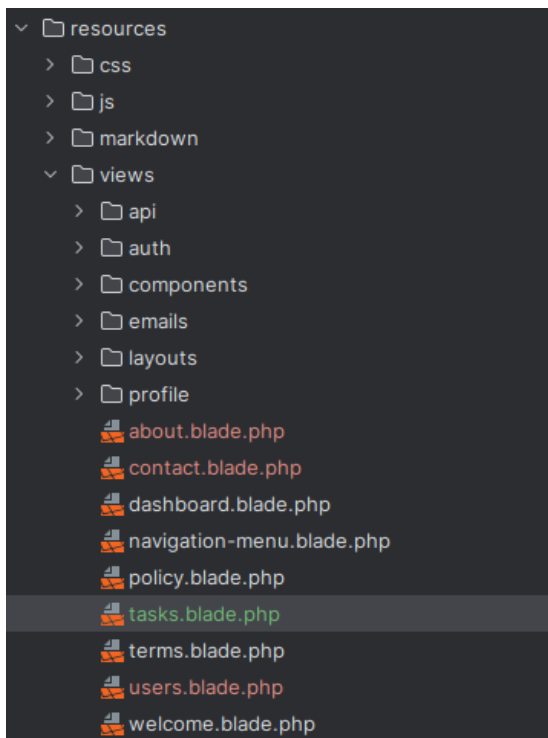
```
Route::get( uri: '/', [\App\Http\Controllers\UserController::class, 'index']);
Route::get( uri: '/users', [\App\Http\Controllers\UserController::class, 'index']);
Route::get( uri: '/tasks', [\App\Http\Controllers\TaskController::class, 'index']);
Route::get( uri: '/contact', [\App\Http\Controllers\PagesController::class, 'contact']);
Route::get( uri: '/about', [\App\Http\Controllers\PagesController::class, 'about']);
```

També poso la ruta de tasques a layout.web.php.

```
<div class="hidden md:block">
  <div class="ml-10 flex items-baseline space-x-4">
    <!-- Current: "bg-gray-900 text-white", Default: "text-gray-300 hover:bg-gray-700 hover:text-white" -->
    <a href="/users" class="text-gray-300 hover:bg-gray-700 hover:text-white rounded-md px-3 py-2 text-sm font-medium">Usuaris</a>
    <a href="/tasks" class="text-gray-300 hover:bg-gray-700 hover:text-white rounded-md px-3 py-2 text-sm font-medium">Tasques</a>
    <a href="/contact" class="text-gray-300 hover:bg-gray-700 hover:text-white rounded-md px-3 py-2 text-sm font-medium">Contact</a>
    <a href="/about" class="text-gray-300 hover:bg-gray-700 hover:text-white rounded-md px-3 py-2 text-sm font-medium">About</a>
  </div>
</div>
```

- Crear la vista tasks.blade.php

Creo la vista tasks.blade.php a partir de a de usuarios (copiar y pegar).




Escribe el código de la tabla según los campos de la tabla de tareas de la base de datos (id, nom, descripción, completada).

```



<div class="px-4 sm:px-6 lg:px-8">
  <div class="mt-8 flow-root">
    <div class="-mx-4 -my-2 overflow-x-auto sm:-mx-6 lg:-mx-8">
      <div class="inline-block min-w-full py-2 align-middle sm:px-6 lg:px-8">
        <table class="min-w-full divide-y divide-gray-300">
          <thead>
            <tr>
              <th scope="col" class="py-3.5 pl-4 pr-3 text-left text-sm font-semibold text-gray-900 sm:pl-3">Id</th>
              <th scope="col" class="px-3 py-3.5 text-left text-sm font-semibold text-gray-900">Nom</th>
              <th scope="col" class="px-3 py-3.5 text-left text-sm font-semibold text-gray-900">Descripció</th>
              <th scope="col" class="px-3 py-3.5 text-left text-sm font-semibold text-gray-900">Completada</th>
            </tr>
          </thead>
          <tbody class="bg-white">
            @foreach($tasks as $task)
              <tr class="even:bg-gray-50">
                <td class="whitespace-nowrap py-4 pl-4 pr-3 text-sm font-medium text-gray-900 sm:pl-3">{{ $task->id }}</td>
                <td class="whitespace-nowrap px-3 py-4 text-sm text-gray-500">{{ $task->name }}</td>
                <td class="whitespace-nowrap px-3 py-4 text-sm text-gray-500">{{ $task->description }}</td>
                <td class="whitespace-nowrap px-3 py-4 text-sm text-gray-500">{{ $task->completed }}</td>
              </tr>
            @endforeach
            <!-- More people... -->
          </tbody>
        </table>
      </div>
    </div>
  </div>
</div>

```

Veo a la página y compruebo que la pestaña de Tareas funciona correctamente. Podemos ver que surt la tabla de tareas con los registros insertados por el seeder.
 <http://laravelapp.test/tasks>



Usuaris
 Tasques
 Contact
 About

TASQUES

Id	Nom	Descripció	Completada
1	Quinton Hoppe	savion24@example.com	1
2	Jacquelyn Champlin	vledner@example.net	1
3	Elias Murray	silas.larkin@example.com	0
4	Khalil Kertzmann	josiianne67@example.com	1
5	Nelson Abshire	danyka.effertz@example.org	1
6	Merritt Flatley	constantin83@example.org	1
7	Dr. Candelario Bartell	miguel59@example.org	0
8	Berenice Cummings	travis99@example.org	0
9	Barry Jacobs	dorothea43@example.net	1
10	Malika Koelpin	jarrod46@example.org	0