
myapp

Versión 1

Deivid Palacios

05 de junio de 2025

Contents:

1. myapp	3
Índice de Módulos Python	7

Add your content using reStructuredText syntax. See the [reStructuredText](#) documentation for details.


```
class myapp.models.Project(id, name)
```

Bases: Model

```
exception DoesNotExist
```

Bases: ObjectDoesNotExist

```
exception MultipleObjectsReturned
```

Bases: MultipleObjectsReturned

id

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

name

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

task_set

Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

Parent.children is a ReverseManyToOneDescriptor instance.

Most of the implementation is delegated to a dynamically defined manager class built by create_forward_many_to_many_manager() defined below.

```
class myapp.models.Task(id, title, description, project, created, datecompleted, important, user)
```

Bases: Model

exception DoesNotExist

Bases: `ObjectDoesNotExist`

exception MultipleObjectsReturned

Bases: `MultipleObjectsReturned`

created

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

datecompleted

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

description

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

get_next_by_created(**, field=<django.db.models.fields.DateTimeField: created>, is_next=True, **kwargs*)

get_previous_by_created(**, field=<django.db.models.fields.DateTimeField: created>, is_next=False, **kwargs*)

id

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

important

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = `<django.db.models.manager.Manager object>`

project

Accessor to the related object on the forward side of a many-to-one or one-to-one (via `ForwardOneToOneDescriptor` subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

project_id**title**

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

user

Accessor to the related object on the forward side of a many-to-one or one-to-one (via `ForwardOneToOneDescriptor` subclass) relation.

In the example:


```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

Child.parent is a ForwardManyToOneDescriptor instance.

user_id

```
class myapp.apps.MyappConfig(app_name, app_module)
    Bases: AppConfig
    default_auto_field = 'django.db.models.BigAutoField'
    name = 'myapp'
```

Vistas de Django para autenticación de usuarios, gestión de proyectos y tareas. Functions:

home(request):

Renderizar la página de inicio.

registro(request):

Gestionar el registro de usuarios. En GET, mostrar el formulario de registro. En POST, crear un usuario si las contraseñas coinciden.

inicio_sesion(request):

Gestionar el inicio de sesión del usuario. En GET, mostrar el formulario de inicio de sesión. En POST, autenticar e iniciar sesión del usuario.

cerrar_sesion(request):

Cerrar la sesión del usuario actual y redirigir a la página de inicio de sesión.

index(request):

Mostrar la página de índice principal para los usuarios conectados, mostrando la información del usuario.

projects(request):

Mostrar una lista de todos los proyectos.

about(request):

Mostrar la página «Acerca de».

tasks(request):

Mostrar una lista de tareas incompletas para el usuario conectado.

tasks_completed(request):

Mostrar una lista de tareas completadas para el usuario conectado.

create_task(request):

Gestionar la creación de una nueva tarea. En GET, mostrar el formulario de creación de tareas. En POST, crear la tarea.

task_detail(request, task_id):

Mostrar y actualizar los detalles de una tarea específica perteneciente al usuario conectado.

complete_task(request, task_id):

Marcar una tarea específica como completada para el usuario conectado.

delete_task(request, task_id):

Elimina una tarea específica del usuario conectado.

create_project(request):

Gestiona la creación de un nuevo proyecto. En GET, muestra el formulario de creación del proyecto. En POST, crea el proyecto.

project_detail(request, id):

Muestra los detalles y las tareas asociadas a un proyecto específico.

```
myapp.views.about(request)
myapp.views.cerrar_sesion(request)
myapp.views.complete_task(request, task_id)
myapp.views.create_project(request)
myapp.views.create_task(request)
myapp.views.delete_task(request, task_id)
myapp.views.home(request)
myapp.views.index(request)
myapp.views.inicio_sesion(request)
myapp.views.project_detail(request, id)
myapp.views.projects(request)
myapp.views.registro(request)
myapp.views.task_detail(request, task_id)
myapp.views.tasks(request)
myapp.views.tasks_completed(request)
```

m

`myapp.apps`, [5](#)
`myapp.models`, [3](#)
`myapp.views`, [5](#)

\spxentryabout()\spxextraen el módulo myapp.views, 6			\spxentrymyapp.apps		
			\spxentrymodule, 5		
\spxentrycerrar_sesion()\spxextraen el módulo myapp.views, 6			\spxentrymyapp.models		
			\spxentrymodule, 3		
\spxentrycomplete_task()\spxextraen el módulo myapp.views, 6			\spxentrymyapp.views		
			\spxentrymodule, 5		
\spxentrycreate_project()\spxextraen el módulo myapp.views, 6			\spxentryMyappConfig\spxextraclase en myapp.apps, 5		
\spxentrycreate_task()\spxextraen el módulo myapp.views, 6			\spxentryname\spxextraatributo de myapp.apps.MyappConfig, 5		
\spxentrycreated\spxextraatributo de myapp.models.Task, 4			\spxentryname\spxextraatributo de myapp.models.Project, 3		
\spxentrydatecompleted\spxextraatributo de myapp.models.Task, 4			\spxentryobjects\spxextraatributo de myapp.models.Project, 3		
\spxentrydefault_auto_field\spxextraatributo de myapp.apps.MyappConfig, 5			\spxentryobjects\spxextraatributo de myapp.models.Task, 4		
\spxentrydelete_task()\spxextraen el módulo myapp.views, 6			\spxentryproject\spxextraatributo de myapp.models.Task, 4		
\spxentrydescription\spxextraatributo de myapp.models.Task, 4			\spxentryProject\spxextraclase en myapp.models, 3		
			\spxentryProject.DoesNotExist, 3		
\spxentryget_next_by_created()\spxextramétodo de myapp.models.Task, 4			\spxentryProject.MultipleObjectsReturned, 3		
\spxentryget_previous_by_created()\spxextramétodo de myapp.models.Task, 4			\spxentryproject_detail()\spxextraen el módulo myapp.views, 6		
			\spxentryproject_id\spxextraatributo de myapp.models.Task, 4		
\spxentryhome()\spxextraen el módulo myapp.views, 6			\spxentryprojects()\spxextraen el módulo myapp.views, 6		
\spxentryid\spxextraatributo de myapp.models.Project, 3			\spxentryregistro()\spxextraen el módulo myapp.views, 6		
\spxentryid\spxextraatributo de myapp.models.Task, 4					
\spxentryimportant\spxextraatributo de myapp.models.Task, 4			\spxentryTask\spxextraclase en myapp.models, 3		
			\spxentryTask.DoesNotExist, 3		
\spxentryindex()\spxextraen el módulo myapp.views, 6			\spxentryTask.MultipleObjectsReturned, 4		
\spxentryinicio_sesion()\spxextraen el módulo myapp.views, 6			\spxentrytask_detail()\spxextraen el módulo myapp.views, 6		
			\spxentrytask_set\spxextraatributo de myapp.models.Project, 3		
\spxentrymodule			\spxentrytasks()\spxextraen el módulo myapp.views, 6		
\spxentrymyapp.apps, 5			\spxentrytasks_completed()\spxextraen el módulo myapp.views, 6		
\spxentrymyapp.models, 3					
\spxentrymyapp.views, 5					

\spxentrytitle\spxextraatributo de myapp.models.Task, 4

\spxentryuser\spxextraatributo de myapp.models.Task, 4

\spxentryuser_id\spxextraatributo de myapp.models.Task,
5