

1.
¿En qué framework está basado FastAPI?
Starlette
2.
¿A partir de qué versión puedes utilizar FastAPI?
3.6
3.
¿Cuál es el comando correcto para ejecutar una aplicación con FastAPI?
uvicorn main:app
4.
¿Cuál es el comando correcto para establecer otro número de puerto al ejecutar tu aplicación?
uvicorn main:app --port 5000
5.
¿Cuál es la ruta para acceder a la documentación autogenerada?
/docs
6.
¿Cuál es la sintaxis para añadir un parámetro a una ruta?
/movies/{id}
7.
¿Cómo se puede añadir un parámetro query a una ruta?
Se agrega un parámetro a la función
def get_movies_by_category(category: str)
8.
¿Qué clase debemos usar para definir que un parámetro pertenece al cuerpo de la petición?
Body
9.
¿Qué método HTTP utilizarías si quisieras modificar información en tu API?
PUT
10.
¿De qué clase debe heredar un esquema de datos de Pydantic?
BaseModel
11.
¿Cómo puedes definir que un campo puede ser opcional?
title: Optional[str] = None
12.
¿Qué nombre debe tener la clase para realizar configuraciones en un esquema de Pydantic?
Config
- 13.

¿Cómo puedes validar que un número sea mayor o igual a otro?

Usando el parámetro `ge`

14.

¿Cuál es la clase que nos sirve para validar parámetros query?

`Query`

15.

¿Qué parámetro puedes usar en la clase `JSONResponse` para enviar la información al cliente?

`content`

16.

Si tuvieras un esquema llamado `Book`, ¿cómo le indicarías a una ruta que debe devolver datos de este tipo?

`response_model = Book`

17.

¿Qué parámetro debes usar para indicar el código de estado que devolverá una ruta?

`status_code`

18.

¿Qué código de estado puedes usar para indicar que se realizó un registro con éxito?

`201`

19.

¿Cuál es el nombre correcto del módulo para manejo de tokens?

`pyjwt`

20.

¿Cómo podemos añadir una clase como dependencia de una ruta?

Usando el parámetro `dependencies`