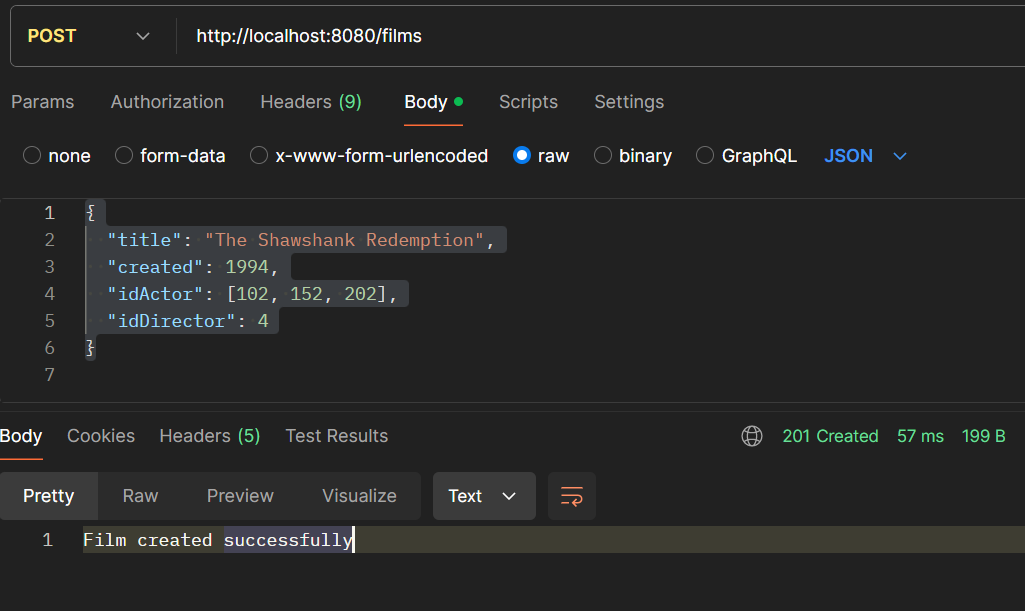
|  |  |
| --- | --- |
| http://localhost:8080/Ejecución Automática de Caso de Test | |
| Dominio | **Proyecto** |
| http://localhost:8080/ | Práctica 4 |
| Aplicación | **Ámbito Ejecución** |
| FilmController | Online |
| Descripción | **Pre-requisitos** |
| Prueba unitaria para PelículasCRUD |  |
| Tester Ejecución | **Fecha Ejecución** |
| x231598 | 29/05/2024 |

|  |  |
| --- | --- |
| # Paso: | 1 |
| Descripción | Operación de inserción correcta. |
| Resultado Esperado | La inserción se completa correctamente obteniendo respuesta HTTP 201 CREATED. La inserción en la base de datos se completa correctamente |
| OPs |  |
| O. Básica Bancaria |  |

Realizamos una petición REST de tipo POST la siguiente URL:

|  |
| --- |
| http://localhost:8080/films |



Con el siguiente body

{

  "title": "The Shawshank Redemption",

  "created": 1994,

  "idActor": [102, 152, 202],

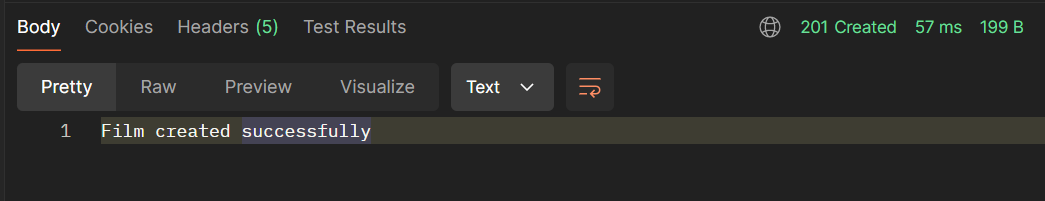
  "idDirector": 4

}



Revisamos la salida:

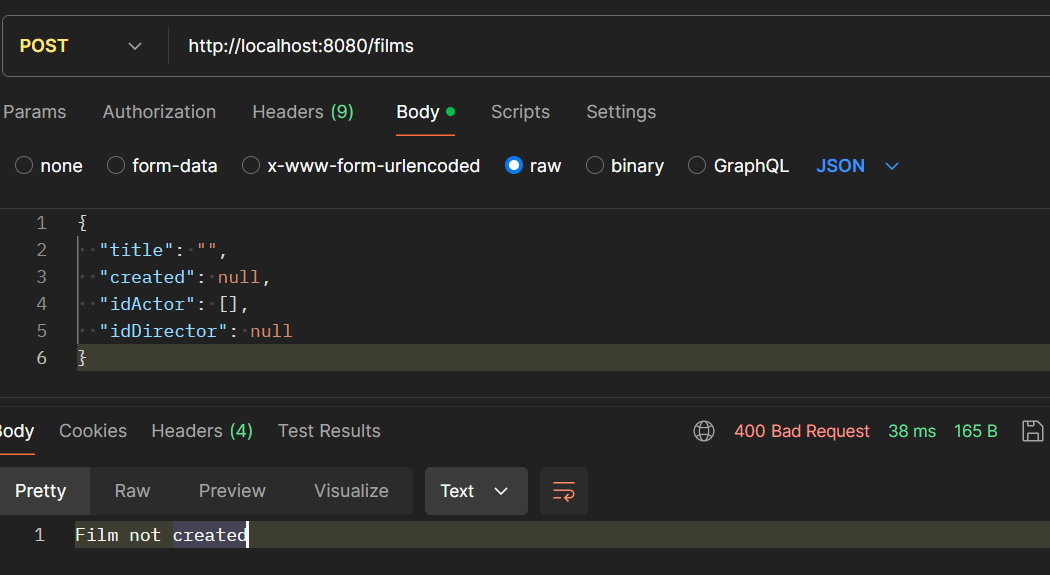




|  |  |
| --- | --- |
| # Paso: | 2 |
| Descripción | Operación de inserción incorrecta. |
| Resultado Esperado | La inserción se completa incorrectamente obteniendo respuesta HTTP 400 BAD REQUEST. La inserción en la base de datos no se realiza |
| OPs |  |
| O. Básica Bancaria |  |

Realizamos una petición REST de tipo POST la siguiente URL:

|  |
| --- |
| http://localhost:8080/films |



Con el siguiente body

{

  "title": "",

  "created": null,

  "idActor": [],

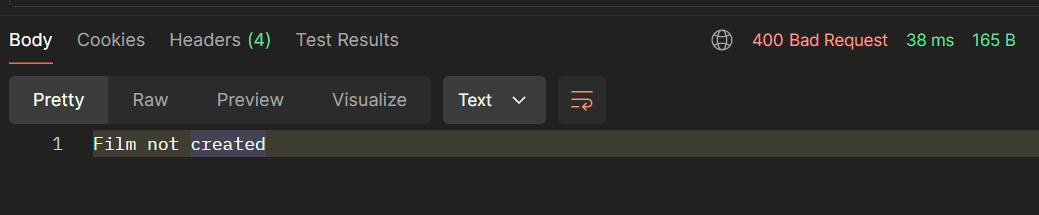
  "idDirector": null

}



Revisamos la salida:

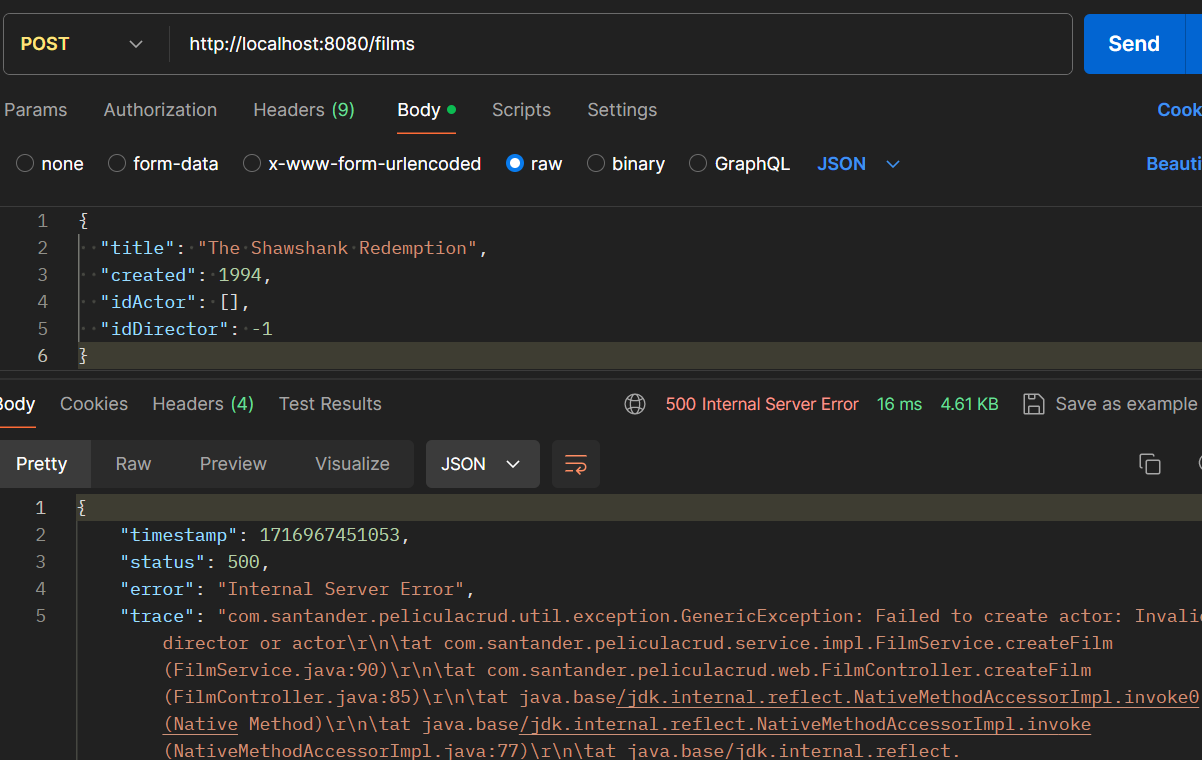




|  |  |
| --- | --- |
| # Paso: | 3 |
| Descripción | Operación de inserción incorrecta. |
| Resultado Esperado | La inserción se completa incorrectamente obteniendo respuesta HTTP 500 Internal Server Error. La inserción en la base de datos no se realiza |
| OPs |  |
| O. Básica Bancaria |  |

Realizamos una petición REST de tipo POST la siguiente URL:

|  |
| --- |
| http://localhost:8080/films |



Con el siguiente body

{

  "title": "The Shawshank Redemption",

  "created": 1994,

  "idActor": [],

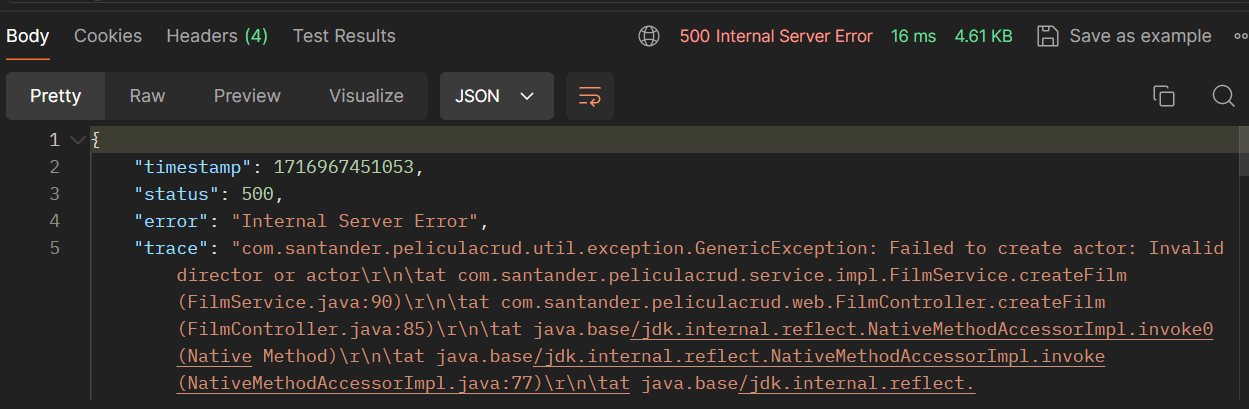
  "idDirector": -1

}



Revisamos la salida:

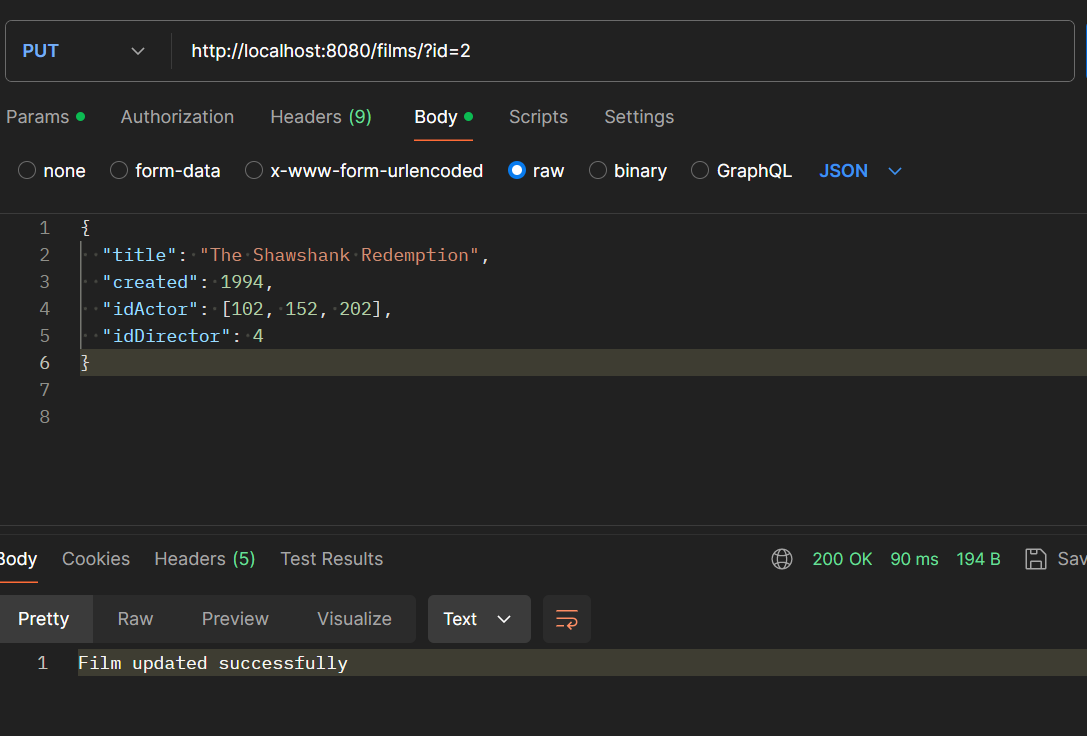




|  |  |
| --- | --- |
| # Paso: | 4 |
| Descripción | Operación de actualización correcta. |
| Resultado Esperado | La actualización se completa correctamente obteniendo respuesta HTTP 200 OK. La actualización en la base de datos se completa correctamente |
| OPs |  |
| O. Básica Bancaria |  |

Realizamos una petición REST de tipo PUT la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/?id=2 |



Con el siguiente body

{

  "title": "The Shawshank Redemption",

  "created": 1994,

  "idActor": [102, 152, 202],

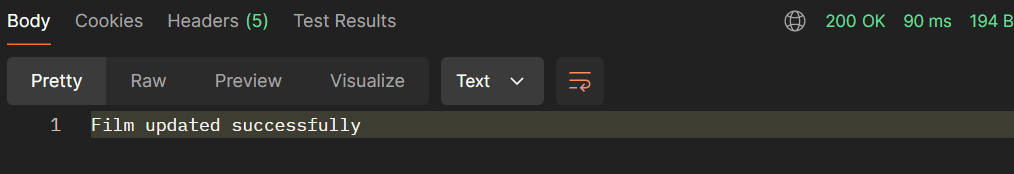
  "idDirector": 4

}



Revisamos la salida:

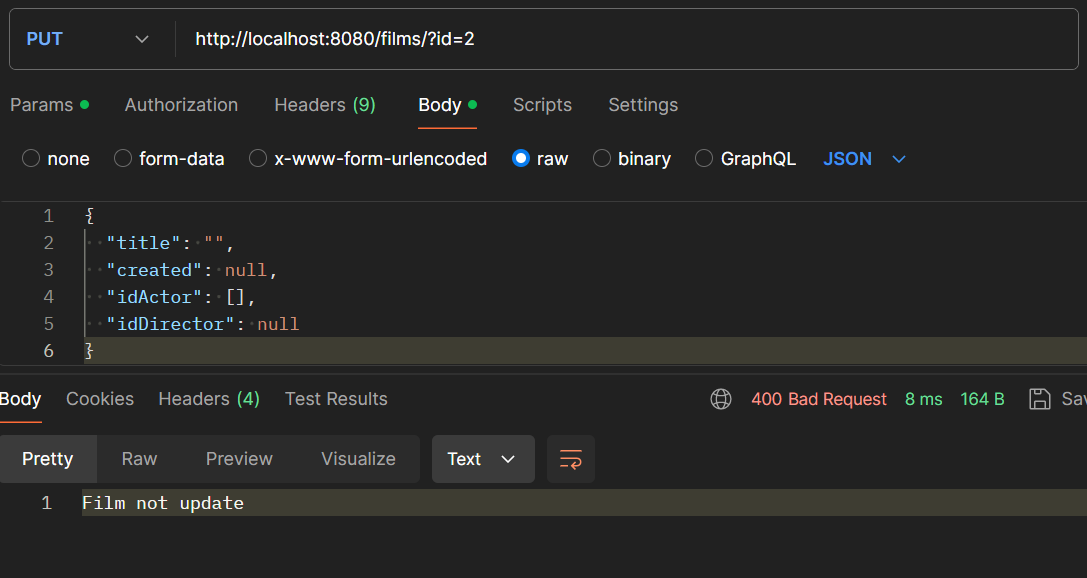




|  |  |
| --- | --- |
| # Paso: | 5 |
| Descripción | Operación de actualización incorrecta. |
| Resultado Esperado | La actualización se completa incorrectamente obteniendo respuesta HTTP 400 BAD REQUEST. La actualización en la base de datos no se realiza |
| OPs |  |
| O. Básica Bancaria |  |

Realizamos una petición REST de tipo PUT la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/?id=2 |



Con el siguiente body

{

  "title": "",

  "created": null,

  "idActor": [],

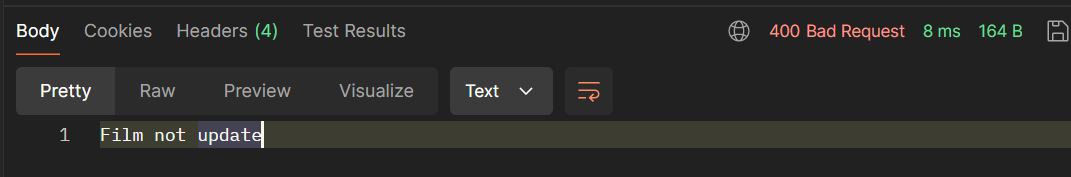
  "idDirector": null

}



Revisamos la salida:

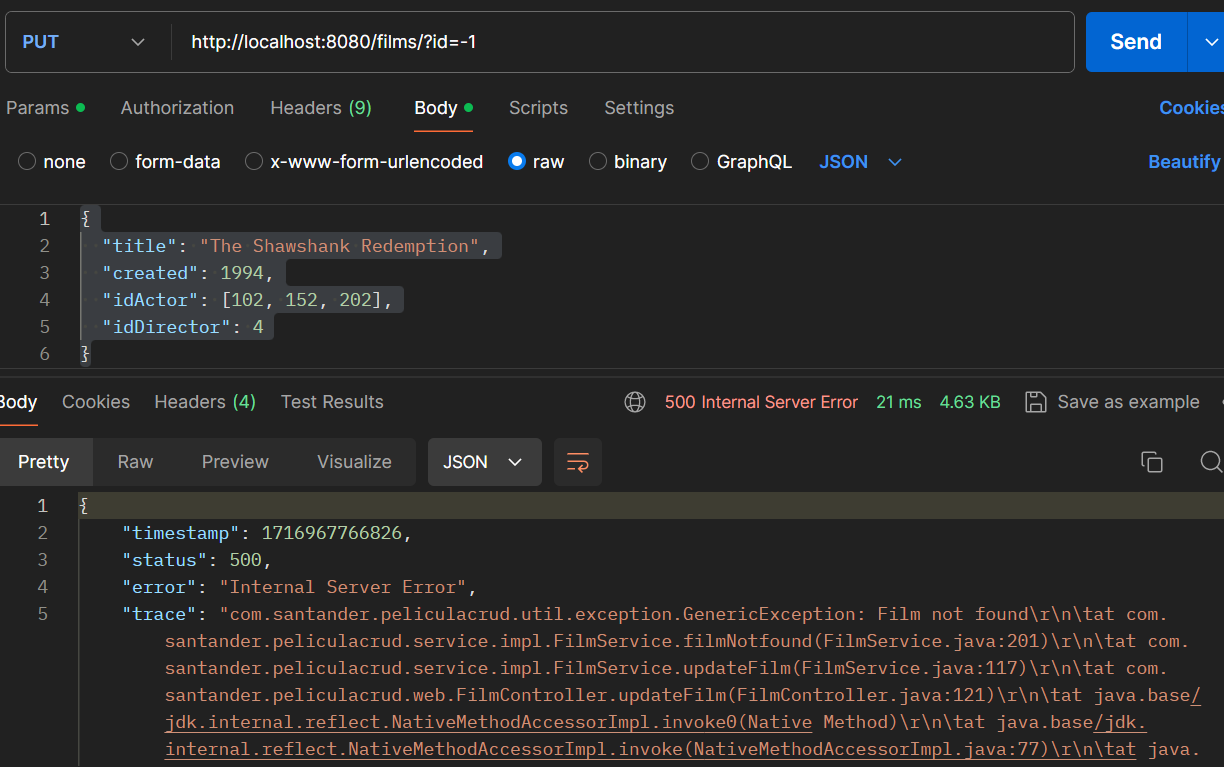




|  |  |
| --- | --- |
| # Paso: | 6 |
| Descripción | Operación de actualización incorrecta. |
| Resultado Esperado | La actualización se completa incorrectamente obteniendo respuesta HTTP 500 Internal Server Error. La actualización en la base de datos no se realiza. |
| OPs |  |
| O. Básica Bancaria |  |

Realizamos una petición REST de tipo PUT la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/?id=-1 |



Con el siguiente body

{

  "title": "The Shawshank Redemption",

  "created": 1994,

  "idActor": [102, 152, 202],

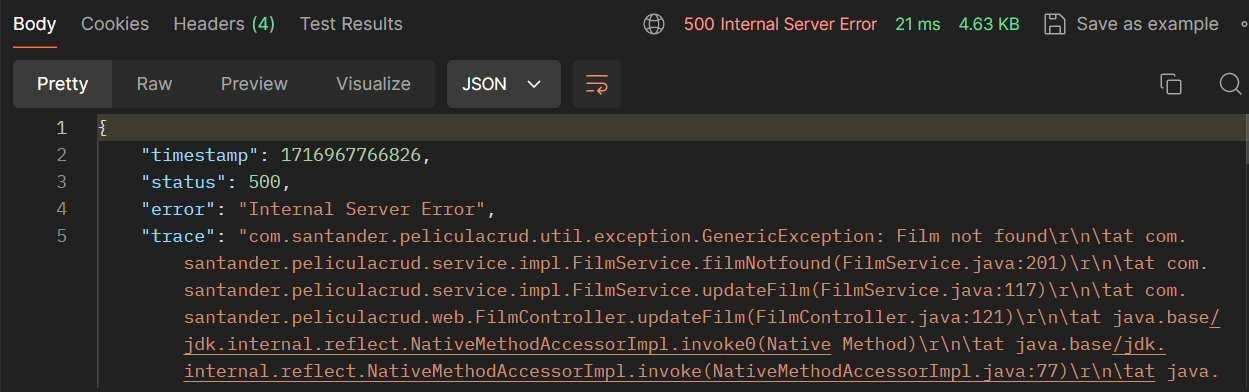
  "idDirector": 4

}



Revisamos la salida:

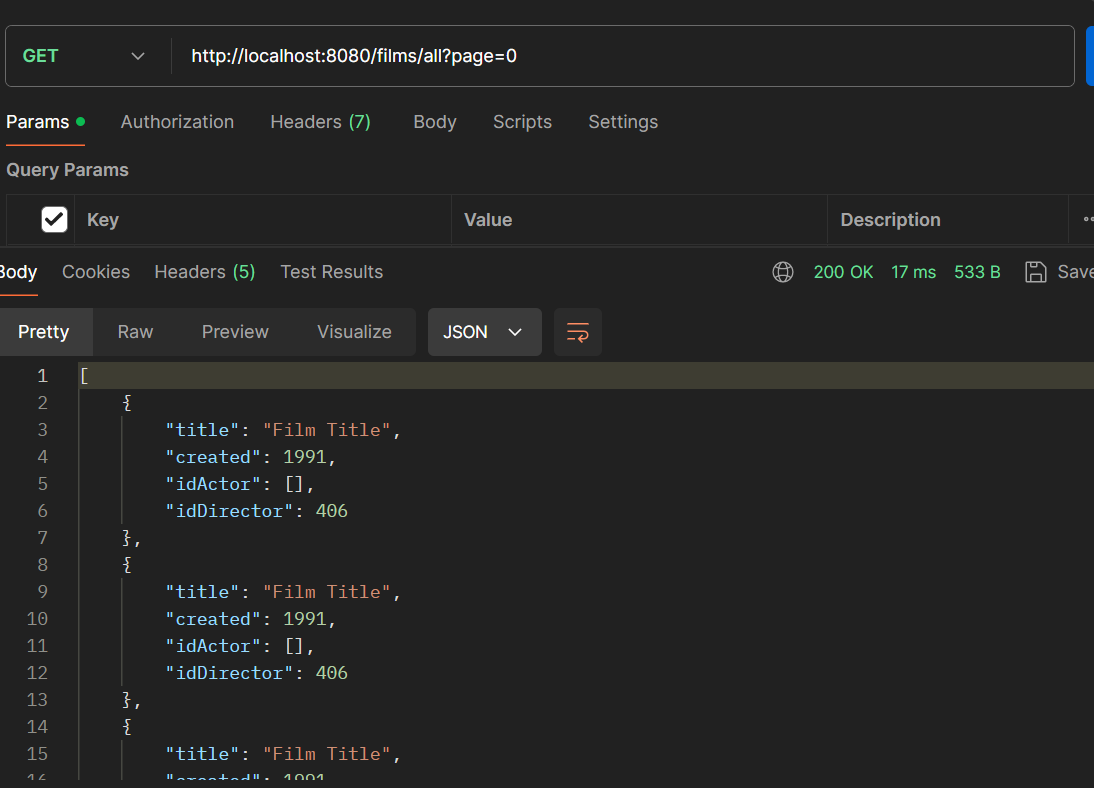




|  |  |
| --- | --- |
| # Paso: | 7 |
| Descripción | Operación de consulta correcta. |
| Resultado Esperado | La consulta se completa correctamente obteniendo respuesta HTTP 200 OK. La consulta de datos en las tablas se realiza correctamente, mostrando todas las películas. |
| OPs |  |
| O. Básica Bancaria |  |

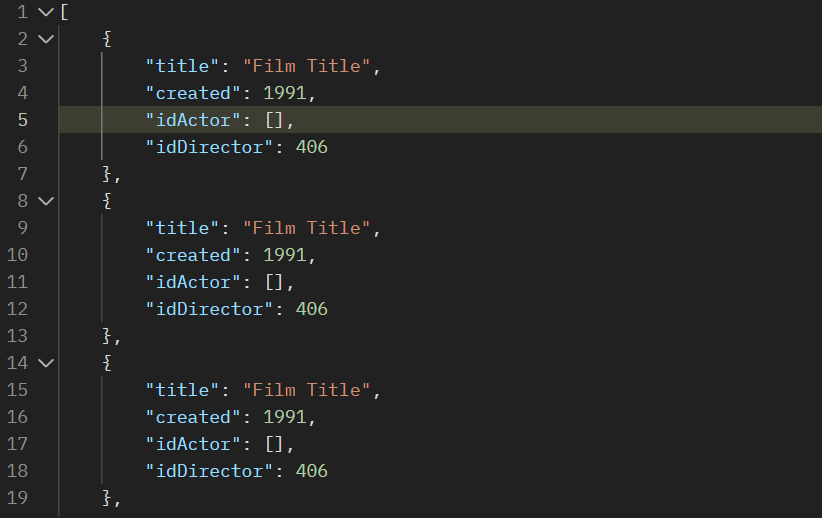
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/all?page=0 |



Revisamos la salida:

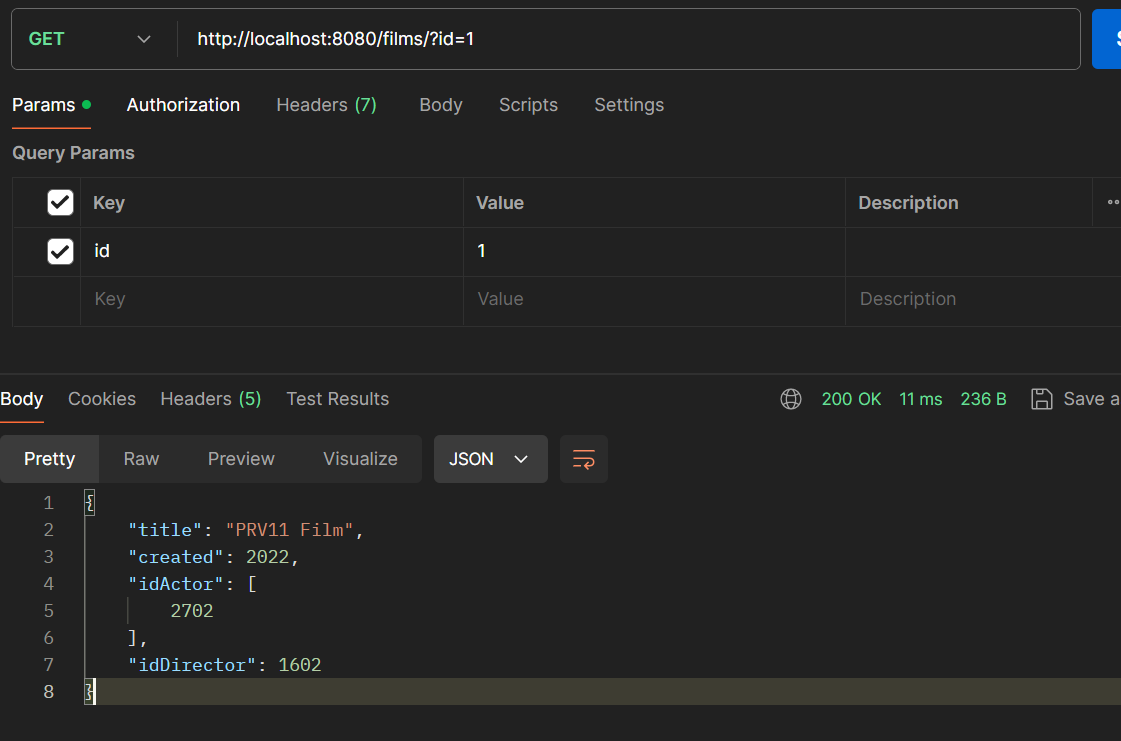




|  |  |
| --- | --- |
| # Paso: | 8 |
| Descripción | Operación de consulta correcta. |
| Resultado Esperado | La consulta se completa correctamente obteniendo respuesta HTTP 200 OK. La consulta de datos en las tablas se realiza correctamente, mostrando la película con el id indicado. |
| OPs |  |
| O. Básica Bancaria |  |

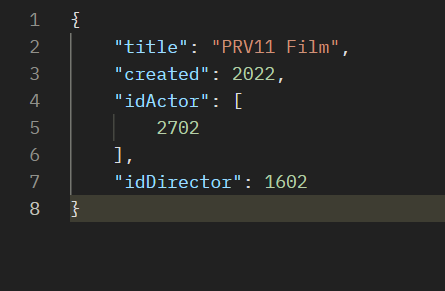
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/?id=1 |



Revisamos la salida:

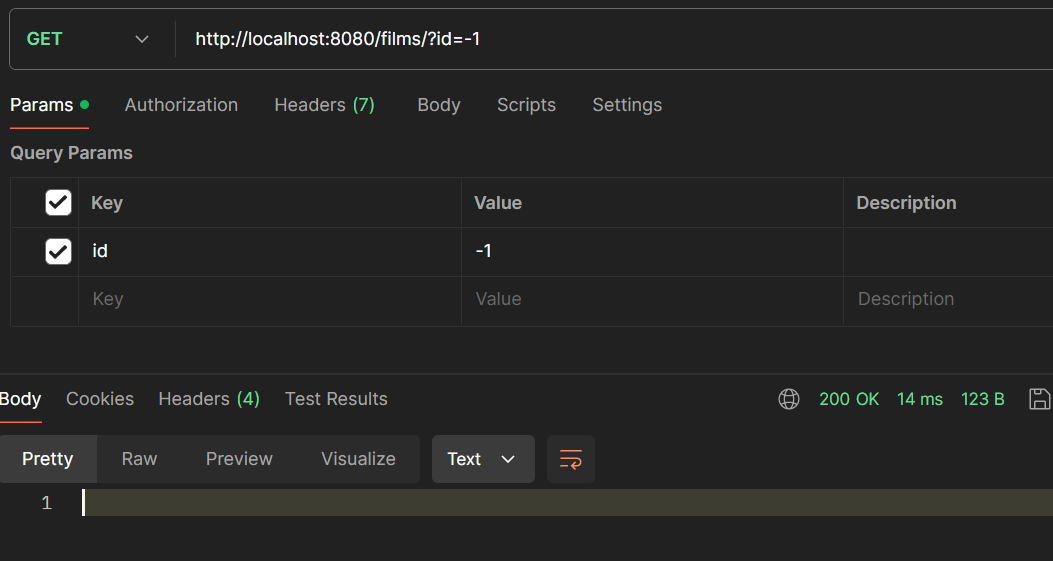




|  |  |
| --- | --- |
| # Paso: | 9 |
| Descripción | Operación de consulta incorrecta. |
| Resultado Esperado | La consulta se completa correctamente obteniendo respuesta HTTP 200 Ok. A pesar de ello, al no existir la películar54 con ese id no devuelve ningún objeto. |
| OPs |  |
| O. Básica Bancaria |  |

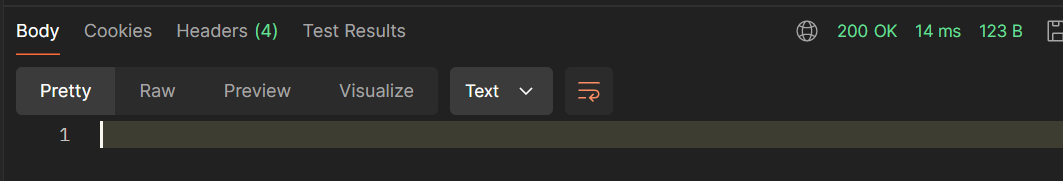
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/?id=-1 |



Revisamos la salida:

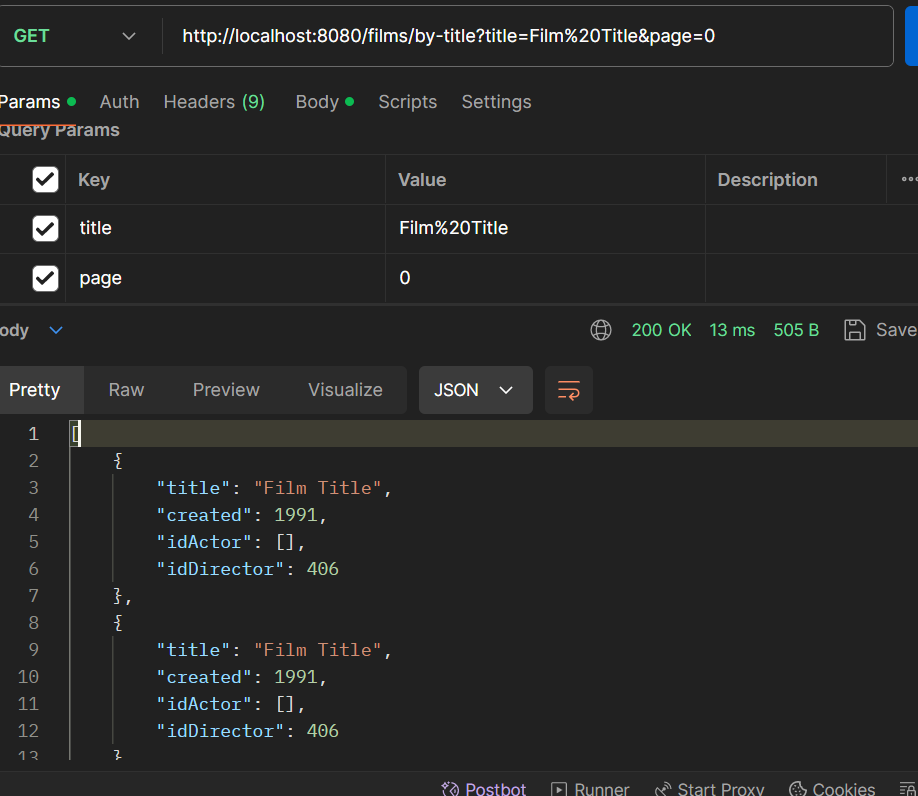




|  |  |
| --- | --- |
| # Paso: | 10 |
| Descripción | Operación de consulta correcta. |
| Resultado Esperado | La consulta se completa correctamente obteniendo respuesta HTTP 200 OK. La consulta de datos en las tablas se realiza correctamente, mostrando las películas con el título indicado. |
| OPs |  |
| O. Básica Bancaria |  |

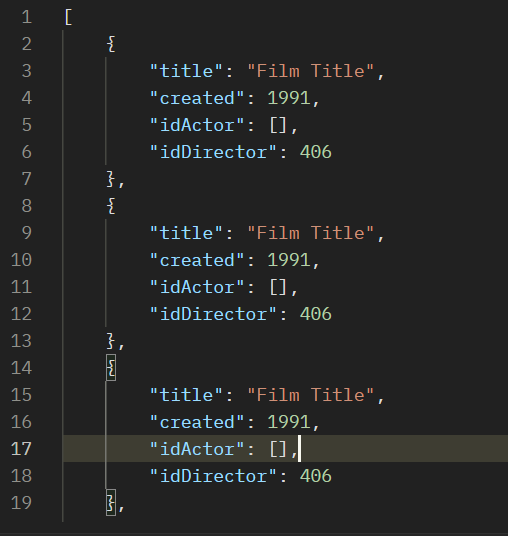
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/by-title?title=Film%20Title&page=0 |



Revisamos la salida:

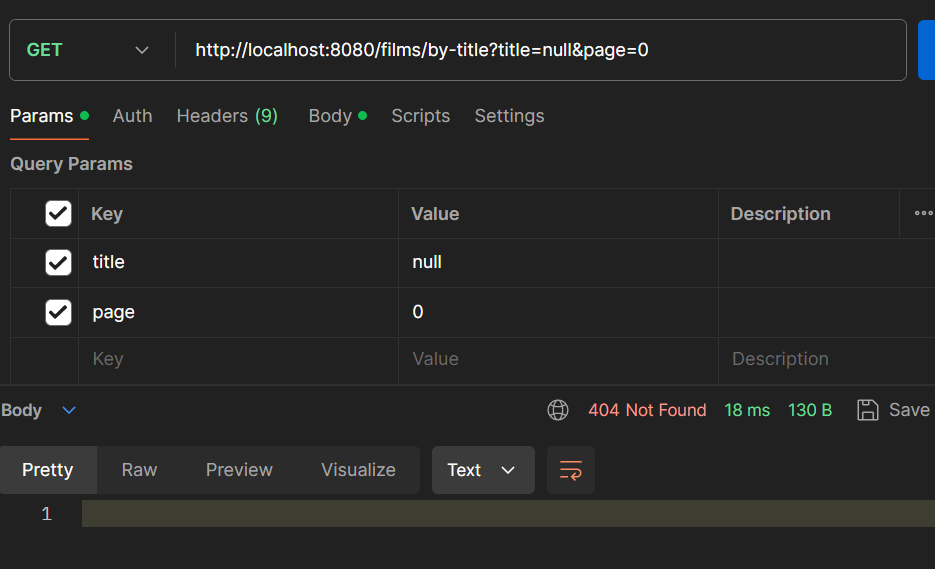




|  |  |
| --- | --- |
| # Paso: | 11 |
| Descripción | Operación de consulta incorrecta. |
| Resultado Esperado | La consulta no se completa correctamente obteniendo respuesta HTTP 404 Bad Request. |
| OPs |  |
| O. Básica Bancaria |  |

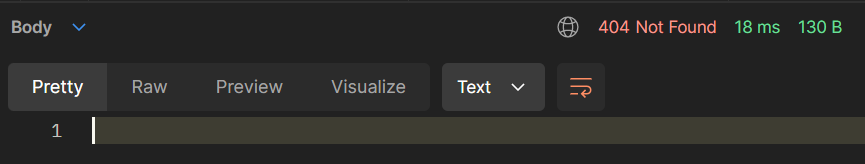
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/by-title?title=null&page=0 |



Revisamos la salida:

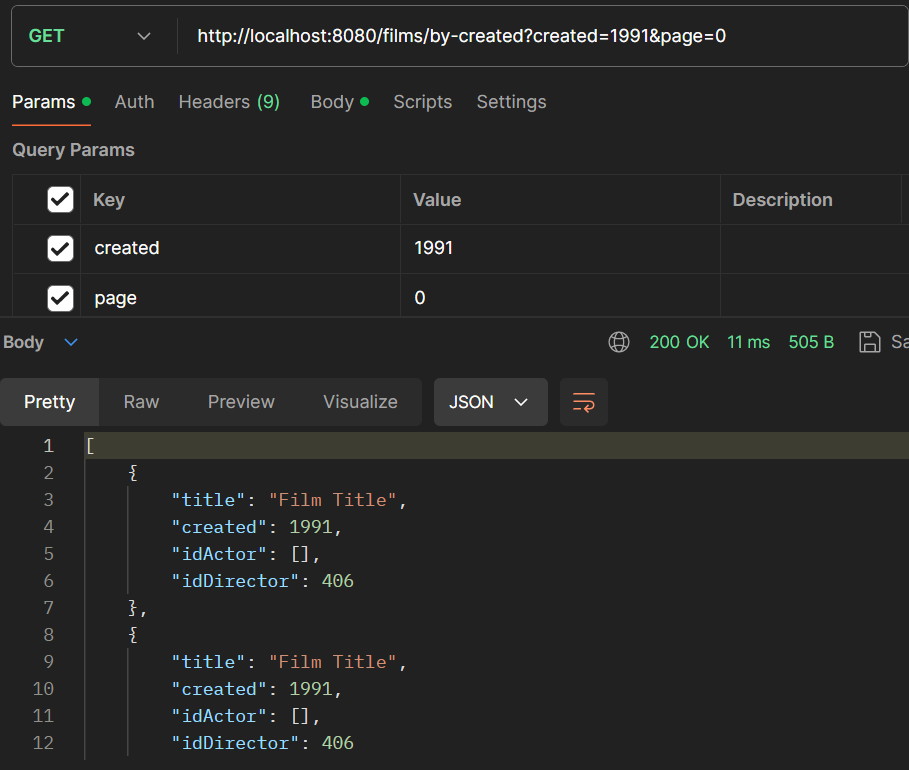




|  |  |
| --- | --- |
| # Paso: | 12 |
| Descripción | Operación de consulta correcta. |
| Resultado Esperado | La consulta se completa correctamente obteniendo respuesta HTTP 200 OK. La consulta de datos en las tablas se realiza correctamente, mostrando la película creada con el año indicado. |
| OPs |  |
| O. Básica Bancaria |  |

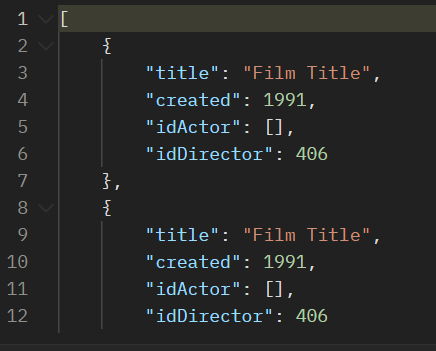
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/by-created?created=1991&page=0 |



Revisamos la salida:

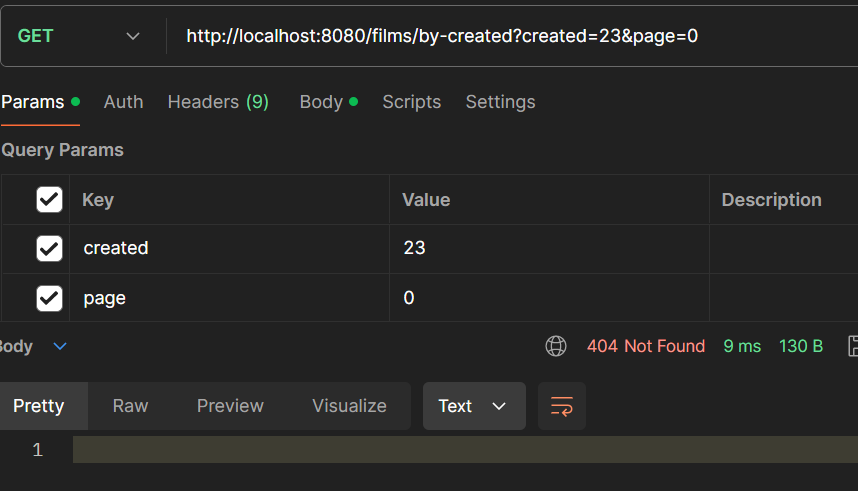




|  |  |
| --- | --- |
| # Paso: | 13 |
| Descripción | Operación de consulta incorrecta. |
| Resultado Esperado | La consulta no se completa correctamente obteniendo respuesta HTTP 404 Not Found. |
| OPs |  |
| O. Básica Bancaria |  |

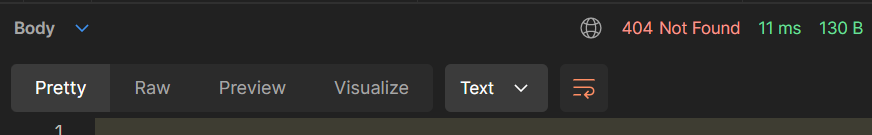
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/by-created?created=23&page=0 |



Revisamos la salida:

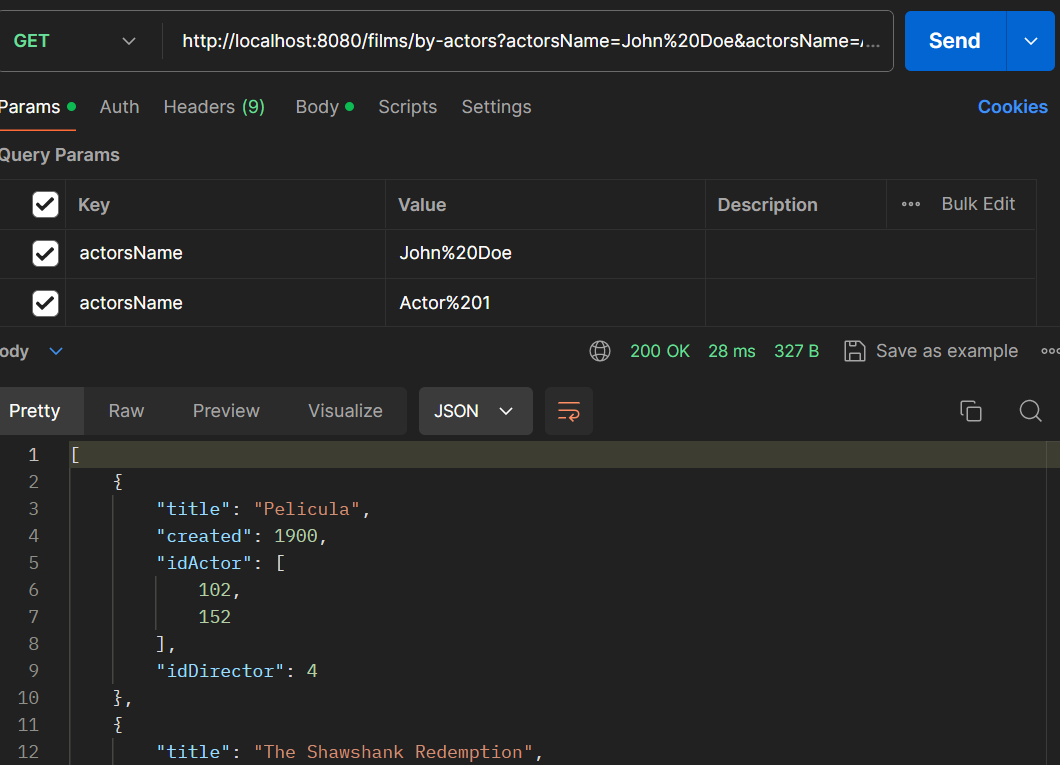




|  |  |
| --- | --- |
| # Paso: | 14 |
| Descripción | Operación de consulta correcta. |
| Resultado Esperado | La consulta se completa correctamente obteniendo respuesta HTTP 200 OK. La consulta de datos en las tablas se realiza correctamente, mostrando la película con los actores indicados. |
| OPs |  |
| O. Básica Bancaria |  |

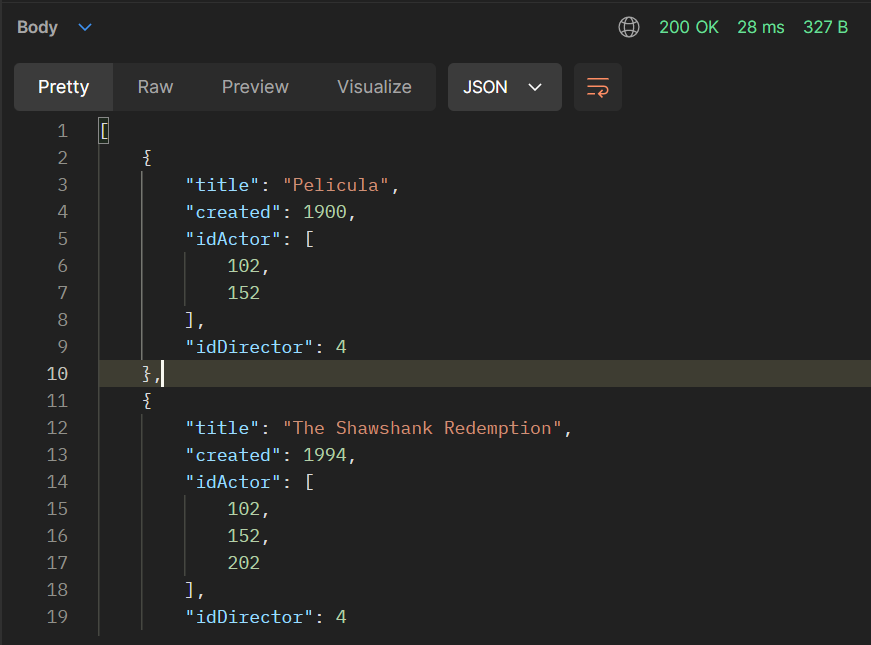
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/by-actors?actorsName=John%20Doe&actorsName=Actor%201&page=0 |



Revisamos la salida:

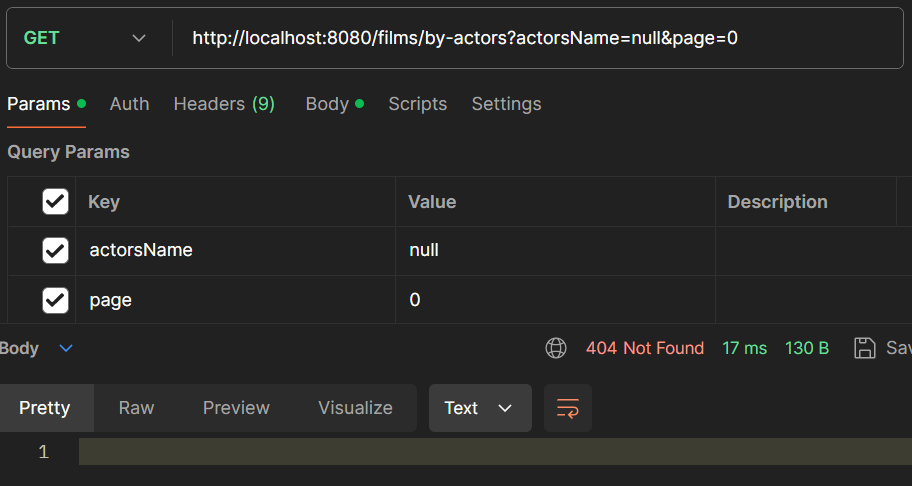




|  |  |
| --- | --- |
| # Paso: | 15 |
| Descripción | Operación de consulta incorrecta. |
| Resultado Esperado | La consulta no se completa correctamente obteniendo respuesta HTTP 404 Not Found. |
| OPs |  |
| O. Básica Bancaria |  |

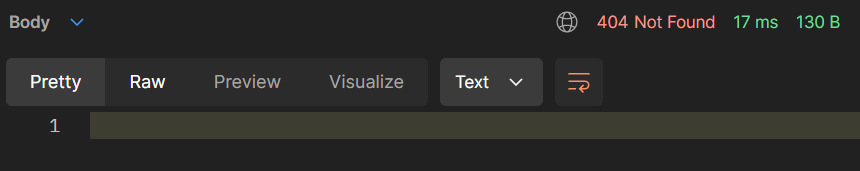
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/by-actors?actorsName=null&page=0 |



Revisamos la salida:

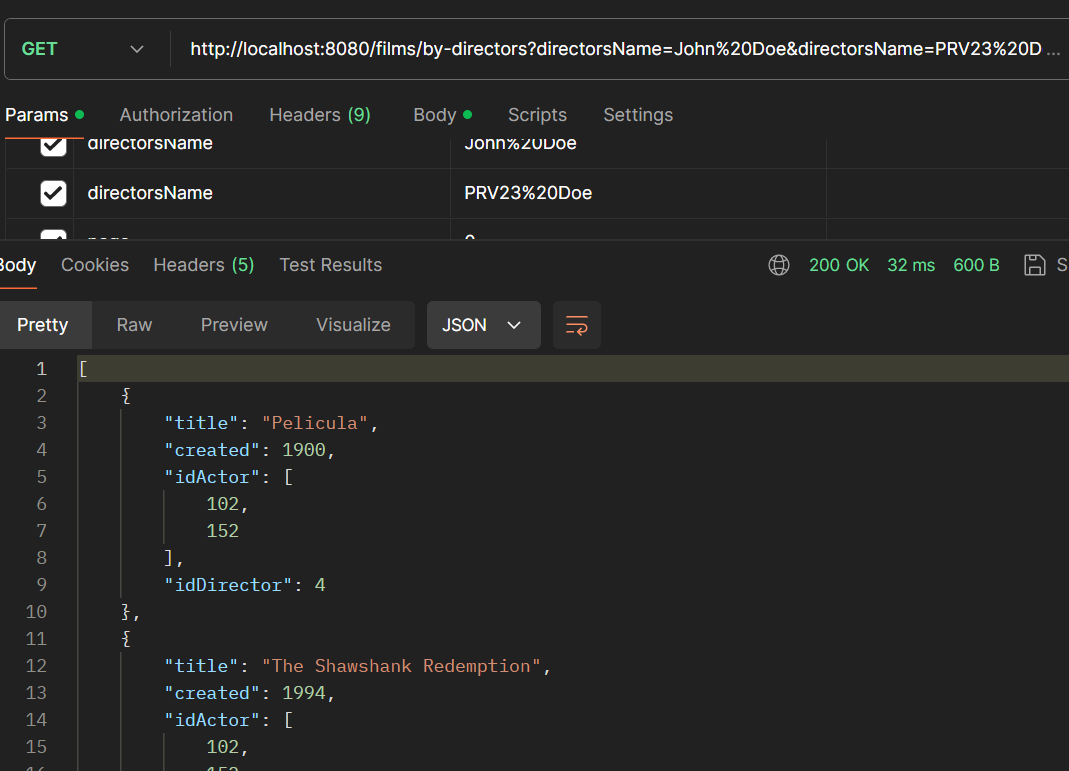




|  |  |
| --- | --- |
| # Paso: | 16 |
| Descripción | Operación de consulta correcta. |
| Resultado Esperado | La consulta se completa correctamente obteniendo respuesta HTTP 200 OK. La consulta de datos en las tablas se realiza correctamente, mostrando las películas con los directores indicados. |
| OPs |  |
| O. Básica Bancaria |  |

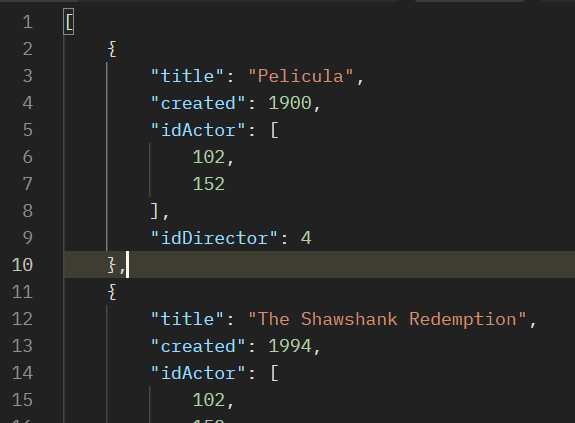
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/by-directors?directorsName=John%20Doe&directorsName=PRV23%20Doe&page=0 |



Revisamos la salida:

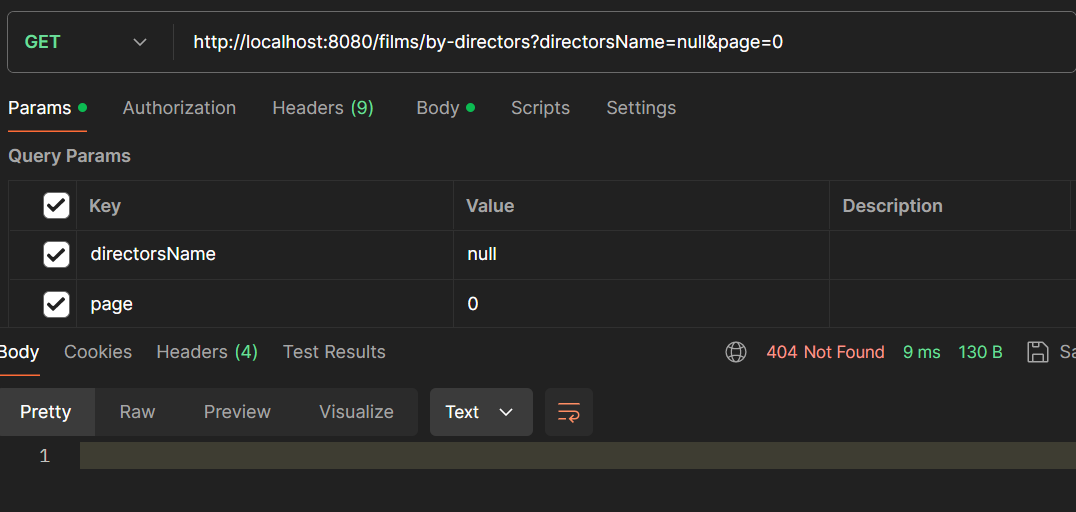




|  |  |
| --- | --- |
| # Paso: | 17 |
| Descripción | Operación de consulta incorrecta. |
| Resultado Esperado | La consulta no se completa correctamente obteniendo respuesta HTTP 404 Not Found. |
| OPs |  |
| O. Básica Bancaria |  |

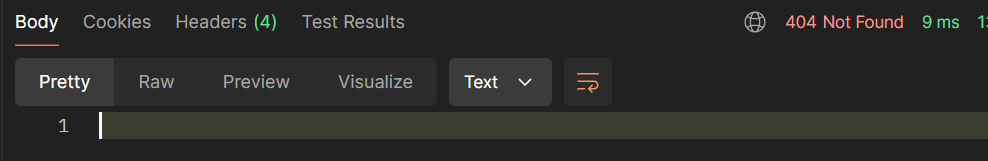
Realizamos una petición REST de tipo GET a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/by-directors?directorsName=null&page=0 |



Revisamos la salida:

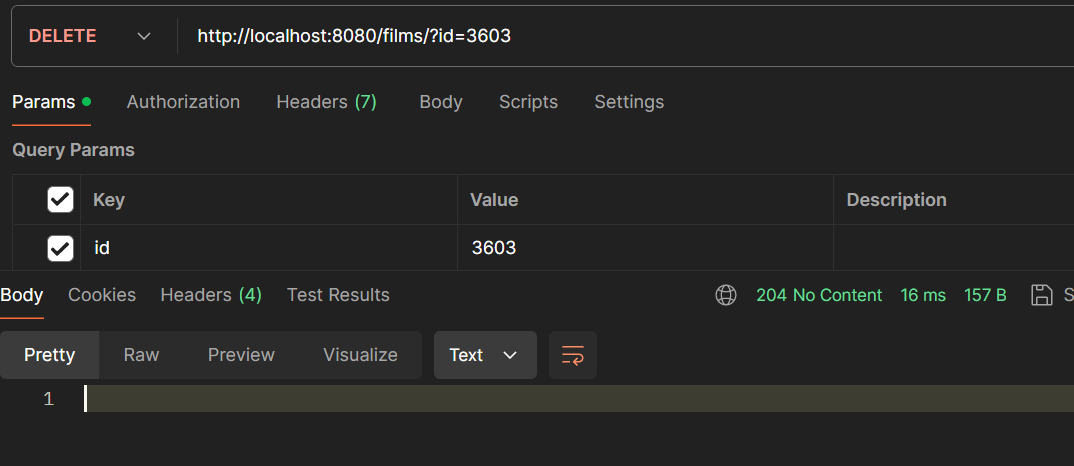




|  |  |
| --- | --- |
| # Paso: | 18 |
| Descripción | Operación de eliminación correcta. |
| Resultado Esperado | El borrado se completa correctamente obteniendo respuesta HTTP 204 No Content. La eliminación de la película se realiza correctamente. |
| OPs |  |
| O. Básica Bancaria |  |

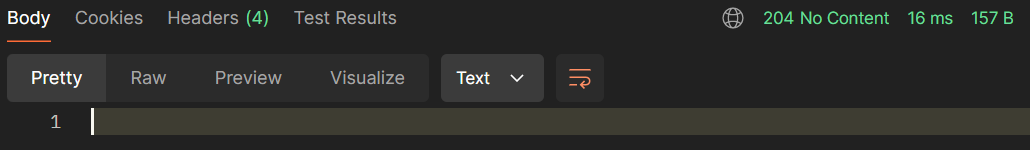
Realizamos una petición REST de tipo DELETE a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/?id=3603 |



Revisamos la salida:

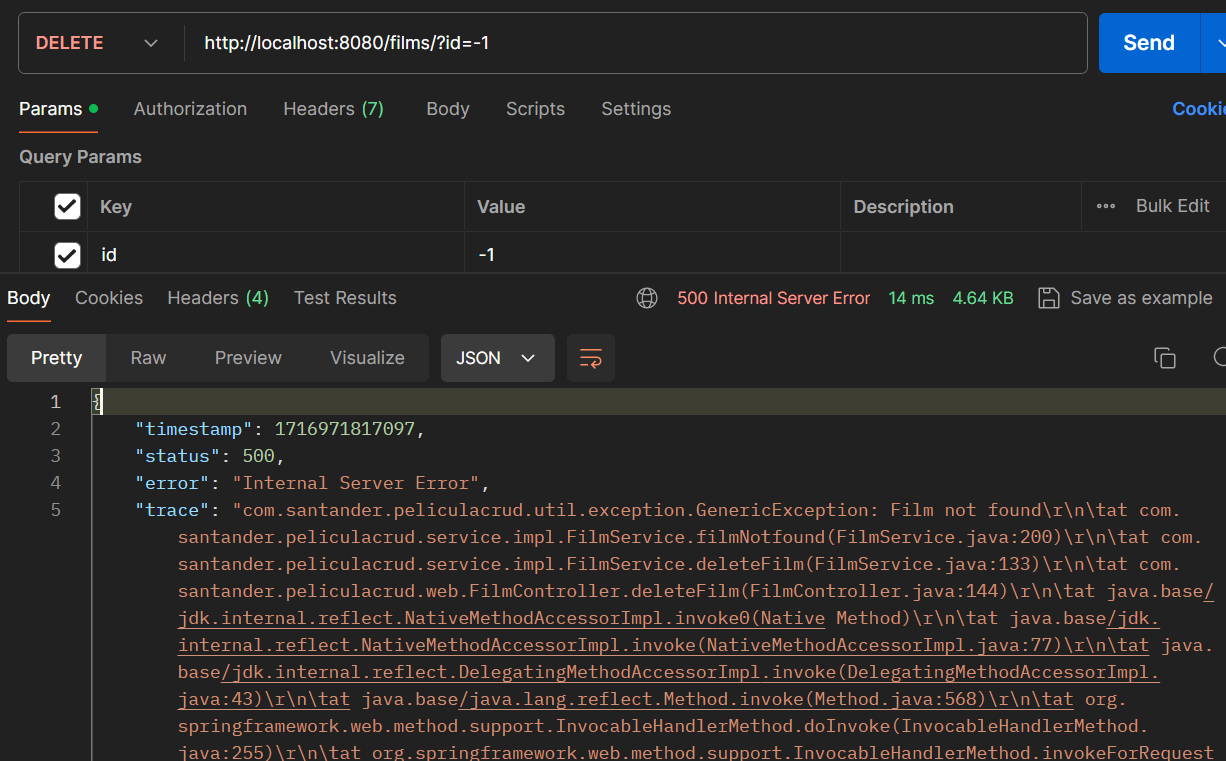




|  |  |
| --- | --- |
| # Paso: | 19 |
| Descripción | Operación de eliminación incorrecta. |
| Resultado Esperado | La eliminación no se completa correctamente obteniendo respuesta HTTP 500 Internal Server Error. |
| OPs |  |
| O. Básica Bancaria |  |

Realizamos una petición REST de tipo DELETE a la siguiente URL:

|  |
| --- |
| http://localhost:8080/films/?id=-1 |



Revisamos la salida:



