

# CRUD Operations

<b>Non-REST</b>	<b>2</b>
JSON	2
Get all films	2
Get film by title	3
Create new film	4
Update film	5
Delete film	6
XML	7
Get all films	7
Get film by title	8
Create new film	9
Update film	10
Delete film	11
CSV	12
Get all films	12
Get film by title	13
Create new film	14
Update film	15
Delete film	16
<b>REST</b>	<b>17</b>
JSON	17
Get all films	17
Get film by title	18
Create new film	19
Update film	20
Delete film	22
XML	23
Get all films	23
Get film by title	24
Create new film	25
Update film	26
Delete film	27
CSV	28
Get all films	28
Get film by title	29
Create new film	30
Update film	31
Delete film	32

# Non-REST

## JSON

### Get all films

**Request:**

**Format:**

Use REST servlet

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

**GET FILMS**

Films

Network tab in Chrome DevTools showing a successful GET request for `getAllFilms?format=json`. The response payload is displayed as follows:

```

Request URL: https://servletcoursework-336513.nw.r.appspot.com/getAllFilms?format=json
Request Method: GET
Status Code: 200
Remote Address: 142.250.200.52:443
Referrer Policy: strict-origin-when-cross-origin

Response Headers
access-control-allow-credentials: true
access-control-allow-headers: Content-Type

[{"id": 10003, "title": "9 1/2 WEEKS", "year": 1986, "director": "ADRIAN LYNE", ...}, ...
  0: {"id": 10003, "title": "9 1/2 WEEKS", "year": 1986, "director": "ADRIAN LYNE", ...}
    director: "ADRIAN LYNE"
    id: 10003
    review: "Director Lyne shows his pop video roots in this erotic tale reputedly based on a true story. B stars: \"MICKEY ROURKE, KIM BASINGER\""
    title: "9 1/2 WEEKS"
    year: 1986
  1: {"id": 10004, "title": "976-EVIL II: THE RETURN", "year": 1988, "director": "JIM WYNORSKI", ...}
  2: {"id": 10005, "title": "A BETTER TOMORROW", "year": 1986, "director": "JOHN WOO", ...}
  3: {"id": 10006, "title": "A BRONX TALE", "year": 1993, "director": "ROBERT DE NIRO", ...}
  ...
]
  
```

**Response:**

**Format:**

Use REST servlet

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

**GET FILMS**

Films

Network tab in Chrome DevTools showing a successful GET request for `getAllFilms?format=json`. The response payload is displayed as follows:

```

Request URL: https://servletcoursework-336513.nw.r.appspot.com/getAllFilms?format=json
Request Method: GET
Status Code: 200
Remote Address: 142.250.200.52:443
Referrer Policy: strict-origin-when-cross-origin

Response Headers
access-control-allow-credentials: true
access-control-allow-headers: Content-Type

[{"id": 10003, "title": "9 1/2 WEEKS", "year": 1986, "director": "ADRIAN LYNE", ...}, ...
  0: {"id": 10003, "title": "9 1/2 WEEKS", "year": 1986, "director": "ADRIAN LYNE", ...}
    director: "ADRIAN LYNE"
    id: 10003
    review: "Director Lyne shows his pop video roots in this erotic tale reputedly based on a true story. B stars: \"MICKEY ROURKE, KIM BASINGER\""
    title: "9 1/2 WEEKS"
    year: 1986
  1: {"id": 10004, "title": "976-EVIL II: THE RETURN", "year": 1988, "director": "JIM WYNORSKI", ...}
  2: {"id": 10005, "title": "A BETTER TOMORROW", "year": 1986, "director": "JOHN WOO", ...}
  3: {"id": 10006, "title": "A BRONX TALE", "year": 1993, "director": "ROBERT DE NIRO", ...}
  ...
]
  
```

## Get film by title

### Request:

JSON XML Text

Get Get Add  
all film new  
films by film  
Update Delete  
film film

Title Alien

**GET FILM(S)**

Films:

Review

1 requests | 1.6 kB transferred | 2.9 kB resources

### Response:

#### Format:

Use REST servlet

JSON XML Text

Get all films  
Get film by title  
Add new film  
Update film  
Delete film

Title Alien

Network

Request URL: https://servetcoursework-336513.nw.r.appspot.com/getFilmByTitle?format=json&title=Alien

Request Method: GET

Status Code: 200

Remote Address: 142.250.200.52:443

Referrer Policy: strict-origin-when-cross-origin

Response Headers

access-control-allow-credentials: true  
access-control-allow-headers: Content-Type

1 requests | 1.6 kB transferred | 2.9 kB resources

```
[{"id": 10044, "title": "ALIEN", "year": 1979, "director": "RIDLEY SCOTT", "stars": "SIGOURNEY WEAVER, IAN HOLM", "review": "The original and many believe still the best, the terrifying tale of what happens when the crew stars: SIGOURNEY WEAVER, IAN HOLM", "title": "ALIEN", "year": 1979}, {"id": 10045, "title": "ALIEN 3", "year": 1992, "director": "DAVID FINCHER", "stars": "LOU DI LAI, ROBERT DUVALL, RICHARD DREYFUS, ROBERT PATTINSON", "review": "A prequel to the original ALIEN, which was a cult classic in its own right, but it failed to live up to the original's terror and suspense.", "title": "ALIEN 3", "year": 1992}, {"id": 10046, "title": "ALIEN NATION", "year": 1988, "director": "GRAHAM BAKER", "stars": "CHRIS REEVE, JEFFREY HARRIS, KATHLEEN TURNER", "review": "A spin-off of the original ALIEN movie, set in a different universe where Earth is invaded by alien creatures.", "title": "ALIEN NATION", "year": 1988}, {"id": 10047, "title": "ALIEN SPECIES", "year": 1995, "director": "DAVE PAYNE", "stars": "CHRISTOPHER LEE, JULIA ROBERTS, ROBERT DUVALL", "review": "A prequel to the original ALIEN movie, set in a different universe where Earth is invaded by alien species.", "title": "ALIEN SPECIES", "year": 1995}, {"id": 10048, "title": "ALIENS: SPECIAL EDITION", "year": 1986, "director": "JAMES CAMERON", "stars": "MICHAEL BIEBERMAN, ROBERT DUVALL, ROBERT PATTINSON", "review": "A re-release of the original ALIEN movie, featuring enhanced special effects and a new ending.", "title": "ALIENS: SPECIAL EDITION", "year": 1986} ]
```

## Create new film

### Request:

#### Format:

Use REST servlet

JSON     XML     Text

<input type="radio"/> Get all films	<input type="radio"/> Get film by title	<input checked="" type="radio"/> Add new film
<input type="radio"/> Update film	<input type="radio"/> Delete film	

Name: insertFilm?format=json

Request URL: https://servetcoursework-336513.nw.r.appspot.com/insertFilm?format=json

Request Method: POST

Status Code: 200

Remote Address: 142.250.200.52:443

Referrer Policy: strict-origin-when-cross-origin

Headers

access-control-allow-credentials: true

#### Film attributes:

Title \_\_\_\_\_

### Request body:

Use REST servlet

JSON     XML     Text

<input type="radio"/> Get all films	<input type="radio"/> Get film by title	<input checked="" type="radio"/> Add new film
<input type="radio"/> Update film	<input type="radio"/> Delete film	

Name: insertFilm?format=json

Query String Parameters

format: json

Request Payload

```
{
  "title": "Bladerunner 2049",
  "year": "2017",
  "director": "Dennis Villenueve",
  "review": "GOAT",
  "stars": "Harrison Ford, Ryan Gosling",
  "title": "Bladerunner 2049",
  "year": "2017"
}
```

#### Film attributes:

Title \_\_\_\_\_

### Response:

Use REST servlet

JSON     XML     Text

<input type="radio"/> Get all films	<input type="radio"/> Get film by title	<input checked="" type="radio"/> Add new film
<input type="radio"/> Update film	<input type="radio"/> Delete film	

Name: insertFilm?format=json

Response

1

#### Film attributes:

Title \_\_\_\_\_

## Update film

### Request:

The screenshot shows the Network tab in Chrome DevTools. A PUT request named "updateFilm?format=json" is selected. The Headers section shows the following details:

- Request URL: https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=json
- Request Method: PUT
- Status Code: 200
- Remote Address: 142.250.200.52:443
- Referrer Policy: strict-origin-when-cross-origin

The Response Headers section includes:

- access-control-allow-credentials: true
- access-control-allow-headers: Content-Type

On the left, there is a sidebar with various options like Get all films by title, Update film, etc.

Selected film ID: 10003

**Films:**

**GET FILM DATA**

### Request body:

The screenshot shows the Network tab in Chrome DevTools. The Request Payload section displays the JSON data sent in the body of the PUT request:

```
{
  "id": 10003,
  "title": "Test",
  "year": 1986,
  "director": "ADRIAN LYNE",
  "stars": "MICKEY ROURKE, KIM BASINGER"
}
```

On the left, there is a sidebar with various options like Get all films by title, Update film, etc.

Selected film ID: 10003

**Films:**

**GET FILM DATA**

**Review**

### Response:

The screenshot shows the Network tab in Chrome DevTools. The Response section shows the status code 1.

On the left, there is a sidebar with various options like Get all films by title, Update film, etc.

Selected film ID: 10003

**Films:**

**GET FILM DATA**

**Review**

Line 1, Column 1

## Delete film

### Request:

The screenshot shows the Network tab of a browser developer tools interface. A request labeled "deleteFilm?format=json&id=10004" is selected. The Headers section shows the following details:

- Request URL: <https://servletcoursework-336513.nw.r.appspot.com/deleteFilm?format=json&id=10004>
- Request Method: DELETE
- Status Code: 200
- Remote Address: 142.250.200.20:443
- Referrer Policy: strict-origin-when-cross-origin

The Response Headers section includes:

- access-control-allow-credentials: true
- access-control-allow-headers: Content-Type
- access-control-allow-methods: DELETE
- access-control-allow-origin: \*

On the left, there is a sidebar with various options for interacting with the REST service, including "Get all films by title" and "Delete film". Below the sidebar, the text "Film ID: 10004" is displayed, followed by a blue "DELETE FILM" button.

### Response:

The screenshot shows the Network tab of a browser developer tools interface. A response labeled "deleteFilm?format=json&id=10004" is selected. The Response tab displays the following content:

```
1
```

Below the response, the text "Line 1, Column 1" is shown. On the left, there is a sidebar with various options for interacting with the REST service, including "Get all films by title" and "Delete film". Below the sidebar, the text "Film ID: 10004" is displayed, followed by a blue "DELETE FILM" button.

## XML

### Get all films

Request:

Format:  Use REST servlet  JSON  XML  Text

Get all films by title  Get new film  Add film  Update film  Delete film

**GET FILMS**

**Films:**

Request URL: https://servletcoursework-336513.nw.r.appspot.com/getAllFilms?format=xml  
Request Method: GET  
Status Code: 200  
Remote Address: 142.250.200.20:443  
Referrer Policy: strict-origin-when-cross-origin

Response Headers

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: GET
access-control-allow-origin: *
```

1 requests | 269 kB transferred | 742 kB resource

Response:

Format:  Use REST servlet  JSON  XML  Text

Get all films by title  Get new film  Add film  Update film  Delete film

**GET FILMS**

**Films:**

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <film>
    <id>10005</id>
    <title>A BETTER TOMORROW</title>
    <year>1986</year>
    <director>JOHN WOO</director>
    <stars>CHOW YUN-FAT, LESLIE CHEUNG</stars>
    <review>The film which brought international stardom to star Chow Yun-Fat and director John Woo. A roller-coaster ride of action and suspense, it's a must-see for any fan of Hong Kong cinema.</review>
  </film>
  <film>
    <id>10006</id>
    <title>A BRONX TALE</title>
    <year>1993</year>
    <director>ROBERT DE NIRO</director>
    <stars>ROBERT DE NIRO, CHAZZ PALMINTERI</stars>
  </film>
</root>
```

1 requests | 269 kB transferred | 742 kB resource

## Get film by title

### Request:

#### Format:

Use REST servlet

JSON

XML

Text

Get all films

Get film by title

Add new film

Update film

Delete film

Title: Alien

**GET FILM(S)**

Network tab selected in developer tools.

Request URL: https://servetcoursework-336513.nw.r.appspot.com/getFilmByTitle?format=xml&title=Alien

Request Method: GET

Status Code: 200

Remote Address: 142.250.200.20:443

Referrer Policy: strict-origin-when-cross-origin

**Response Headers**

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: GET
access-control-allow-origin: *
```

1 requests | 1.7 kB transferred | 3.4 kB resources

### Response:

#### Format:

Use REST servlet

JSON

XML

Text

Get all films

Get film by title

Add new film

Update film

Delete film

Title: Alien

**GET FILM(S)**

Network tab selected in developer tools.

Name: getFilmByTitle?format=xml&title=Alien

Headers tab selected in developer tools.

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <film>
    <id>10044</id>
    <title>ALIEN</title>
    <year>1979</year>
    <director>RIDLEY SCOTT</director>
    <stars>SIGOURNEY WEAVER, IAN HOLM</stars>
    <review>The original and many believe still the best, the terrifying tale of what happens when the crew of the space freighter Nostromo encounter an alien life form that has been dormant in one of the ship's storage tanks since the 19th century. The movie is a masterpiece of practical effects and tension, with a score by John Williams that is now iconic. The performances are excellent, particularly Sigourney Weaver as Ripley, who becomes the last line of defense against the alien. The film is a classic of science fiction horror and is considered a cult classic.</review>
  </film>
  <film>
    <id>10045</id>
    <title>ALIEN 3</title>
    <year>1992</year>
    <director>DAVID FINCHER</director>
    <stars>SIGOURNEY WEAVER, CHARLES DANCE</stars>
```

Line 1, Column 1

## Create new film

### Request:

The screenshot shows the Network tab in Chrome DevTools. A POST request named "insertFilm?format=xml" is selected. The Headers section shows the following:

- Request URL: https://servletcoursework-336513.nw.r.appspot.com/insertFilm?format=xml
- Request Method: POST
- Status Code: 200
- Remote Address: 142.250.200.20:443
- Referrer Policy: strict-origin-when-cross-origin

The Response Headers section includes:

- access-control-allow-credentials: true
- access-control-allow-headers: Content-Type
- access-control-allow-methods: POST
- access-control-allow-origin: \*

**Film attributes:**

- Title: 2001: A Space Odyssey
- Year: 1968
- Director: Stanley Kubrick
- Stars: Keir Dullea

Request body:

The screenshot shows the Network tab in Chrome DevTools. A POST request named "insertFilm?format=xml" is selected. The Payload section shows the XML payload:

```
<Film><title>2001: A Space Odyssey</title><year>1968</year><director>Stanley Kubrick</director><stars>Keir Dullea</stars><review>My favourite film</review></Film>
```

**Film attributes:**

- Title: 2001: A Space Odyssey
- Year: 1968
- Director: Stanley Kubrick
- Stars: Keir Dullea

### Request body:

The screenshot shows the Network tab in Chrome DevTools. A POST request named "insertFilm?format=xml" is selected. The Payload section shows the XML payload:

```
<Film><title>2001: A Space Odyssey</title><year>1968</year><director>Stanley Kubrick</director><stars>Keir Dullea</stars><review>My favourite film</review></Film>
```

**Film attributes:**

- Title: 2001: A Space Odyssey
- Year: 1968
- Director: Stanley Kubrick
- Stars: Keir Dullea

### Response:

The screenshot shows the Network tab in Chrome DevTools. A POST request named "insertFilm?format=xml" is selected. The Response section shows the status code 1.

**Film attributes:**

- Title: 2001: A Space Odyssey
- Year: 1968
- Director: Stanley Kubrick
- Stars: Keir Dullea

Line 1, Column 1

## Update film

### Request:

The screenshot shows the Network tab of a browser's developer tools. A PUT request is selected, with the URL being `https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=xml`. The Headers section shows the following:

- Request URL: `https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=xml`
- Request Method: PUT
- Status Code: 200
- Remote Address: 142.250.200.20:443
- Referrer Policy: strict-origin-when-cross-origin

The Payload section contains the XML payload for updating the film:

```
<film><id>10005</id><title>Test</title><year>1986</year><director>JOHN WOO</director><stars>CHOW YUN-FAT, LESLIE CHEUNG</stars><review>The film which brought international stardom to star Chow Yun-Fat and director John Woo. A roller-coaster of a ride with brothers on both sides of the law becoming united in their quest for personal revenge. After forger Mark (Chow) is crippled and double crossed by an associate, cop Kit (Cheung) risks his badge to help him. Violent and gripping. Also available as a boxed set with a supplementary documentary and booklet.</review></film>
```

Below the Network tab, the main page content shows the updated film entry with ID 10005.

### Request body:

The screenshot shows the Network tab of a browser's developer tools. A PUT request is selected, with the URL being `https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=xml`. The Headers section shows the following:

- Request URL: `https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=xml`
- Request Method: PUT
- Status Code: 200
- Remote Address: 142.250.200.20:443
- Referrer Policy: strict-origin-when-cross-origin

The Payload section contains the XML payload for updating the film:

```
<film><id>10005</id><title>Test</title><year>1986</year><director>JOHN WOO</director><stars>CHOW YUN-FAT, LESLIE CHEUNG</stars><review>The film which brought international stardom to star Chow Yun-Fat and director John Woo. A roller-coaster of a ride with brothers on both sides of the law becoming united in their quest for personal revenge. After forger Mark (Chow) is crippled and double crossed by an associate, cop Kit (Cheung) risks his badge to help him. Violent and gripping. Also available as a boxed set with a supplementary documentary and booklet.</review></film>
```

Below the Network tab, the main page content shows the updated film entry with ID 10005.

### Response:

The screenshot shows the Network tab of a browser's developer tools. A response is selected, with the URL being `https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=xml`. The Response section shows the following:

Name	Headers	Payload	Preview	Response	Initiator	Timing
<code>updateFilm?format=xml</code>				1		
<code>updateFilm?format=xml</code>						

Below the Network tab, the main page content shows the updated film entry with ID 10005.

## Delete film

### Request:

Format:

- Use REST servlet
- JSON
- XML
- Text

Operations:

- Get all films
- Get film by title
- Add new film
- Update film
- Delete film

Film ID: 10005

**DELETE FILM**

The screenshot shows the Network tab of the Chrome DevTools. A single request is listed under '3rd-party requests': 'deleteFilm?format=xml&id=10005'. The 'General' section shows the Request URL as 'https://servletcoursework-336513.nw.r.appspot.com/deleteFilm?format=xml&id=10005', Request Method as 'DELETE', Status Code as '200', and Remote Address as '142.250.200.20:443'. The 'Response Headers' section includes 'access-control-allow-credentials: true', 'access-control-allow-headers: Content-Type', 'access-control-allow-methods: DELETE', and 'access-control-allow-origin: \*'. At the bottom of the Network tab, it says '2 requests | 21 B transferred | 1 B resources'.

### Response:

Format:

- Use REST servlet
- JSON
- XML
- Text

Operations:

- Get all films
- Get film by title
- Add new film
- Update film
- Delete film

Film ID: 10005

**DELETE FILM**

The screenshot shows the Network tab of the Chrome DevTools. A single response is listed under '3rd-party requests': 'deleteFilm?format=xml&id=10005'. The 'Response' section shows the status as '1'. At the bottom of the Network tab, it says '2 requests | 21 B transferred | 1 B resources'.

# CSV

## Get all films

Request:

**Format:**

- Use REST servlet
- JSON
- XML
- Text

<input checked="" type="radio"/> Get	<input type="radio"/> Get	<input type="radio"/> Add
all	film	new
films	by	film
<input type="radio"/> Update	<input type="radio"/> Delete	
film	film	

**GET FILMS**

**Films:**

Request URL: https://servletcoursework-336513.nw.r.appspot.com/getAllFilms?format=csv  
 Request Method: GET  
 Status Code: 200 OK  
 Remote Address: 142.250.200.52:443  
 Referrer Policy: strict-origin-when-cross-origin

Response Headers

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: GET
access-control-allow-origin: *
```

Response:

**Format:**

- Use REST servlet
- JSON
- XML
- Text

<input checked="" type="radio"/> Get	<input type="radio"/> Get	<input type="radio"/> Add
all	film	new
films	by	film
<input type="radio"/> Update	<input type="radio"/> Delete	
film	film	

**GET FILMS**

**Films:**

1 18006., A BRONX TALE., 1993., ROBERT DE NIRO,, ROBERT DE NIRO, CHAZZ PALMINTERI,, De Niro's directorial debut, taken from a true story.  
 2 18007., A CHILD LOST FOREVER., 1993., CLAUDIA WEILL,, BEVERLY DANGELO, WILL PATTON., Another of the Odyssey labels :  
 3 18008., A FEW GOOD MEN., 1992., ROB REINER,, TOM CRUISE, JACK NICHOLSON., After the mysterious death of a young Mar:  
 4 18009., A LEAGUE OF THEIR OWN., 1992., PENNY MARSHALL,, GENA DAVIS, TOM HANKS., Loosely based on the true story of:  
 5 18010., A LITTLE PRINCESS., 1995., ALFONSO CUARÓN,, LISELE MATTHEWS, ELEANOR BRON., Enchanting family story, with a:  
 6 18011., A LOW DOWN DIRTY SHAME., 1994., KEANAN IVORY WAYANS,, KEANAN IVORY WAYANS, JADA PINKETT., Cool and sleek cor:  
 7 18012., A NIGHTMARE ON ELM ST. 4., 1988., RENNY HARLIN,, ROBERT ENGLUND, RODNEY EASTMAN., Not one of the better out:  
 8 18013., A NIGHTMARE ON ELM ST. PART 2., 1985., JACKIE SHOLDER,, ROBERT ENGLUND, KIM MYERS., Confused sequel with the :  
 9 18014., A NIGHTMARE ON ELM STREET., 1984., WES CRAVEN,, HEATHER LANGENKAMP, JOHN SAXON., The one that started the leg:  
 10 18015., A NIGHTMARE ON ELM STREET PART 6 - FREDDYS DEAD., 1991., RACHEL TALALAY,, ROBERT ENGLUND, LISA ZANE., The mi:  
 11 18016., A.P.E.X., 1994., PHILIP J.ROTH,, RICHARD KEATS, LISA ANN RUSSELL., Okay sci-fi with good ideas but hampered by:  
 12 18017., A PERFECT WORLD., 1993., CLINT EASTWOOD,, KEVIN COSTNER, CLINT EASTWOOD., Costner is a prison escapee who tries to:  
 13 18018., A RAGE IN HARLEM., 1991., BILL DUKE,, GREGORY HINES, FOREST WHITAKER., Very entertaining period comedy drama.  
 14 18019., A RIVER RUNS THROUGH IT., 1992., ROBERT REDFORD,, BRAD PITT, EMILY LLOYD., Slightly rambling but thoroughly:  
 15 18020., A SIMPLE WISH., 1997., MICHAEL RITCHIE,, MARA WILSON, MARTIN SHORT., Childrens fantasy starring Mara Wilson.  
 16 18021., A THIN LINE BETWEEN LOVE AND HATE., 1996., MARTIN LAWRENCE,, MARTIN LAWRENCE, LYNN WHITFIELD., The director of the film.  
 17 |

1 requests 252 kB transferred | 580 kB resource

Line 1, Column 1

## Get film by title

### Request:

**Format:**

- Use REST servlet
- JSON
- XML
- Text

**Action:**

- Get all films
- Get film by title
- Add new film
- Update film
- Delete film

Title: Alien

**GET FILM(S)**

Request URL: https://servletcoursework-336513.nw.r.appspot.com/getFilmByTitle?format=csv&title=Alien  
 Request Method: GET  
 Status Code: 200 OK  
 Remote Address: 142.250.200.52:443  
 Referrer Policy: strict-origin-when-cross-origin

Response Headers

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: GET
access-control-allow-origin: *
```

1 requests | 1.5 kB transferred | 2.6 kB resources

### Response:

**Format:**

- Use REST servlet
- JSON
- XML
- Text

**Action:**

- Get all films
- Get film by title
- Add new film
- Update film
- Delete film

Title: Alien

**GET FILM(S)**

Request URL: https://servletcoursework-336513.nw.r.appspot.com/getFilmByTitle?format=csv&title=Alien  
 Request Method: GET  
 Status Code: 200 OK  
 Remote Address: 142.250.200.52:443  
 Referrer Policy: strict-origin-when-cross-origin

Response Headers

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: GET
access-control-allow-origin: *
```

1 requests | 1.5 kB transferred | 2.6 kB resources | Line 1, Column 1

Line	Film Details
1	10044., ALIEN., 1979., RIDLEY SCOTT., SIGOURNEY WEAVER, IAN HOLM., The original and many believe still the best, though it's been surpassed by its sequels. 2 10045., ALIEN 3., 1992., DAVID FINCHER., SIGOURNEY WEAVER, CHARLES DANCE., Third outing for Sigourney Weaver as Ripley, this one set on the space station Nostromo. 3 10046., ALIEN NATION., 1988., GRAHAM BAKER., JAMES CAAN, MANDY PATINKIN., A sci-fi thriller with political undertones. 4 10047., ALIEN SPECIES., 1995., DAVE PAYNE., MARIA FORD, EMILE LEVISETTI., Alien underground as a bunch of genetic scumbags. 5 10048., ALIENS: SPECIAL EDITION., 1986., JAMES CAMERON., SIGOURNEY WEAVER, MICHAEL BIEHN., The Special Edition of the movie. 6 10689., MY STEPMOTHER IS AN ALIEN., 1988., RICHARD BENJAMIN., KIM BASINGER, DAN AYKROYD., The title rather gives it away.

## Create new film

### Request:

SIMPLY SERVER

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

Film attributes:  Title

Network tab (Network tab selected)

Name: insertFilm?format=csv

General

- Request URL: https://servletcoursework-336513.nw.r.appspot.com/insertFilm?format=csv
- Request Method: POST
- Status Code: 200
- Remote Address: 142.250.200.20:443
- Referrer Policy: strict-origin-when-cross-origin

1 requests | 21 B transferred | 1 B resources

### Request body:

SIMPLY SERVER

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

Film attributes:  Title

Network tab (Network tab selected)

Name: insertFilm?format=csv

Query String Parameters

- format: csv

Request Payload

```
"Stalker", "1979", "Andrei Tarkovsky", "Alexander Kaidan"
```

1 requests | 21 B transferred | 1 B resources

### Response:

SIMPLY SERVER

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

Film attributes:  Title

Network tab (Network tab selected)

Name: insertFilm?format=csv

Response

1	1
---	---

1 requests | 21 B transferred | 1 B resources

Line 1, Column 1

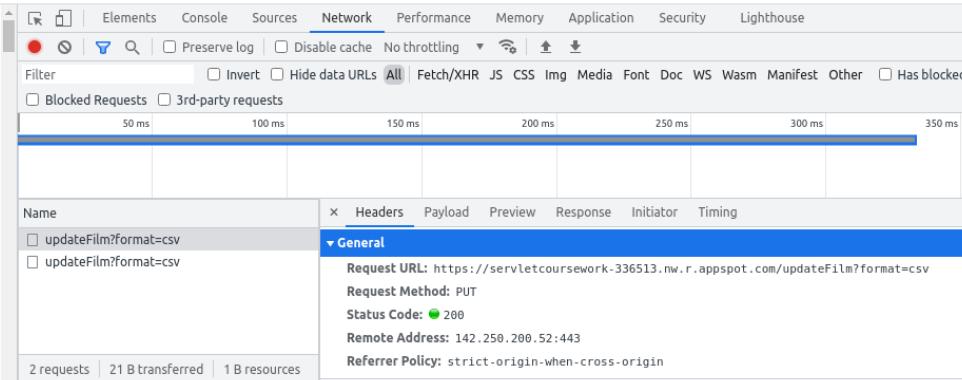
## Update film

### Request:

**Format:**

- Use REST servlet
- JSON
- XML
- Text

- Get all films
- Get film by title
- Add new film
  
- Update film
- Delete film



Request URL: https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=csv  
 Request Method: PUT  
 Status Code: 200  
 Remote Address: 142.250.200.52:443  
 Referrer Policy: strict-origin-when-cross-origin

2 requests | 21 B transferred | 1 B resources

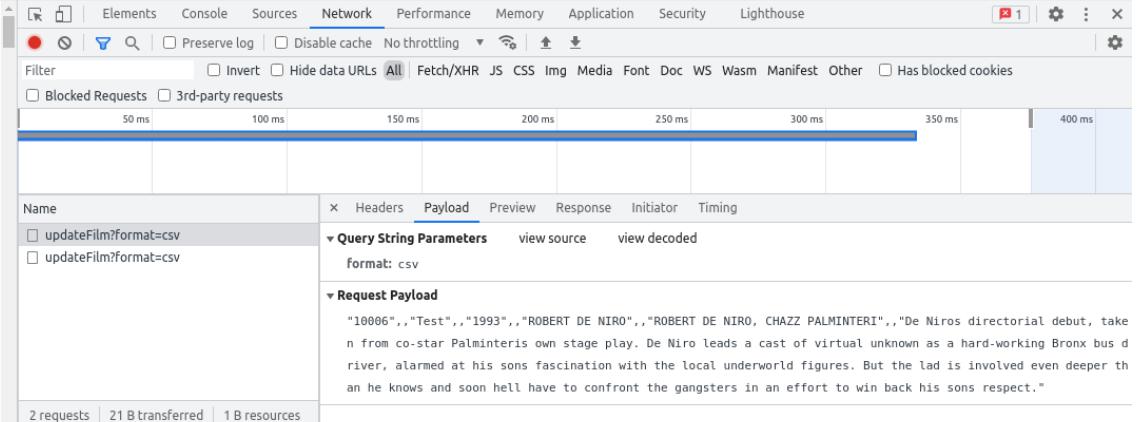
### Request body:

**Format:**

- Use REST servlet
- JSON
- XML
- Text

- Get all films
- Get film by title
- Add new film
  
- Update film
- Delete film

Selected film ID: 10006



Request URL: https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=csv  
 Request Method: PUT  
 Status Code: 200  
 Remote Address: 142.250.200.52:443  
 Referrer Policy: strict-origin-when-cross-origin

2 requests | 21 B transferred | 1 B resources

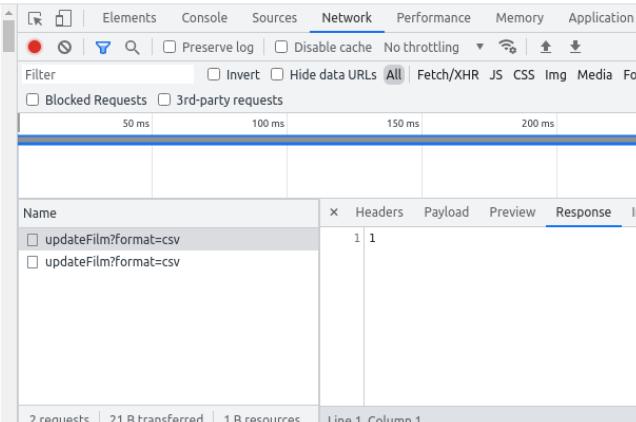
### Response:

**Format:**

- Use REST servlet
- JSON
- XML
- Text

- Get all films
- Get film by title
- Add new film
  
- Update film
- Delete film

Selected film ID: 10006



Request URL: https://servletcoursework-336513.nw.r.appspot.com/updateFilm?format=csv  
 Request Method: PUT  
 Status Code: 200  
 Remote Address: 142.250.200.52:443  
 Referrer Policy: strict-origin-when-cross-origin

2 requests | 21 B transferred | 1 B resources

## Delete film

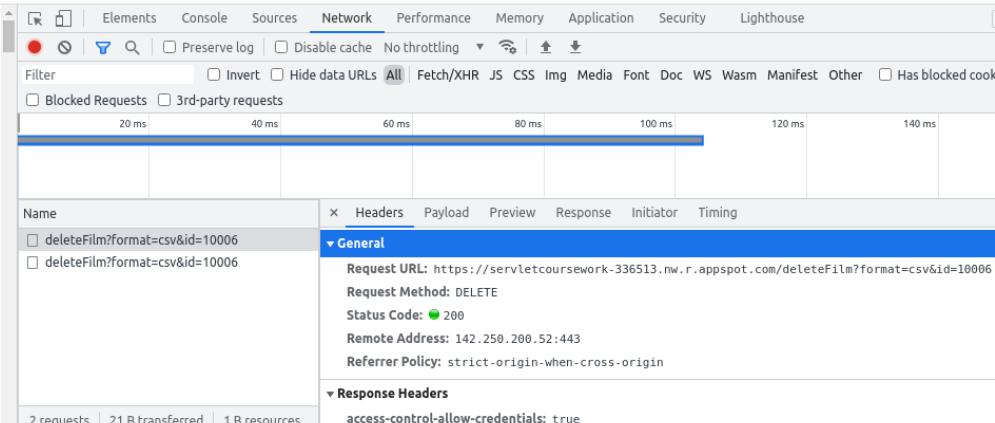
### Request:

Format:

Use REST servlet  
 JSON  XML  Text

Get all films  Get film by title  Add new film  
 Update film  Delete film

Film ID: 10006



The screenshot shows the Network tab in the Chrome DevTools. A single request is listed under 'General':  
Request URL: https://servletcoursework-336513.nw.r.appspot.com/deleteFilm?format=csv&id=10006  
Request Method: DELETE  
Status Code: 200  
Remote Address: 142.250.200.52:443  
Referrer Policy: strict-origin-when-cross-origin  
Response Headers:  
access-control-allow-credentials: true

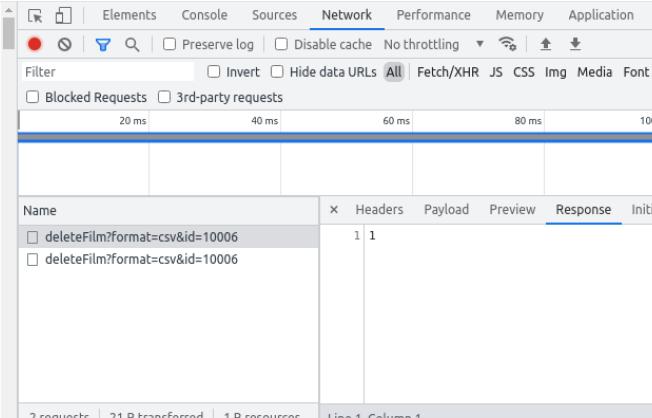
### Response:

Format:

Use REST servlet  
 JSON  XML  Text

Get all films  Get film by title  Add new film  
 Update film  Delete film

Film ID: 10006



The screenshot shows the Network tab in the Chrome DevTools. A single response is listed under 'Response':  
1 | 1  
Line 1, Column 1

# REST

## JSON

### Get all films

#### Request:

Format:  Use REST servlet  JSON  XML  Text

Get all films by title  Get film new film  Add film  Update film  Delete film

**GET FILMS**

Network tab in developer tools showing a request to `https://servetcoursework-336513.nw.r.appspot.com/REST?format=json&getType=all`. Request Method: GET. Status Code: 200. Response Headers include `access-control-allow-credentials: true`.

#### Response:

Network tab in developer tools showing a response from the same URL. Preview tab shows the JSON data:

```

[{"id": 10001, "title": "1492 - CONQUEST OF PARADISE", "year": 1992, "director": "RIDLEY SCOTT",...},...
[0 ... 99]
  ▼ 0: {"id": 10001, "title": "1492 - CONQUEST OF PARADISE", "year": 1992, "director": "RIDLEY SCOTT",...}
    director: "RIDLEY SCOTT"
    id: 10001
    review: "The story of Columbus discovering America, beautifully shot in Costa Rica and excellently played by an top-notch c..."
    stars: "GERARD DEPARDIEU, ARMAND ASSANTE"
    title: "1492 - CONQUEST OF PARADISE"
    year: 1992
  ▷ 1: {"id": 10002, "title": "187", "year": 1997, "director": "KEVIN REYNOLDS",...}
  ▷ 2: {"id": 10003, "title": "9 1/2 WEEKS", "year": 1986, "director": "ADRIAN LYNE",...}
  ▷ 3: {"id": 10004, "title": "976-EVIL II: THE RETURN", "year": 1988, "director": "JIM WYNORSKI",...}
  ▷ 4: {"id": 10005, "title": "A BETTER TOMORROW", "year": 1986, "director": "JOHN WOO",...}
  ▷ 5: {"id": 10006, "title": "A BRONX TALE", "year": 1993, "director": "ROBERT DE NIRO",...}
  ▷ 6: {"id": 10007, "title": "A CHILD LOST FOREVER", "year": 1993, "director": "CLAUDIA WEILL",...}
  ▷ 7: {"id": 10008, "title": "A FEW GOOD MEN", "year": 1992, "director": "ROB REINER",...}
  ▷ 8: {"id": 10009, "title": "A LEAGUE OF THEIR OWN", "year": 1992, "director": "PENNY MARSHALL",...}

```

GET FILM(S)

Films:

Review ID: 10001  
Title: CONQUEST OF PARADISE Story of - SCORPIUS AND US CONQUEST OF PARADISE

## Get film by title

### Request:

The screenshot shows the Network tab of the Chrome DevTools. A single request is listed: `REST?format=json&title=Evil&getype=title`. The response tab is selected, displaying the following JSON data:

```

[{"id": 10004, "title": "976-EVIL II: THE RETURN", "year": 1988, "director": "JIM WYNORSKI",...},
 ▾ 0: {"id": 10004, "title": "976-EVIL II: THE RETURN", "year": 1988, "director": "JIM WYNORSKI",...},
   director: "JIM WYNORSKI"
   id: 10004
   review: "Sequel that still finds the ultimate telephone line, the Devils equivalent of an 0898 number, acquiring victims. The stars: \"PATRICK O'BRYAN, DEBBIE JAMES\""
   title: "976-EVIL II: THE RETURN"
   year: 1988
▶ 1: {"id": 10228, "title": "CONFessions - TWO FACES OF EVIL", "year": 1994, "director": "GILBERT CATES",...}
▶ 2: {"id": 10292, "title": "DEVIL IN A BLUE DRESS", "year": 1995, "director": "CARL FRANKLIN",...}
▶ 3: {"id": 10325, "title": "DUST DEVIL", "year": 1992, "director": "RICHARD STANLEY",...}
▶ 4: {"id": 10340, "title": "EVIL DEAD II - DEAD BY DAWN", "year": 1987, "director": "SAM RAIMI",...}
▶ 5: {"id": 10460, "title": "HEAR NO EVIL", "year": 1992, "director": "ROBERT GREENWALD",...}
▶ 6: {"id": 10661, "title": "MOBSTERS: THE EVIL EMPIRE", "year": 1991, "director": "MICHAEL KARBELNIKOFF",...}
▶ 7: {"id": 10807, "title": "PROPHET OF EVIL", "year": 1993, "director": "JUD TAYLOR",...}
▶ 8: {"id": 10881, "title": "SEE NO EVIL, HEAR NO EVIL", "year": 1989, "director": "ARTHUR HILLER",...}
▶ 9: {"id": 11042, "title": "THE DEVILS OWN", "year": 1997, "director": "ALAN J. PAKULA",...}]

```

### Response:

The screenshot shows the Network tab of the Chrome DevTools. A single request is listed: `REST?format=json&title=Evil&getype=title`. The response tab is selected, displaying the following JSON data:

```

[{"id": 10004, "title": "976-EVIL II: THE RETURN", "year": 1988, "director": "JIM WYNORSKI",...},
 ▾ 0: {"id": 10004, "title": "976-EVIL II: THE RETURN", "year": 1988, "director": "JIM WYNORSKI",...},
   director: "JIM WYNORSKI"
   id: 10004
   review: "Sequel that still finds the ultimate telephone line, the Devils equivalent of an 0898 number, acquiring victims. The stars: \"PATRICK O'BRYAN, DEBBIE JAMES\""
   title: "976-EVIL II: THE RETURN"
   year: 1988
▶ 1: {"id": 10228, "title": "CONFessions - TWO FACES OF EVIL", "year": 1994, "director": "GILBERT CATES",...}
▶ 2: {"id": 10292, "title": "DEVIL IN A BLUE DRESS", "year": 1995, "director": "CARL FRANKLIN",...}
▶ 3: {"id": 10325, "title": "DUST DEVIL", "year": 1992, "director": "RICHARD STANLEY",...}
▶ 4: {"id": 10340, "title": "EVIL DEAD II - DEAD BY DAWN", "year": 1987, "director": "SAM RAIMI",...}
▶ 5: {"id": 10460, "title": "HEAR NO EVIL", "year": 1992, "director": "ROBERT GREENWALD",...}
▶ 6: {"id": 10661, "title": "MOBSTERS: THE EVIL EMPIRE", "year": 1991, "director": "MICHAEL KARBELNIKOFF",...}
▶ 7: {"id": 10807, "title": "PROPHET OF EVIL", "year": 1993, "director": "JUD TAYLOR",...}
▶ 8: {"id": 10881, "title": "SEE NO EVIL, HEAR NO EVIL", "year": 1989, "director": "ARTHUR HILLER",...}
▶ 9: {"id": 11042, "title": "THE DEVILS OWN", "year": 1997, "director": "ALAN J. PAKULA",...}]

```

## Create new film

### Request:

Get Get Add  
all film new  
films by film  
title

Update Delete  
film film

**Film attributes:**

- Title 500 Days of Summer
- Year 2009
- Director Marc Webb
- Stars Joseph Gordon-Levitt
- Review Sad :(

**CREATE NEW FILM**

1 requests | 110 B transferred | 1 B resources

### Request body:

Get Get Add  
all film new  
films by film  
title

Update Delete  
film film

**Film attributes:**

- Title 500 Days of Summer
- Year 2009
- Director Marc Webb
- Stars Joseph Gordon-Levitt
- Review Sad :(

format: json

**Request Payload**

```
{
  "title": "500 Days of Summer",
  "year": "2009",
  "director": "Marc Webb",
  "stars": "Joseph Gordon-Levitt, Zoey Deschanel",
  "review": "Sad :( "
}
```

### Response:

Get Get Add  
all film new  
films by film  
title

Update Delete  
film film

**Film attributes:**

- Title 500 Days of Summer
- Year 2009
- Director Marc Webb

1

## Update film

### Request:

<input type="radio"/> Get all films	<input type="radio"/> Get film by title	<input type="radio"/> Add new film
<input checked="" type="radio"/> Update film	<input type="radio"/> Delete film	

Selected film ID: 10001

**GET FILM DATA**

### Films:

ID	Title	Year	Stars	Review
100092	1992 - CONQUEST OF PARADISE	1992	GERARD DEPARDIEU, ARMAND ASSANTE	The story of Columbus - SCOTTY SCOTTLY bring America, beautifully CONQUEST ARMAND AND in Costa Rica and excellently OF ASSANTE PARADISE

Network tab in developer tools showing a PUT request to https://servletcoursework-336513.nw.r.appspot.com/REST?format=json with status code 200.

Request Headers:

- access-control-allow-credentials: true
- access-control-allow-headers: Content-Type
- access-control-allow-methods: PUT
- access-control-allow-origin: \*

Request Payload:

```
id: 10001, title: "1492 - CONQUEST OF PARADISE", year: 1992, director: "RIDLEY SCOTT", stars: "GERARD DEPARDIEU, ARMAND ASSANTE", review: "Test"
```

### Request body:

<input type="radio"/> films	<input type="radio"/> by title	<input type="radio"/> film
<input checked="" type="radio"/> Update film	<input type="radio"/> Delete film	

Selected film ID: 10001

**GET FILM DATA**

### Films:

ID	Title	Year	Stars	Review
100092	1992 - CONQUEST OF PARADISE	1992	GERARD DEPARDIEU, ARMAND ASSANTE	The story of Columbus - SCOTTY SCOTTLY bring America, beautifully CONQUEST ARMAND AND in Costa Rica and excellently OF ASSANTE PARADISE

Network tab in developer tools showing a PUT request to https://servletcoursework-336513.nw.r.appspot.com/REST?format=json with status code 200.

Request Headers:

- access-control-allow-credentials: true
- access-control-allow-headers: Content-Type
- access-control-allow-methods: PUT
- access-control-allow-origin: \*

Request Payload:

```
id: 10001, title: "1492 - CONQUEST OF PARADISE", year: 1992, director: "RIDLEY SCOTT", stars: "GERARD DEPARDIEU, ARMAND ASSANTE", review: "Test"
```

### Response:

films      by title      film

Update film       Delete film

Selected film ID: 10001

**GET FILM DATA**

**Films:**

ID	Title	Year	Director	Stars	Review
10001	1992 PARADISE	1992	SCOTT DEPARTMENT	ARMANDO ASSANTE	The story of Columbus - CONQUEST OF PARADISE. A tale of Spring America, beautifully OF ASSANTE. Directed by an top-notch cast. PARADISE

Network

Preserve log   Disable cache   No throttling    Invert  Hide data URLs **All** | Fetch/XHR JS CSS Img Media Font

Blocked Requests   3rd-party requests

50 ms   100 ms   150 ms   200 ms   250 ms

Name	Headers	Payload	Preview	Response	Initiat
REST?format=json				1	1
REST?format=json					

2 requests | 21 B transferred | 1 B resources   Line 1, Column 1

## Delete film

### Request:

The screenshot shows the Network tab of the browser developer tools. A DELETE request is selected. The General section shows the URL: `https://servletcoursework-336513.nw.r.appspot.com/REST?format=json&id=10001`, Method: `DELETE`, Status Code: `200`, and Headers including `access-control-allow-credentials: true` and `access-control-allow-headers: Content-Type`. The Response tab shows a single line of JSON data: `1 | 1`.

Film ID: 10001

**DELETE FILM**

**Films:**

### Response:

The screenshot shows the Network tab of the browser developer tools. A DELETE request is selected. The General section shows the URL: `https://servletcoursework-336513.nw.r.appspot.com/REST?format=json&id=10001`, Method: `DELETE`, Status Code: `200`, and Headers including `access-control-allow-credentials: true` and `access-control-allow-headers: Content-Type`. The Response tab shows a single line of JSON data: `1 | 1`.

Film ID: 10001

**DELETE FILM**

**Films:**

# XML

## Get all films

### Request:

#### Format:

Use REST servlet

JSON  XML  Text

- Get all films
- Get film by title
- Add new film
  
- Update film
- Delete film

**GET FILMS**

Request URL: https://servletcoursework-336513.nw.r.appspot.com/REST?format=xml&getType=all

Request Method: GET

Status Code: 200

Remote Address: 142.250.200.52:443

Referrer Policy: strict-origin-when-cross-origin

Response Headers

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: GET
```

1 requests | 270 kB transferred | 743 kB resource

### Response:

#### Format:

Use REST servlet

JSON  XML  Text

- Get all films
- Get film by title
- Add new film
  
- Update film
- Delete film

**GET FILMS**

### Films:

ID Title Year Stars Review

10002	1987 EVIL SAMUEL L JACKSON	The title number is the police code for a homicide, and its
REYNOLDS	YEAR	REYNOLDS

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <root>
3   <film>
4     <id>10002</id>
5     <title>187</title>
6     <year>1997</year>
7     <director>KEVIN REYNOLDS</director>
8     <stars>SAMUEL L. JACKSON, JOHN HEARD</stars>
9     <review>The title number is the police code for a homicide, and its heard more than once in
10    </film>
11    <film>
12      <id>10003</id>
13      <title>9 1/2 WEEKS</title>
14      <year>1986</year>
15      <director>ADRIAN LYNE</director>
16      <stars>MICKEY ROURKE, KIM BASINGER</stars>
17      <review>Director Lyne shows his pop video roots in this erotic tale reputedly based on a tru
18    </film>
19    <film>
20      <id>10004</id>
21      <title>976-EVIL II: THE RETURN</title>
22      <year>1988</year>
23      <director>JIM WYNORSKI</director>
24      <stars>PATRICK O'BRYAN, DEBBIE JAMES</stars>
25

```

1 requests | 270 kB transferred | 743 kB resource

## Get film by title

### Request:

Use REST servlet

JSON     XML     Text

Get all films     Get film by title     Add new film

Update film     Delete film

Title: Harry

**GET FILM(S)**

**Films:**

ID	Title	Year	Stars	Review
11283ENI980BBILLY	BILLY	1980	★★★½	Brilliantly well-observed adult comedy following the characters of the title, two friends...

Request URL: https://servletcoursework-336513.nw.r.appspot.com/REST?format=xml&title=Harry&getype=title  
 Request Method: GET  
 Status Code: 200  
 Remote Address: 142.250.179.244:443  
 Referrer Policy: strict-origin-when-cross-origin

Response Headers

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: GET
access-control-allow-origin: *
alt-svc: h3=":443"; ma=2592000,h3-29=:443; ma=2592000,h3-0850=:443; ma=2592000,h3-0846=:443; ma=2592000,h3-0843=:443; ma=2592000,quic=:443; ma=2592000; v="46,43"
cache-control: no-cache
```

1 requests | 700 B transferred | 1.2 kB resource

### Response:

Use REST servlet

JSON     XML     Text

Get all films     Get film by title     Add new film

Update film     Delete film

Title: Harry

**GET FILM(S)**

**Films:**

ID	Title	Year	Stars	Review
11283ENI980BBILLY	BILLY	1980	★★★½	Brilliantly well-observed adult comedy following the characters of the title, two friends...

Request URL: https://servletcoursework-336513.nw.r.appspot.com/REST?format=xml&title=Harry&getype=title  
 Request Method: GET  
 Status Code: 200  
 Remote Address: 142.250.179.244:443  
 Referrer Policy: strict-origin-when-cross-origin

Response

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <film>
    <id>11283</id>
    <title>WHEN HARRY MET SALLY</title>
    <year>1989</year>
    <director>ROB REINER</director>
    <stars>BILLY CRYSTAL, MEG RYAN</stars>
    <review>Brilliantly well-observed adult comedy following the characters of the title, two friends...
```

1 requests | 700 B transferred | 1.2 kB resource

## Create new film

### Request:

Film attributes:

Title	Her
Year	2014
Director	Spike Jonze
Stars	Joaquin Phoenix, Scarlett Johansson
Review	Made me cry

### Request body:

Film attributes:

Title	Her
Year	2014
Director	Spike Jonze
Stars	

### Response:

Film attributes:

Title	Her
-------	-----

## Update film

### Request:

**Format:**

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Update film    Delete film

Selected film ID: 10006

### Requests body:

**Format:**

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Update film    Delete film

Selected film ID: 10006

**GET FILM DATA**

### Response:

**Format:**

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

## Delete film

### Request:

**Format:**

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Update film    Delete film

Film ID: 10006

Request URL: https://servletcoursework-336513.nw.r.appspot.com/REST?format=xml&id=10006  
Request Method: DELETE  
Status Code: 200  
Remote Address: 142.250.179.244:443  
Referrer Policy: strict-origin-when-cross-origin

Response Headers

access-control-allow-credentials: true  
access-control-allow-headers: Content-Type  
access-control-allow-methods: DELETE

### Response:

**Format:**

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Response

1 | 1

## CSV

### Get all films

#### Request:

##### Format:

Use REST servlet

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

**GET FILMS**

Network tab in Chrome DevTools showing a GET request to `https://servletcoursework-336513.nw.r.appspot.com/REST?format=csv&getType=all`. The response is a CSV file with the following content:

Name	Year	Director	Plot Summary
10002..	187..	KEVIN REYNOLDS,, SAMUEL L.JACKSON, JOHN HEARD,,	The title number is the police
10003..	9 1/2 WEEKS..	ADRIAN LYNE,, MICKEY ROURKE, KIM BASINGER,,	Director Lyne shows his
3	976-EVIL II: THE RETURN..	JIM WYNIORSKI,, PATRICK O'BRYAN, DEBBIE JAMES,,	Sequel tha
4	A BETTER TOMORROW..	JOHN WOO,, CHOW YUN-FAT, LESLIE CHEUNG,,	The film which brough
5	A CHILD LOST FOREVER..	CLAUDIA WEILL,, BEVERLY DANGELO, WILL PATTON,,	Another of t
6	A FEW GOOD MEN..	RON REIMER,, TOM CRUISE, JACK NICHOLSON,,	After the mysterious de
7	A LEAGUE OF THEIR OWN..	PENNY MARSHALL,, GEENA DAVIS, TOM HANKS,, Loosely based on	7
8	A LITTLE PRINCESS..	ALFONSO CUARON,, LISELE MATTHEWS, ELEANOR BRON,,	Enchanting fa
9	A LOW DOWN DIRTY SHAME..	KEEAN IVORY WAYANS,, KEEAN IVORY WAYANS, JADA PINKETT,,	9
10	A NIGHTMARE ON ELM ST. 4..	ROBERT ENGLUND, RODNEY EASTMAN,, Not one	10
11	A NIGHTMARE ON ELM ST. PART 2..	JACK SHOLDER,, ROBERT ENGLUND, KID MYERS,, Confuse	11
12	A NIGHTMARE ON ELM STREET..	WES CRAVEN,, HEATHER LANGENKAMP, JOHN SAXON,,	12
13	A NIGHTMARE ON ELM STREET PART 6 - FREDDYS DEAD..	RACHEL TALALAY,, ROBERT ENGLUND,	13
14	A.P.E.X... 1994..	PHILIP J.ROTH,, RICHARD KEATS, LISA ANN RUSSELL,, Okay sci-fi with good	14
15	A PERFECT WORLD..	CLINT EASTWOOD,, KEVIN COSTNER, CLINT EASTWOOD,, Costner is a pr	15
16	A RAGE IN HARLEM..	BILL DUKE,, GREGORY HINES, FOREST WHITAKER,, Very entertaining	16
17	A RIVER RUNS THROUGH IT..	ROBERT REDFORD,, BRAD PITT, EMILY LLOYD,, Slightly rambl	17
18	A SIMPLE WISH..	MICHAEL RITCHIE,, MARA WILSON, MARTIN SHORT,, Childrens fantasy st	18
19	A THIN LINE BETWEEN LOVE AND HATE..	MARTIN LAWRENCE, MARTIN LAWRENCE, LYNN WHITFI	19
20	A TIME TO KILL..	JOEL SCHUMACHER,, MATTHEW MC CONAUGHEY,, Based on another John Gr	20
21	A VIEW TO A KILL..	ROGER MOORE, CHRISTOPHER WALKEN,, Not the best movi	21
22	A.W.O.L. - ABSENT WITHOUT LEAVE..	SHELDON LETTICH,, JEAN CLAUDE VAN DAMME,, A Fore	22
23	A WALK IN THE CLOUDS..	ALFONSO ARAU,, KEANU REEVES, GIANCARLO GIANNINI,, Moody lo	23
24	A WOMAN SCORCHED..	ANDREW STEVENS,, ANDREW STEVENS, SHANNON TWEED,, After her husba	24

#### Response:

##### Format:

Use REST servlet

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

**GET FILMS**

#### Films:

ID Title Year Director Review

10002 1987 SAMUEL The title number is the police

Network tab in Chrome DevTools showing a GET request to `https://servletcoursework-336513.nw.r.appspot.com/REST?format=csv&getType=all`. The response is a CSV file with the following content:

ID	Title	Year	Director	Review
10002	1987 SAMUEL	1987	SAMUEL	The title number is the police

## Get film by title

### Request:

**Format:**

Use REST servlet

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

Title: Star

**GET FILM(S)**

Request URL: https://servletcoursework-336513.nw.r.appspot.com/REST?format=csv&title=Star&getType=title

Request Method: GET

Status Code: 200

Remote Address: 172.217.169.20:443

Referrer Policy: strict-origin-when-cross-origin

Response Headers

- access-control-allow-credentials: true
- access-control-allow-headers: Content-Type
- access-control-allow-methods: GET
- access-control-allow-origin: \*

1 requests | 2.2 kB transferred | 4.2 kB resource

### Response:

**Format:**

Use REST servlet

JSON  XML  Text

Get all films  Get film by title  Add new film

Update film  Delete film

Title: Star

**GET FILM(S)**

1, 10377, FIST OF THE NORTH STAR, 1996, TONY RANDALL, GARY DANIELS, COSTAS MANDYLOR,, Live-action^  
2, 10742, ONE TOUGH BASTARD, 1995,, KURT WIMMER, BRIAN BOSWORTH, BRUCE PAYNE, Somewhat indelicat  
3, 10949,, STAR TREK: FIRST CONTACT, 1996, JONATHAN FRAKES,, PATRICK STEWART, JONATHAN FRAKES,, On  
4, 10950,, STAR TREK: GENERATIONS, 1994, DAVID CARSON,, PATRICK STEWART, WILLIAM SHATNER,, An astr  
5, 10951,, STAR TREK III: THE SEARCH FOR SPOCK, 1984, LEONARD NIMOY,, WILLIAM SHATNER, DEFOREST KE  
6, 10952,, STAR TREK IV: THE VOYAGE HOME, 1986, LEONARD NIMOY,, WILLIAM SHATNER, LEONARD NIMOY,, C  
7, 10953,, STAR TREK VI : THE UNDISCOVERED COUNTRY,, 1991, NICHOLAS MEYER,, WILLIAM SHATNER, LEONAR  
8, 10954,, STAR WARS,, 1978,, GEORGE LUCAS,, MARK HAMILL, HARRISON FORD,, The original, the one that  
9, 10955,, STARGATE,, 1994, ROLAND EMMERICH,, KURT RUSSELL, JAMES SPADER,, Spader is the scientist

1 requests | 2.2 kB transferred | 4.2 kB resource

Line 1, Column 1

## Create new film

### Request:

**Format:**

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Update film    Delete film

**Film attributes:**

Title	The Shining
Year	

Request URL: https://servletcoursework-336513.nw.r.appspot.com/REST?format=csv  
 Request Method: POST  
 Status Code: 200  
 Remote Address: 172.217.169.20:443  
 Referrer Policy: strict-origin-when-cross-origin

Response Headers

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: POST
access-control-allow-origin: *
alt-svc: h3=":443"; ma=2592000,h3-29=:443; ma=2592000,h3-0050=:443; ma=2592000,
```

1 requests | 21 B transferred | 1 B resources

### Request body:

**Format:**

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Update film    Delete film

**Film attributes:**

Title	The Shining
Year	

1 | 1

1 requests | 21 B transferred | 1 B resources | Line 1, Column 1

### Response:

**Format:**

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Update film    Delete film

1 | 1

## Update film

## Request:

## Request body:

The screenshot shows a browser developer tools interface with the Network tab selected. A request labeled "REST?format=csv" is highlighted. The response payload contains the following JSON data:

```
[{"id": 1, "title": "A BETTER TOMORROW", "year": 1986, "director": "Woo John", "actor": "CHOW YUN-FAT, LESLIE CHEUNG", "plot": "The film which brought international stardom to star Chow Yun-Fat and director John Woo. A roller-coaster of a ride with rothers on both sides of the law becoming united in their quest for personal revenge. After forger Mar K (Chow) is crippled and double crossed by an associate, cop Kit (Cheung) risks his badge to help him. Violent and gripping. Also available as a boxed set with a supplementary documentary and booklet."}]
```

## Response:

The screenshot shows a browser's developer tools Network tab. At the top, there are tabs for Elements, Console, Sources, Network (which is selected), Performance, Memory, and Application. Below the tabs, there are filters for Preserve log, Disable cache, and No throttling. The main area displays a table of network requests. The first row shows a request to 'REST?format=csv' with a status of '1'. The table has columns for Name, Headers, Payload, Preview, Response (which is selected), and Initiat. On the left side of the screen, there is a sidebar with several options: 'Use REST servlet' (radio button selected), 'JSON', 'XML', 'Text', 'Get all films' (radio button selected), 'Get film by title', 'Add new film', 'Update film' (radio button selected), and 'Delete film'. At the bottom left, the text 'Selected film ID: 10005' is displayed.

## Delete film

### Request:

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Update film    Delete film

Film ID: 10005

**DELETE FILM**

**Films:**

Network tab details:

- Request URL: <https://servetcoursework-336513.nw.r.appspot.com/REST?format=csv&id=10005>
- Request Method: DELETE
- Status Code: 200
- Remote Address: 142.250.179.244:443
- Referrer Policy: strict-origin-when-cross-origin

Response Headers:

```
access-control-allow-credentials: true
access-control-allow-headers: Content-Type
access-control-allow-methods: DELETE
access-control-allow-origin: *
alt-svc: h3=":443"; ma=2592000, h3-29=:443; ma=2592000, h3-0850=":443"; ma=2592000, h3-0046="
```

2 requests | 21 B transferred | 1 B resources

### Response:

Use REST servlet

JSON    XML    Text

Get all films    Get film by title    Add new film

Update film    Delete film

Network tab details:

- Request URL: <https://servetcoursework-336513.nw.r.appspot.com/REST?format=csv&id=10005>
- Request Method: DELETE
- Status Code: 200
- Remote Address: 142.250.179.244:443
- Referrer Policy: strict-origin-when-cross-origin

Response tab details:

Name	Value
REST?format=csv&id=10005	1
REST?format=csv&id=10005	1