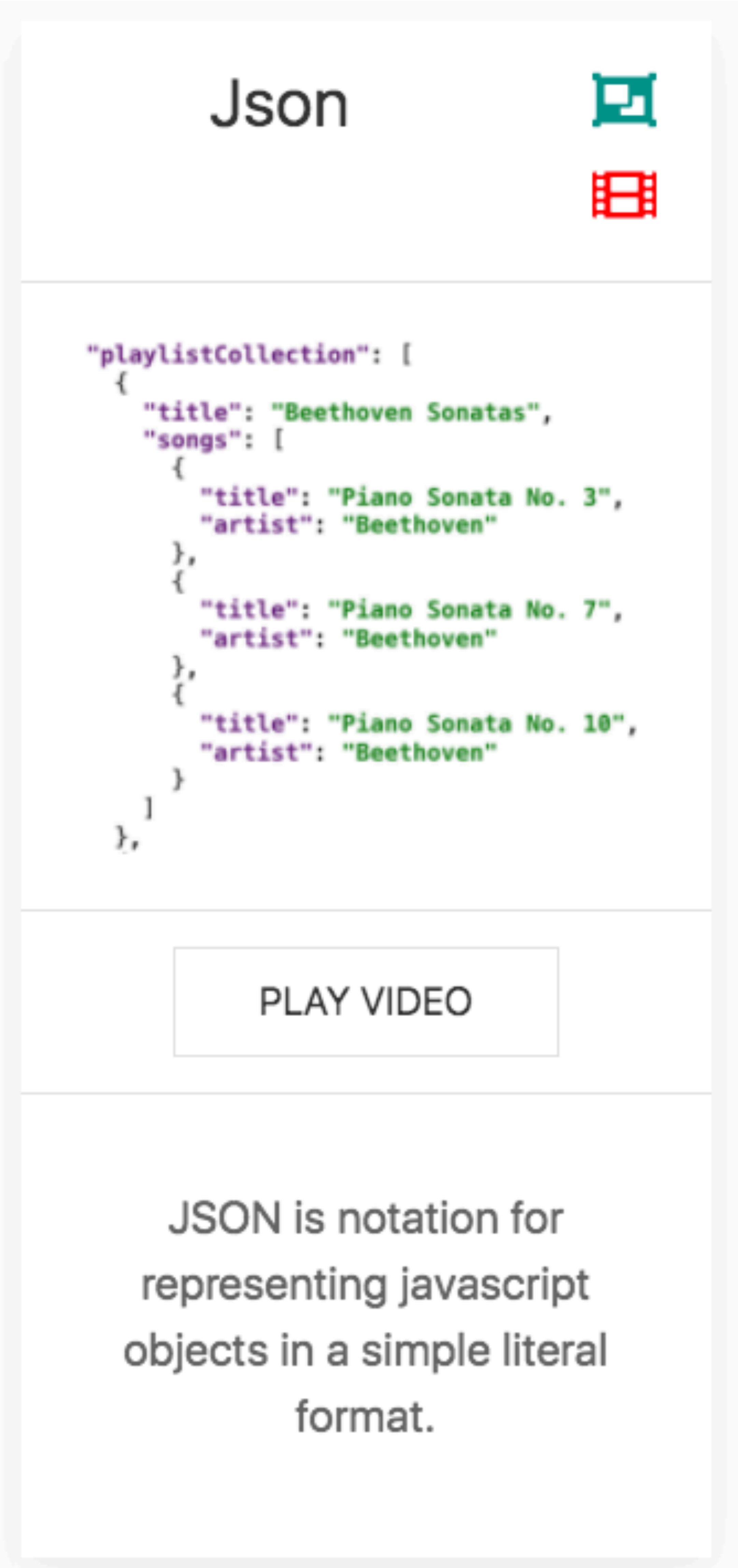
JSON



```
"playlistCollection": [
  {
    "title": "Beethoven Sonatas",
    "songs": [
      {
        "title": "Piano Sonata No. 3",
        "artist": "Beethoven"
      },
      {
        "title": "Piano Sonata No. 7",
        "artist": "Beethoven"
      },
      {
        "title": "Piano Sonata No. 10",
        "artist": "Beethoven"
      }
    ]
  },
```

JSON is notation for representing javascript objects in a simple literal format.

Model Data in Play

java

```java
Song s1 = new Song("Piano Sonata No. 3", "Beethoven");
Song s2 = new Song("Piano Sonata No. 7", "Beethoven");
Song s3 = new Song("Piano Sonata No. 10", "Beethoven");
Playlist p1 = new Playlist("Beethoven Sonatas");
p1.songs.add (s1);
p1.songs.add (s2);
p1.songs.add (s3);
```

Embedded in a compiled program.
When running, objects occupy appropriate in memory data structures.

yaml

```yaml
Song(s1):
  title: Piano Sonata No. 3
  artist: Beethoven
  duration: 5
Song(s2):
  title: Piano Sonata No. 7
  artist: Beethoven
  duration: 6
Song(s3):
  title: Piano Sonata No. 10
  artist: Beethoven
  duration: 8

Playlist(p1):
  title: Bethoven Sonatas
  duration: 19
  songs:
  - s1
  - s2
  - s3
```

Just a File format.
Used to represent structured information in a flat file.
Must be processed by various tools in order to be useful.

JSON Playlist Collection

- Javascript Object Notation

- This is a data file in JSON format

- It is a subset of javascript notation describing a data structure

playlist-store.json

```json
{
  "playlistCollection": [
    {
      "title": "Beethoven Sonatas",
      "songs": [
        {
          "title": "Piano Sonata No. 3",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Sonata No. 7",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Sonata No. 10",
          "artist": "Beethoven"
        }
      ]
    },
    {
      "title": "Beethoven Concertos",
      "songs": [
        {
          "title": "Piano Concerto No. 0",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Concerto No. 4",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Concerto No. 6",
          "artist": "Beethoven"
        }
      ]
    }
  ]
}
```

- A song object, containing:
  - **title** - a string
  - **artist** - a string

playlist-store.json

```json
{
  "title": "Piano Concerto No. 6",
  "artist": "Beethoven"
}
```

playlist-store.json

- Three song objects, each containing
  - **title** - a string
  - **artist** - a string

```
{
  "title": "Piano Concerto No. 0",
  "artist": "Beethoven"
},
{
  "title": "Piano Concerto No. 4",
  "artist": "Beethoven"
},
{
  "title": "Piano Concerto No. 6",
  "artist": "Beethoven"
}
```

playlist-store.json

- **songs** - an array of three song objects, each containing
  - **title** - a string
  - **artist** - a string

```
"songs": [
  {
    "title": "Piano Concerto No. 0",
    "artist": "Beethoven"
  },
  {
    "title": "Piano Concerto No. 4",
    "artist": "Beethoven"
  },
  {
    "title": "Piano Concerto No. 6",
    "artist": "Beethoven"
  }
]
```

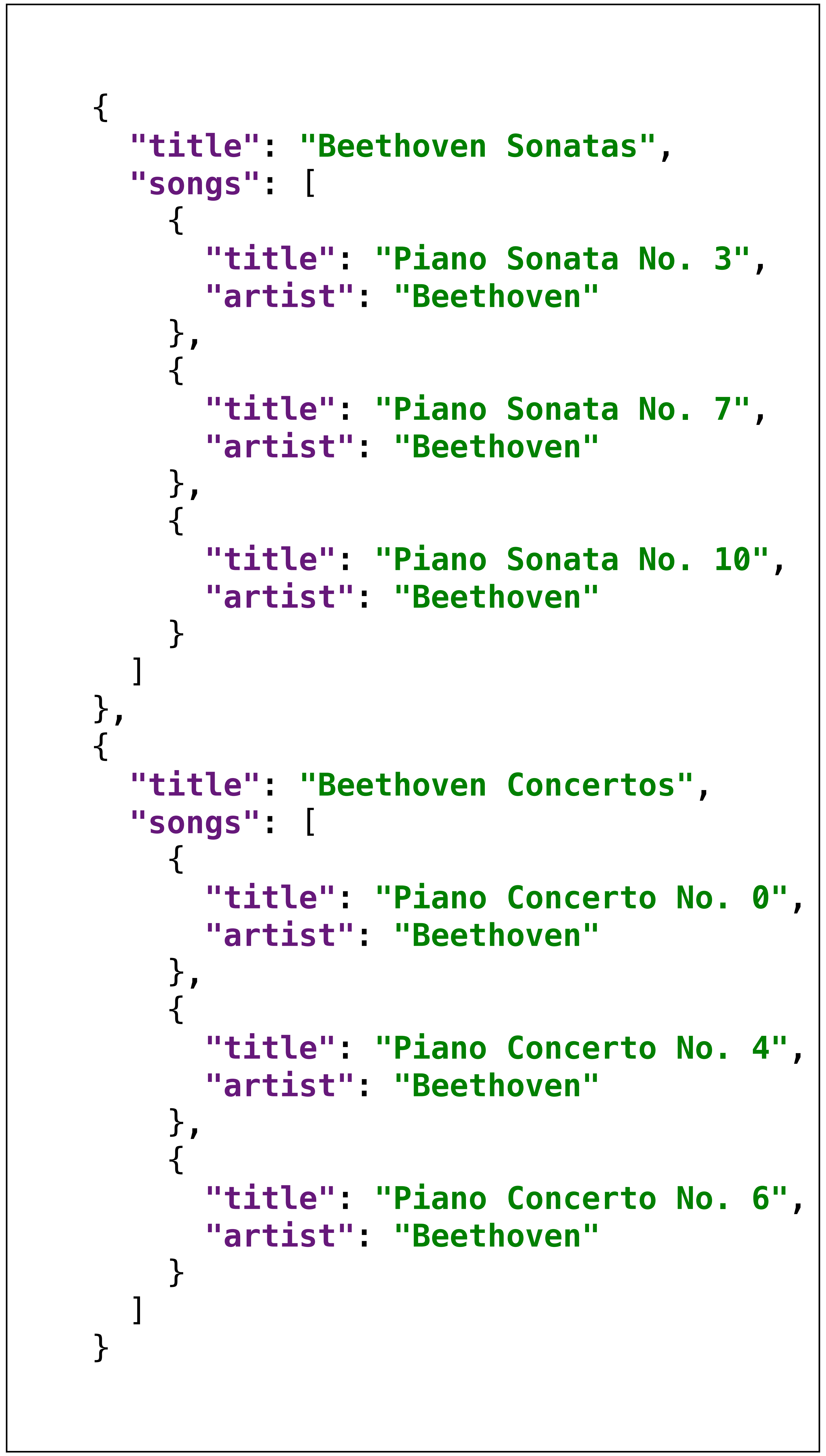
playlist-store.json

- A **playlist** object, containing:
  - **title** - a string
  - **songs** - an array of objects, each containing
    - **title** - a string
    - **artist** - a string

```json
{
  "title": "Beethoven Concertos",
  "songs": [
    {
      "title": "Piano Concerto No. 0",
      "artist": "Beethoven"
    },
    {
      "title": "Piano Concerto No. 4",
      "artist": "Beethoven"
    },
    {
      "title": "Piano Concerto No. 6",
      "artist": "Beethoven"
    }
  ]
}
```

playlist-store.json

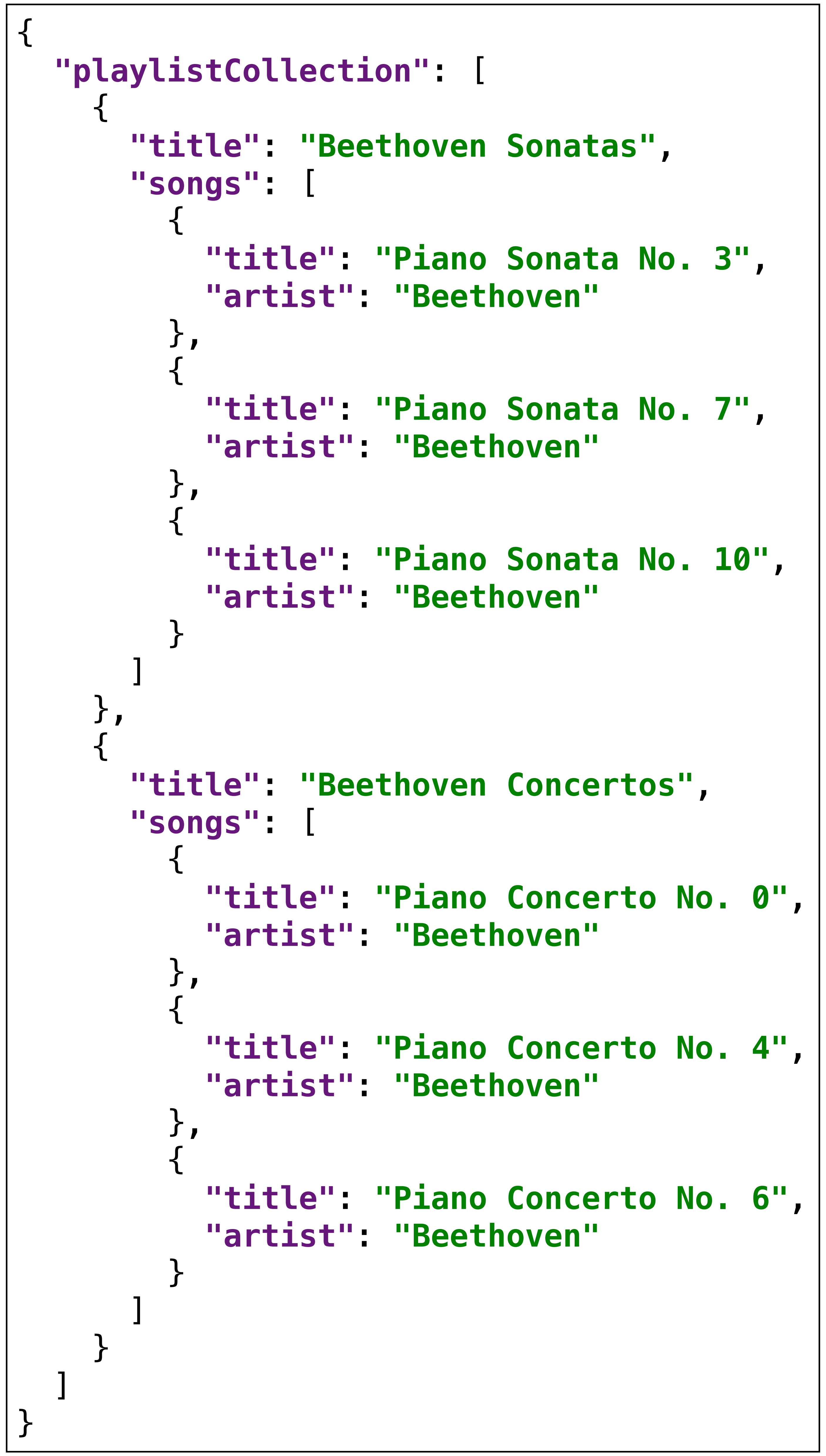
- 2 **playlist** objects, each containing:
  - **title** - a string
  - **songs** - an array of objects, each containing:
    - **title** - a string
    - **artist** - a string



```
{
  "title": "Beethoven Sonatas",
  "songs": [
    {
      "title": "Piano Sonata No. 3",
      "artist": "Beethoven"
    },
    {
      "title": "Piano Sonata No. 7",
      "artist": "Beethoven"
    },
    {
      "title": "Piano Sonata No. 10",
      "artist": "Beethoven"
    }
  ]
},
{
  "title": "Beethoven Concertos",
  "songs": [
    {
      "title": "Piano Concerto No. 0",
      "artist": "Beethoven"
    },
    {
      "title": "Piano Concerto No. 4",
      "artist": "Beethoven"
    },
    {
      "title": "Piano Concerto No. 6",
      "artist": "Beethoven"
    }
  ]
}
```

playlist-store.json

- A **playlistCollection** object, containing:
  - an array of 2 **playlist** objects, each containing:
    - **title** - a string
    - **songs** - an array of objects, each containing:
      - **title** - a string
      - **artist** - a string



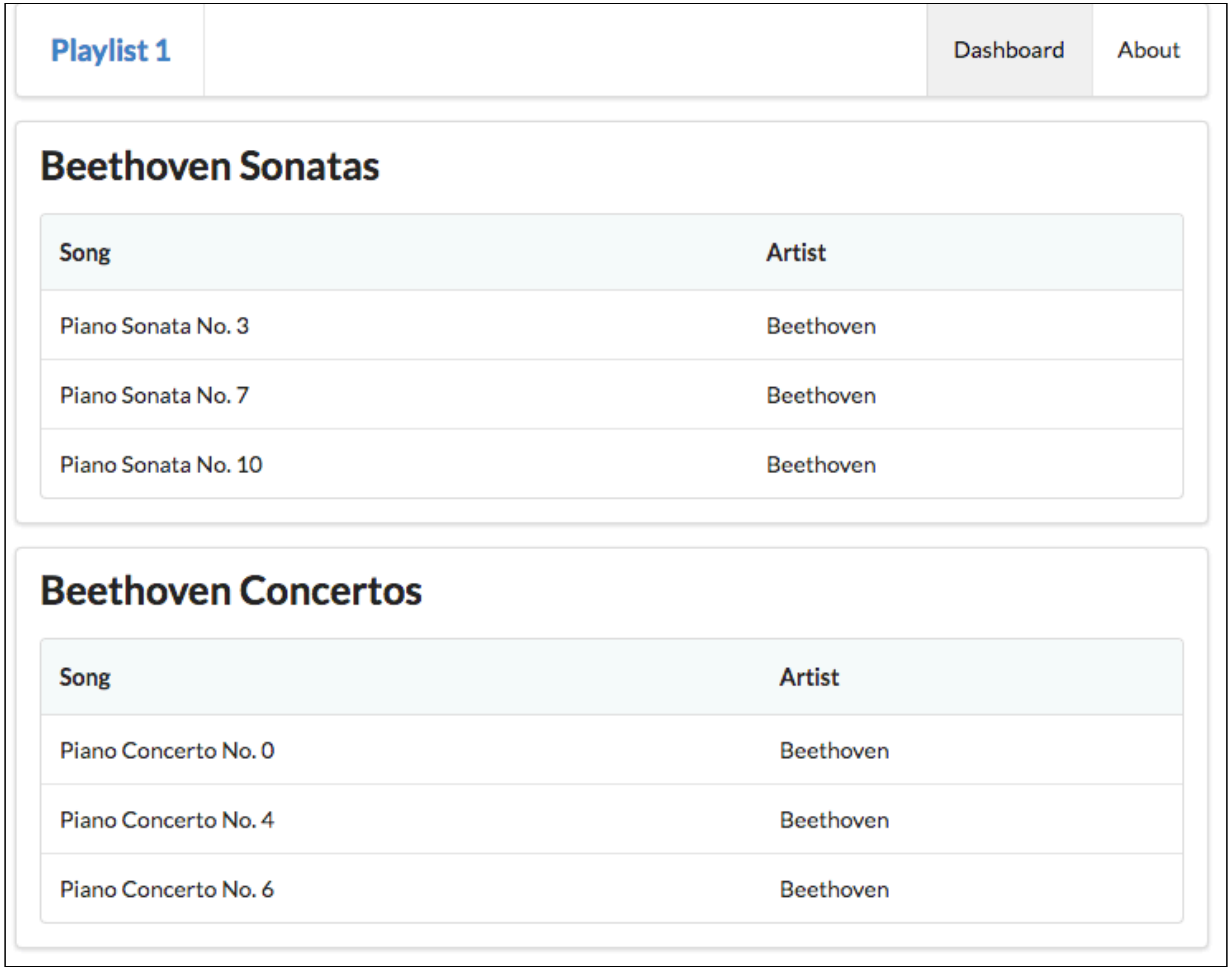
```json
{
  "playlistCollection": [
    {
      "title": "Beethoven Sonatas",
      "songs": [
        {
          "title": "Piano Sonata No. 3",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Sonata No. 7",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Sonata No. 10",
          "artist": "Beethoven"
        }
      ]
    },
    {
      "title": "Beethoven Concertos",
      "songs": [
        {
          "title": "Piano Concerto No. 0",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Concerto No. 4",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Concerto No. 6",
          "artist": "Beethoven"
        }
      ]
    }
  ]
}
```

yaml

```
Song(s1):
  title: Piano Sonata No. 3
  artist: Beethoven
Song(s2):
  title: Piano Sonata No. 7
  artist: Beethoven
Song(s3):
  title: Piano Sonata No. 10
  artist: Beethoven
Song(s4):
  title: Piano Concerto No. 0
  artist: Beethoven
Song(s5):
  title: Piano Concerto No. 4
  artist: Beethoven
Song(s6):
  title: Piano Concerto No. 6
  artist: Beethoven

Playlist(p1):
  title: Bethoven Sonatas
  songs:
  - s1
  - s2
  - s3
Playlist(p2):
  title: Bethoven Concertos
  songs:
  - s1
  - s2
  - s3

PlayCollection(l1):
  playlists:
  - p1
  - p2
```

json

```
{
  "playlistCollection": [
    {
      "title": "Beethoven Sonatas",
      "songs": [
        {
          "title": "Piano Sonata No. 3",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Sonata No. 7",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Sonata No. 10",
          "artist": "Beethoven"
        }
      ]
    },
    {
      "title": "Beethoven Concertos",
      "songs": [
        {
          "title": "Piano Concerto No. 0",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Concerto No. 4",
          "artist": "Beethoven"
        },
        {
          "title": "Piano Concerto No. 6",
          "artist": "Beethoven"
        }
      ]
    }
  ]
}
```



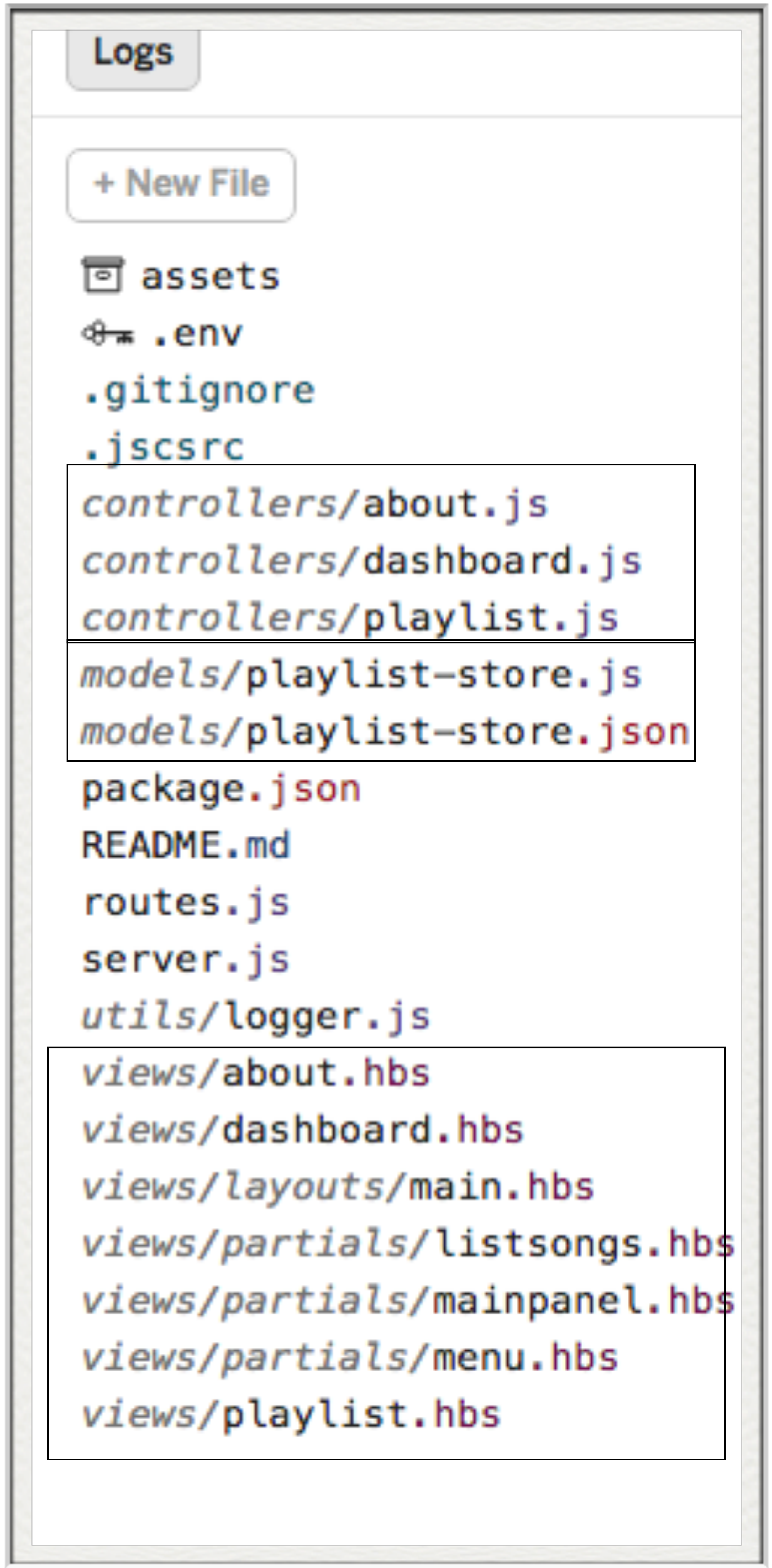
Beethoven Sonatas

Song	Artist
Piano Sonata No. 3	Beethoven
Piano Sonata No. 7	Beethoven
Piano Sonata No. 10	Beethoven

Beethoven Concertos

Song	Artist
Piano Concerto No. 0	Beethoven
Piano Concerto No. 4	Beethoven
Piano Concerto No. 6	Beethoven

playlist-2



```
assets
.env
.gitignore
.jscsrc
controllers/about.js
controllers/dashboard.js
controllers/playlist.js
models/playlist-store.js
models/playlist-store.json
package.json
README.md
routes.js
server.js
utils/logger.js
views/about.hbs
views/dashboard.hbs
views/layouts/main.hbs
views/partials/listsongs.hbs
views/partials/mainpanel.hbs
views/partials/menu.hbs
views/playlist.hbs
```

- Model-View-Controller Pattern

- Controllers:

- Model Playlists stored in JSON

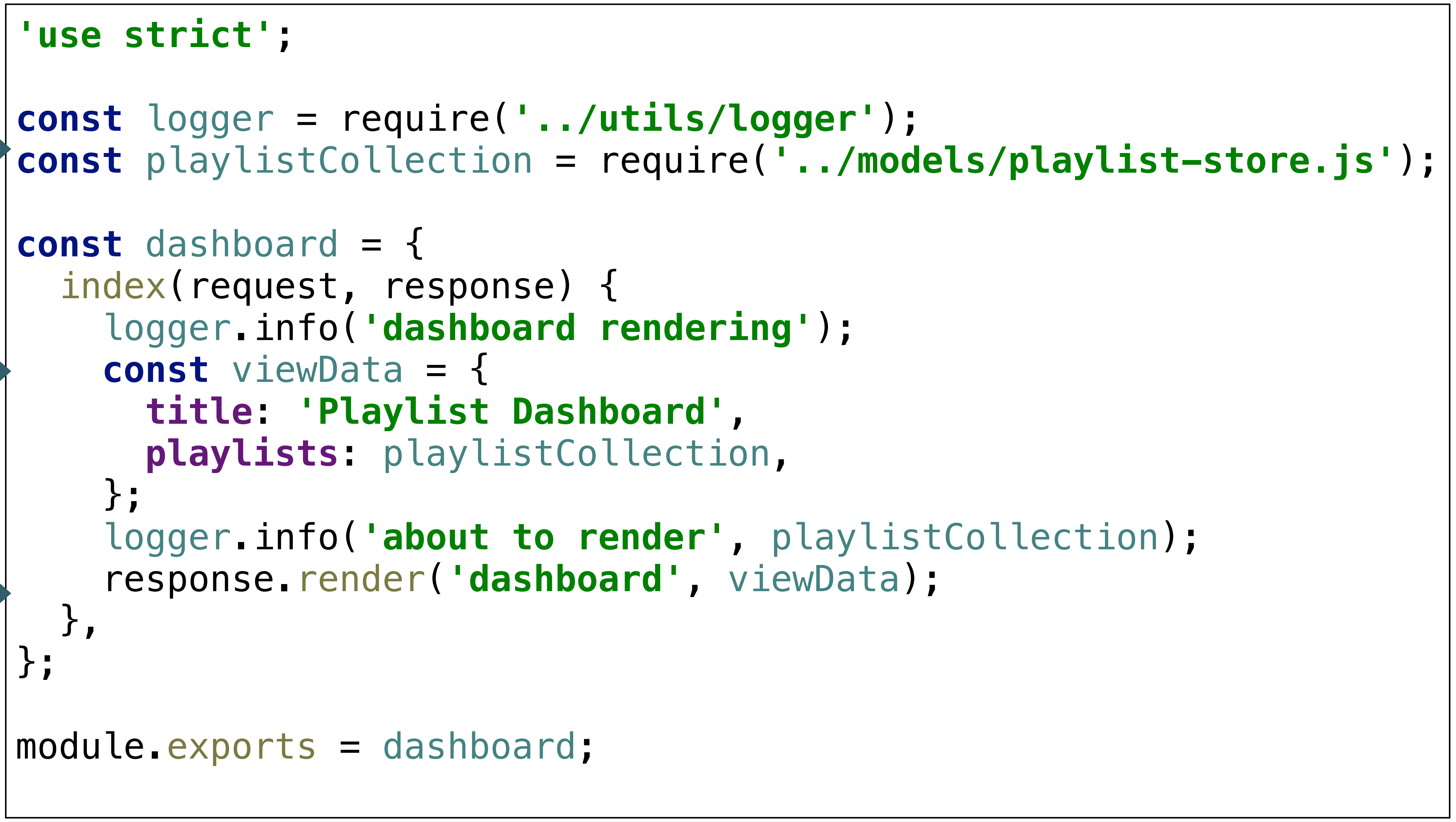
- Views:

Controller / Model

Load the model (the playlists)

Place model in viewData object

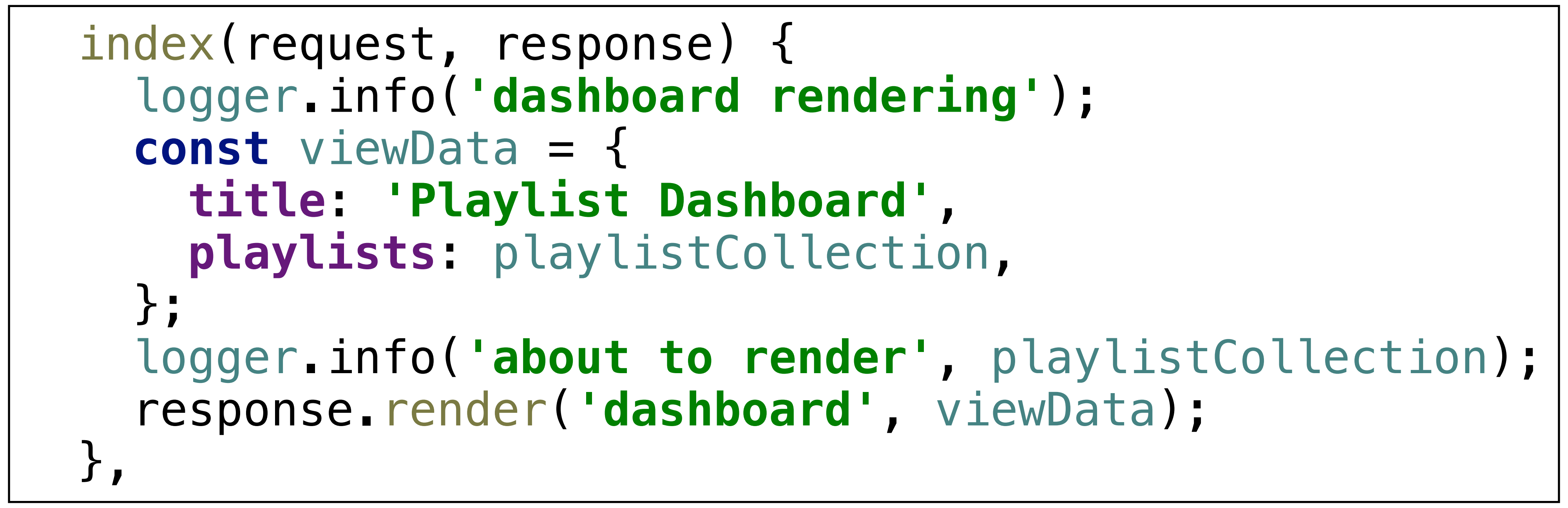
Send viewData to view



```
'use strict';

const logger = require('../utils/logger');
const playlistCollection = require('../models/playlist-store.js');

const dashboard = {
  index(request, response) {
    logger.info('dashboard rendering');
    const viewData = {
      title: 'Playlist Dashboard',
      playlists: playlistCollection,
    };
    logger.info('about to render', playlistCollection);
    response.render('dashboard', viewData);
  },
};

module.exports = dashboard;
```

Controller/View

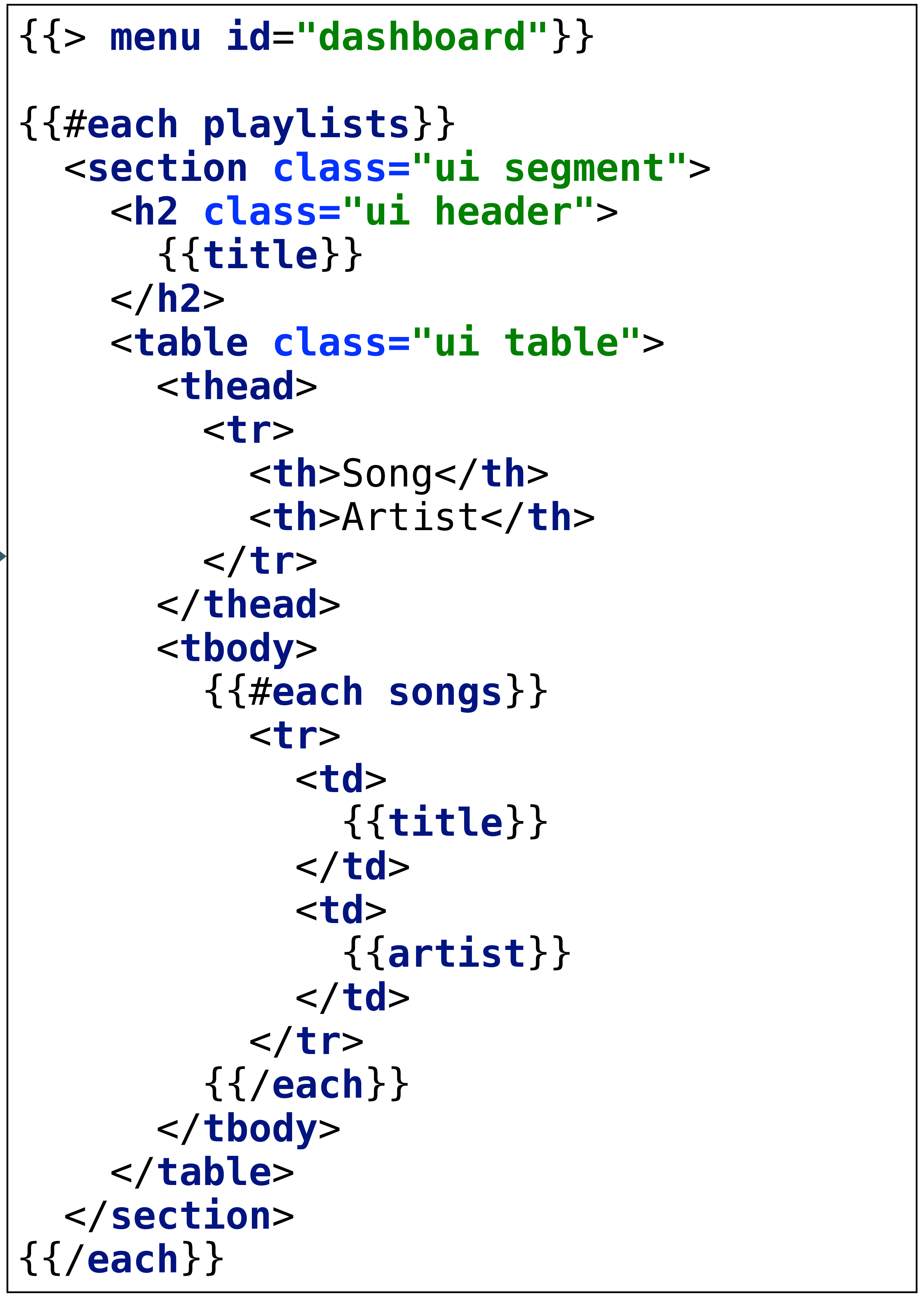
dashboard.js



```
  index(request, response) {
    logger.info('dashboard rendering');
    const viewData = {
      title: 'Playlist Dashboard',
      playlists: playlistCollection,
    };
    logger.info('about to render', playlistCollection);
    response.render('dashboard', viewData);
  },
```

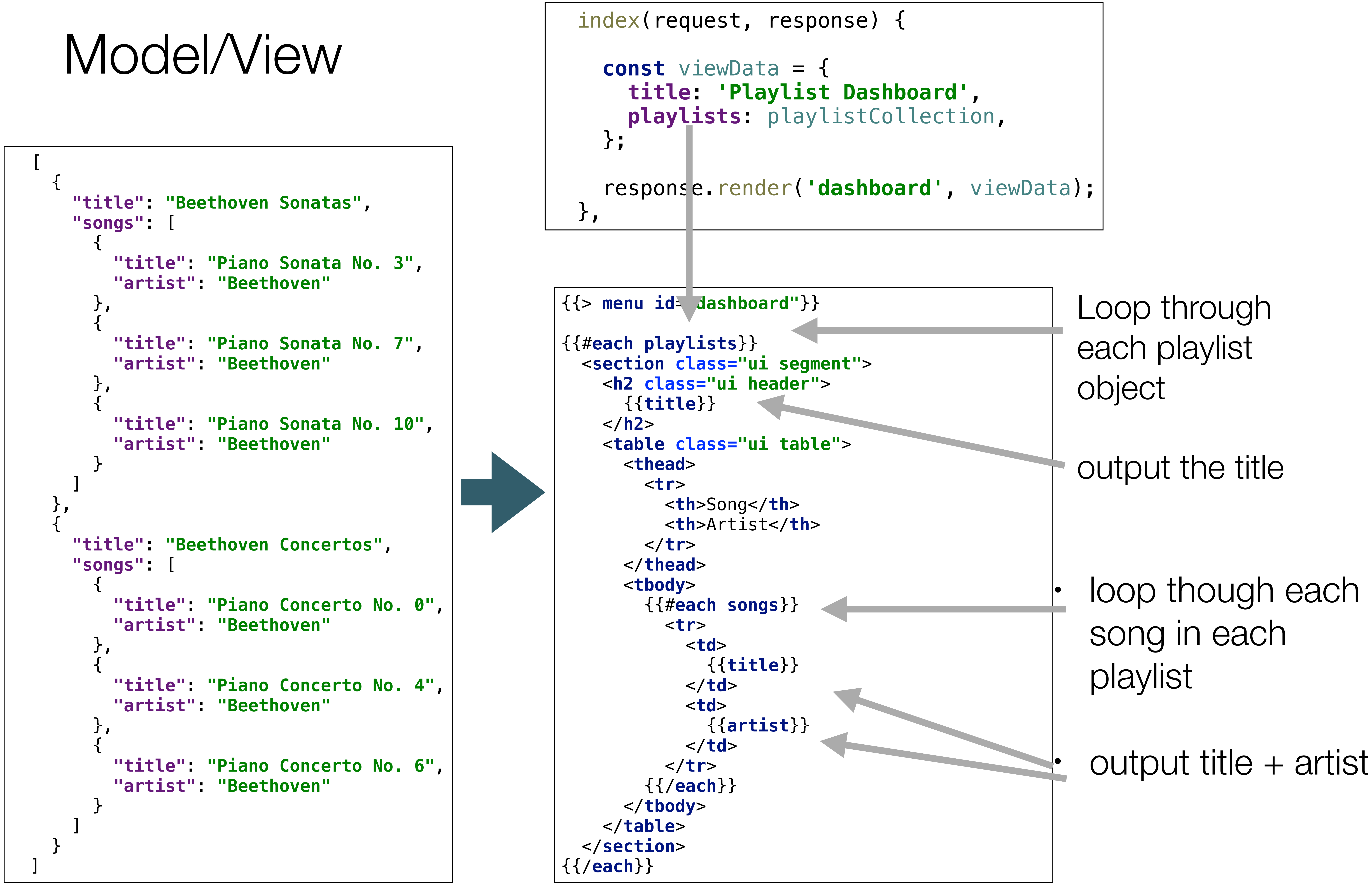
- viewData send to the dashboard view

dashboard.hbs



```
{{> menu id="dashboard"}}

{{#each playlists}}
  <section class="ui segment">
    <h2 class="ui header">
      {{title}}
    </h2>
    <table class="ui table">
      <thead>
        <tr>
          <th>Song</th>
          <th>Artist</th>
        </tr>
      </thead>
      <tbody>
        {{#each songs}}
          <tr>
            <td>
              {{title}}
            </td>
            <td>
              {{artist}}
            </td>
          </tr>
        {{/each}}
      </tbody>
    </table>
  </section>
{{/each}}
```

Model/View



```
index(request, response) {

  const viewData = {
    title: 'Playlist Dashboard',
    playlists: playlistCollection,
  };

  response.render('dashboard', viewData);
},
```

```
[
  {
    "title": "Beethoven Sonatas",
    "songs": [
      {
        "title": "Piano Sonata No. 3",
        "artist": "Beethoven"
      },
      {
        "title": "Piano Sonata No. 7",
        "artist": "Beethoven"
      },
      {
        "title": "Piano Sonata No. 10",
        "artist": "Beethoven"
      }
    ]
  },
  {
    "title": "Beethoven Concertos",
    "songs": [
      {
        "title": "Piano Concerto No. 0",
        "artist": "Beethoven"
      },
      {
        "title": "Piano Concerto No. 4",
        "artist": "Beethoven"
      },
      {
        "title": "Piano Concerto No. 6",
        "artist": "Beethoven"
      }
    ]
  }
]
```

```
{{> menu id="dashboard"}}

{{#each playlists}}
  <section class="ui segment">
    <h2 class="ui header">
      {{title}}
    </h2>
    <table class="ui table">
      <thead>
        <tr>
          <th>Song</th>
          <th>Artist</th>
        </tr>
      </thead>
      <tbody>
        {{#each songs}}
          <tr>
            <td>
              {{title}}
            </td>
            <td>
              {{artist}}
            </td>
          </tr>
        {{/each}}
      </tbody>
    </table>
  </section>
{{/each}}
```

Loop through each playlist object

output the title

- loop though each song in each playlist
- output title + artist