

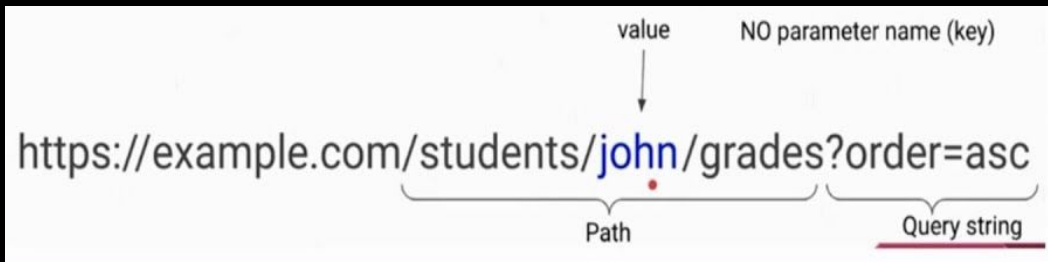
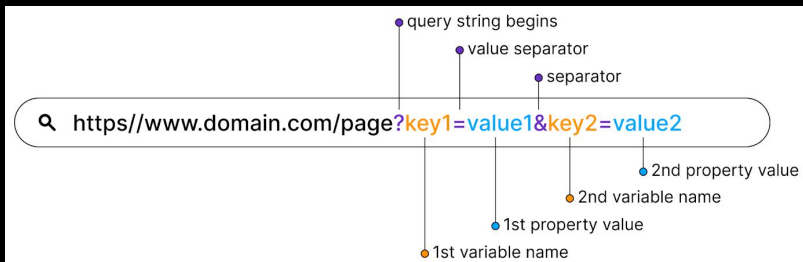
14. Dynamic Path & Model deep-dive

1. What are dynamic paths
2. Adding Home Details Page
3. Showing Real Home Data
4. Adding Favourites Feature
5. Adding Favourites Model
6. Edit-Home: Adding Feature Skeleton
7. Edit-Home: Showing Existing Data
8. Edit-Home: Handling Edit Request
9. Adding Delete Feature
10. Removing Home from Favourites





14.1 What are dynamic paths



- **Path parameters** are variables embedded directly in the URL path to capture dynamic values, like `/users/:userId` where `:userId` is replaced with a specific user ID.
- **Query parameters** are key-value pairs appended to the URL after a `?`, used to send additional data, like `/search?query=nodejs` where `query=nodejs` specifies the search term.



14.2 Adding Home Details Page

1. Add a details button in /homes-list to go to link path /homes/:home-id
2. Add a random id to each home in the data file.
3. Add a random id on home object before saving in the home model.
4. Add a route in the store router for /homes/:home-id
5. Add a method in store controller to get the home-id using req.params and log it, before sending out a dummy response with home id.



14.2 Adding Home Details Page

1.

```
home-list.ejs x homes.json home.js
views > store > home-list.ejs > ?
1 <%- include('../partials/head') %>
3 <body>
4 <%- include('../partials/nav') %>
5 <main class="container mx-auto bg-white shadow-lg rounded-lg p-8 mt-10 max-w-6xl">
9 <div class="flex flex-wrap justify-center gap-6">
25 <a href="/homes/<%= home.id %>" class="bg-blue-300 p-2">Details</a>
26 <a href="#" class="bg-green-300 p-2">Book</a>
27 </div>
```

2.

```
home-list.ejs homes.json x home.js
data > homes.json > ...
1 [{"id":"1", "houseName":"Utsav","price":"1299","location":"Ghaziabad","rating":"4",
  "photoUrl":"asdf"}, {"id":"2", "houseName":"Second House","price":"999",
  "location":"Ghaziabad","rating":"4","photoUrl":"asdf"}]
```

3.

```
home-list.ejs homes.json home.js x
models > home.js > ...
6 module.exports = class Home {
14
15   save() {
16     this.id = Math.random().toString();
17     Home.fetchAll((registeredHomes) => {
```



Utsav

Ghaziabad

Rs1299 / night

★ 4

Details

Book



14.2 Adding Home Details Page

4.

```
storeRouter.js × storeController.js
routes > storeRouter.js > ...
8 storeRouter.get("/", homesController.getIndex);
9 storeRouter.get("/homes", homesController.getHomes);
10 // homes/1
11 storeRouter.get("/homes/:homeId", homesController.getHome);
12 storeRouter.get("/bookings", homesController.getBookings);
```

← → ↻ ⓘ localhost:3000/homes/1

Details of home 1

5.

```
storeRouter.js storeController.js ×
controllers > storeController.js > ...
23 exports.getHome = (req, res, next) => {
24   const homeId = req.params.homeId;
25   console.log(homeId);
26   res.send("Details of home " + homeId);
27 };
28
```



14.3 Showing Real Home Data

1. Add a static `findByld` method in `Home` model that takes a `callback`.
2. Use this `findByld` method in the `controller` to load home details and log them.
3. Now change the `controller`:
 - a. Redirect to `/homes` in case the home is not found, logging an error.
 - b. Rendering the `/home-detail` page with home data.
4. Create a `home-detail` page to use the entire page to show all home data.
5. Download from Github:
 - a. The styled `home-detail` file.
 - b. 10 image of `homes` that you can put locally and use.



14.3 Showing Real Home Data

1.

```
storeController.js  home.js ×
models > home.js > ...
  6  module.exports = class Home {
    32
    33    static findById(id, callback) {
    34      this.fetchAll((homes) => {
    35        const home = homes.find((home) => home.id === id);
    36        callback(home);
    37      });
    38    }
    39  };
```

```
Server running on address http://localhost:3000
{
  id: '1',
  houseName: 'Utsav',
  price: '1299',
  location: 'Ghaziabad',
  rating: '4',
  photoUrl: 'asdf'
}
```

2.

```
exports.getHome = (req, res, next) => {
  const homeId = req.params.homeId;
  Home.findById(homeId, (home) => {
    console.log(home);
  });
};
```



14.3 Showing Real Home Data

```
3. exports.getHome = (req, res, next) => {
  const homeId = req.params.homeId;
  Home.findById(homeId, (home) => {
    if (!home) {
      return res.redirect('/homes'); // Redirect if home not found
    }
    res.render("store/home-detail", {
      home: home,
      pageTitle: home.houseName,
      currentPage: "homes",
    });
  });
};
```




14.3 Showing Real Home Data

4.

```
home-detail.ejs x storeController.js home-list.ejs
views > store > home-detail.ejs > ?
1 <%= include('../partials/head') %>
2 </head>
3 <body>
4 <%= include('../partials/nav') %>
5 <main class="container mx-auto bg-white shadow-lg rounded-lg p-8 mt-10 max-w-6xl">
6 <h2 class="text-3xl text-red-500 font-bold text-center mb-6">
7   Details of <%= home.houseName %>
8 </h2>
9
10 <div class="grid grid-cols-1 md:grid-cols-2 gap-8">
11 <div class="rounded-lg overflow-hidden">
12 " class="w-full h-96 object-cover">
13 </div>
14
15 <div class="space-y-6">
16 <div class="border-b pb-4">
17 <h3 class="text-2xl font-semibold mb-2">Location</h3>
18 <p class="text-gray-600"><%= home.location %></p>
19 </div>
20
21 <div class="border-b pb-4">
22 <h3 class="text-2xl font-semibold mb-2">Price</h3>
23 <p class="text-green-600 text-xl font-bold"><%= home.price %> / night</p>
24 </div>
25
26 <div class="border-b pb-4">
27 <h3 class="text-2xl font-semibold mb-2">Rating</h3>
28 <div class="flex items-center">
29 <span class="text-yellow-400 text-xl"><%= home.rating %></span>
30 <span class="ml-2 text-lg"><%= home.rating %> / 5</span>
31 </div>
32 </div>
33
34 <div class="mt-8">
35 <form action="/favourite" method="post">
36 <button type="submit" class="bg-red-500 text-white px-8 py-3 rounded-lg hover:bg-
37   Add to Favorites
38 </button>
39 </form>
40 </div>
41 </div>
42 </div>
43 </main>
44 </body>
45 </html>
```

5.

images

- house1.png
- house2.png
- house3.png
- house4.png
- house5.png
- house6.png
- house7.png
- house8.png



14.4 Adding Favourites Feature

1. Create a partial with a form and a button that submits to /favourites path with a hidden input having home-id value.
2. Add this partial to home-detail page
3. Add this partial to home-list page.
4. Add router for handling POST request to /favourites path.
5. Add a method in store controller to add a home id as favourites, logging the id for now, before redirecting to /favourites path.



14.4 Adding Favourites Feature

1.

```
home-detail.ejs storeController.js storeRouter.js home-list.ejs favourite.ejs X
views > partials > favourite.ejs > form
1 <form action="/favourites" method="post">
2   <button type="submit" class="bg-red-500 text-white px-8 py-3 rounded-lg
3     Add to Favorites
4   </button>
5   <input type="hidden" name="homeId" value="<%= home.id %>">
6 </form>
```

2.

```
home-detail.ejs X storeController.js storeRouter.js home-list.ejs
views > store > home-detail.ejs > body > main.container.mx-auto.bg-white.shadow-lg.rounded-lg.p-8.mt-10.max-w-6xl > div.grid.grid-cols-1.md:gr
1 <%= include('../partials/head') %>
3 <body>
4   <%= include('../partials/nav') %>
5   <main class="container mx-auto bg-white shadow-lg rounded-lg p-8 mt-10 max-w-6xl">
10     <div class="grid grid-cols-1 md:grid-cols-2 gap-8">
34       <div class="mt-8">
35         <%= include('../partials/favourite') %>
36       </div>
37     </div>
38   </main>
39 </body>
40 </html>
```



14.4 Adding Favourites Feature

3.

```
storeController.js storeRouter.js home-list.ejs X
views > store > home-list.ejs > ?
1 <%- include('../partials/head') %>
3 <body>
4 <%- include('../partials/nav') %>
5 <main class="container mx-auto bg-white shadow-lg rounded-lg p-8 mt-10 max-w-6xl">
9 <div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-8">
32 | | | | | <div class="flex gap-3">
33 | | | | | | <a href="/homes/<%= home.id %>" class="flex-1 bg-white text-red-500 py
| | | | | | | hover:text-white border border-red-500">View Details</a>
34 | | | | | | <%- include('../partials/favourite', { home: home }) %>
35 | | | | | </div>
36 | | | | </div>
37 | | | </div>
38 | | <%= }) %>
39 | </div>
40 </main>
41 </body>
42 </html>
```



14.4 Adding Favourites Feature

4. `storeRouter.post("/favourites", homesController.postFavourites);`
`storeRouter.get("/favourites", homesController.getFavouriteList);`

5.

storeController.js ×

controllers > storeController.js > ...

53

```
54 exports.postFavourites = (req, res, next) => {  
55   const homeId = req.body.homeId;  
56   console.log(homeId);  
57   res.redirect('/favourites');  
58 };
```



14.5 Adding Favourites Model

1. **Create** a new Model to handle Favourites:
 - a. A **static** method **getFavourites** that reads the **favourites.json** file and return the ids of all homes marked favourite.
 - b. A **static** method **addFavourites** that adds the **home-id** to **favourites-id** array if not already there, then updates the file.
2. **Use** this **model** in **addFavourites** controller.
3. **Use** this **model** in **getFavourites** controller while also **fetching** details of all homes from **Homes Model**.
4. **Change** the **UI of favourites page** to show new content.



14.5 Adding Favourites Model

1.

```
const fs = require('fs');
const path = require('path');

const rootDir = require('../utils/pathUtil');
const favouritesPath = path.join(rootDir, "data", "favourites.json");

module.exports = class Favourites {
  static addFavourite(homeId) {
    this.getFavourites((favourites) => {
      if (favourites.includes(homeId)) {
        console.log("Home already in favourites");
      } else {
        favourites.push(homeId);
        fs.writeFile(favouritesPath, JSON.stringify(favourites), (err) => {
          console.log("File Writing Concluded", err);
        });
      }
    });
  }

  static getFavourites(callback) {
    fs.readFile(favouritesPath, (err, data) => {
      callback(!err ? JSON.parse(data) : []);
    });
  }
};
```



14.5 Adding Favourites Model

2.

JS favourites.js JS storeController.js × {} favourites.json

controllers > JS storeController.js > ...

54

```
55 exports.postFavourites = (req, res, next) => {  
56   const homeId = req.body.homeId;  
57   Favourites.addFavourite(homeId);  
58   res.redirect('/favourites');  
59 };
```

JS favourites.js JS storeController.js {} favourites.json ×

data > {} favourites.json > ...

1

```
["1", "2"]
```




14.5 Adding Favourites Model

3.

```
exports.getFavouriteList = (req, res, next) => {
  Favourites.getFavourites((favourites) => {
    Home.fetchAll((registeredHomes) => {
      const favouritesWithDetails = favourites.map(homeId => registeredHomes.find(home => home.id === homeId));
      res.render("store/favourite-list", {
        favourites: favouritesWithDetails,
        pageTitle: "My Favourites",
        currentPage: "favourites",
      });
    });
  });
};
```



14.5 Adding Favourites Model

4.

```
favourites.js  storeController.js  homes.json  favourites.json  favourite-list.ejs  home-detail.ejs  home-list.ejs
views > store > favourite-list.ejs > ?
1  <!-- include('../partials/head') %>
2  </head>
3  <body>
4  <!-- include('../partials/nav') %>
5  <main class="container mx-auto bg-white shadow-lg rounded-lg p-8 mt-10 max-w-6xl">
6    <h2 class="text-3xl text-red-500 font-bold text-center mb-6">
7      Here are your favourites
8    </h2>
9    <div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-8">
10     <!-- favourites.forEach(home => { %>
11       <div class="bg-white rounded-lg shadow-md overflow-hidden hover:shadow-xl transition duration-300">
12         <div class="aspect-square">
13           " class="w-full h-full object-cover">
14         </div>
15         <div class="p-6">
16           <h3 class="text-2xl font-semibold text-gray-800 mb-3"><%= home.houseName %></h3>
17           <p class="text-gray-600 mb-4 flex items-center">
18             <svg xmlns="http://www.w3.org/2000/svg" class="h-5 w-5 mr-2" viewBox="0 0 20 20" fill="currentColor">
19               <path fill-rule="evenodd" d="M5.05 4.05a7 7 0 119.9 9.9L10 18.91-4.95-4.95a7 7 0 010-9.9zM10 11a2 2 0 10-4 2 2 0 00 4 2"
20               clip-rule="evenodd" />
21             <%= home.location %>
22           </p>
23           <div class="flex justify-between items-center mb-4">
24             <span class="text-xl font-bold text-red-500"><%= home.price %> / night</span>
25             <div class="flex items-center bg-yellow-50 px-3 py-1 rounded-full">
26               <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 24 24" fill="currentColor" class="w-5 h-5 text-yellow-400 mr-1">
27                 <path fill-rule="evenodd" d="M10.788 3.21c.448-1.077 1.976-1.077 2.424 0l2.082 5.007 5.404 4.33c1.164.093 1.636 1.545 749 2.
3051-4.117 3.527 1.257 5.273c.271 1.136-.964 2.033-1.96 1.425l12 18.354 7.373 21.18c-.996.608-2.231-.29-1.96-1.425l11.257-5.
273-4.117-3.527c-.887-.76-.415-2.212.749-2.305l5.404-.433 2.082-5.006z" clip-rule="evenodd" />
28               </svg>
29               <span class="text-gray-700 font-medium"><%= home.rating %></span>
30             </div>
31           </div>
32           <div class="flex gap-3">
33             <a href="/homes/<%= home.id %>" class="flex-1 bg-white text-red-500 py-2 px-4 rounded-lg text-center hover:bg-red-500
34               text-white border border-red-500">View Details</a>
35           </div>
36         </div>
37       <!-- %>
38     </div>
39   </main>
40 </body>
41 </html>
```