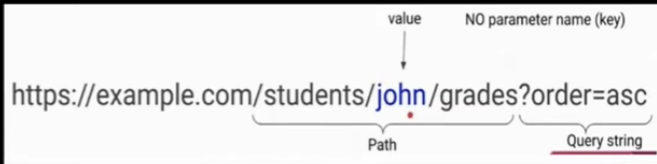
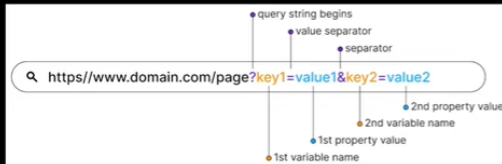


Dynamic Path::



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. `home-list.ejs`

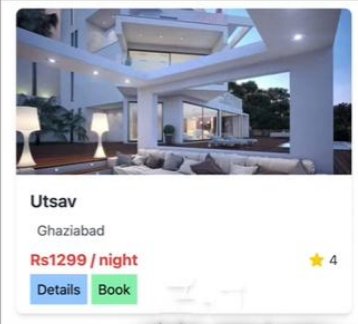
```
1 <%- include('../partials/head') %>
2
3 <body>
4   <%- include('../partials/nav') %>
5   <main class="container mx-auto bg-white shadow-lg rounded-lg p-8 mt-10 max-w-6xl">
6     <div class="flex flex-wrap justify-center gap-6">
7       <a href="/homes/<%= home.id %>" class="bg-blue-300 p-2">Details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

2. `homes.json`

```
1 [{"id": "1", "houseName": "Utsav", "price": "1299", "location": "Ghaziabad", "rating": "4",
2   "photoUrl": "asdf"}, {"id": "2", "houseName": "Second House", "price": "999",
3   "location": "Ghaziabad", "rating": "4", "photoUrl": "asdf"}]
```

3. `home.js`

```
6 module.exports = class Home {
14
15   save() {
16     this.id = Math.random().toString();
17     Home.fetchAll((registeredHomes) => {
```





14.3 Showing Real Home Data

1.

```
storeController.js  home.js x
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",
    });
  });
};

```



14.4 Adding Favourites Feature

1.

```
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.

```
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') %>
2 <body>
3   <%= include('../partials/nav') %>
4   <main class="container mx-auto bg-white shadow-lg rounded-lg p-8 mt-10 max-w-6xl">
5     <div class="grid grid-cols-1 md:grid-cols-2 gap-8">
6       <div class="mt-8">
7         <%= include('../partials/favourite') %>
8       </div>
9     </div>
10   </main>
11 </body>
12 </html>
```



14.4 Adding Favourites Feature

4.

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

5.

```
controllers > storeController.js > ...
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

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",
      });
    });
  });
};
```