



TEAM LICENSES: [Save money and learn new skills through a Hacking with Swift+ team license >>](#)

user's current search text:

```
@State private var searchText = ""
```

We then need to bind that to a `searchable()` modifier, so put this next to `navigationTitle()` in `ContentView`:

```
.searchable(text: $searchText)
```

Third, we need to update the initializer of `DestinationListView` to accept a search string, and use it for the query predicate:

```
init(sort: SortDescriptor<Destination>, searchString: String) {
    _destinations = Query(filter: #Predicate {
        if searchString.isEmpty {
            return true
        } else {
            return $0.name.localizedStandardContains(searchString)
        }
    }, sort: [sort])
}
```

Note: `localizedStandardContains()` is almost always the best way to do user-facing string searches. If you use the regular `contains()` method you'll get case-sensitive searching.