


✓ LAB3





>  .expo


✓  app


 _layout.tsx

 add-performer.tsx

 create-project.tsx


 index.tsx

 project.tsx


>  assets


>  node_modules


 .gitignore

 app.json


 babel.config.js

 expo-env.d.ts

 package-lock.json

 package.json

 README.md

 tsconfig.json

```
export default function RootLayout() {
  const [projects, setProjects] = useState([
    {
      id: "1",
      name: "Project 1",
      description: "Project details...",
      price: 100,
      hours: 10,
      sum: 1000,
      performers: [
        {
          id: "1",
          code: "P001",
          companyName: "Company A",
          phone: "123-456-7890",
        },
        {
          id: "2",
          code: "P002",
          companyName: "Company B",
          phone: "987-654-3210",
        },
      ],
    },
    {
      id: "2",
      name: "Project 2",
      description: "Project details...",
      price: 200,
      hours: 20,
      sum: 4000,
      performers: [],
    },
  ]
  );
```

```

app > index.tsx > Index > <function> > <function> > params
4   import { projectsContext } from "../_layout";
5
6   export default function Index() {
7     const { projects } = useContext(projectsContext);
8
9     return (
10      <View>
11        <View
12          style={{
13            padding: 20,
14            marginBottom: 20,
15            display: "flex",
16            flexDirection: "row",
17            justifyContent: "space-between",
18          }}
19        >
20          <Text
21            style={{
22              fontSize: 20,
23              marginBottom: 10,
24            }}
25          >
26            List of Projects
27          </Text>
28          <View style={{}}>
29            <Button
30              color="green"
31              title="New Project"
32              onPress={() => router.navigate("/create-project")}
33            />
34          </View>
35        </View>
36
37        <FlatList
38          data={projects}
39          renderItem={({ item }) => (
40            <View
41              style={{
42                padding: 10,
43                borderBottomWidth: 1,
44              }}
45            >
46              <Text
47                style={{
48                  fontSize: 20,
49                  marginBottom: 10,
50                }}
51              >
52                {item.name}
53              </Text>
54              <Button
55                title="View Project"
56                onPress={() =>
57                  router.navigate({
58                    pathname: "/project",

```



```

import { router, useLocalSearchParams } from "expo-router";
import React, { useContext } from "react"; 6.9k (gzipped: 2.7k)
import { View, Text, FlatList, Button, StyleSheet } from "react-native";
import { projectsContext } from "../_layout";

const ProjectPage = () => {
  const { projectId } = useLocalSearchParams();
  const { projects } = useContext(projectsContext);

  console.log("projectId", projectId);

  const project = projects.find((p) => p.id === projectId);
  const performers = project?.performers || [];

  const handleAddPerformer = () => {
    router.navigate({
      pathname: "/add-performer",
      params: { projectId },
    });
  };

  const renderPerformer = ({ item }: any) => (
    <View style={styles.performerContainer}>
      <Text style={styles.performerText}>Code: {item.code}</Text>
      <Text style={styles.performerText}>Company: {item.companyName}</Text>
      <Text style={styles.performerText}>Phone: {item.phone}</Text>
    </View>
  );

  return (
    <View style={styles.container}>
      <Text style={styles.projectTitle}>Project: {project.name}</Text>
      <Text style={styles.projectText}>Description: {project.description}</Text>
      <Text style={styles.projectText}>Price: ${project.price}</Text>
      <Text style={styles.projectText}>Hours: {project.hours}</Text>
      <Text style={styles.projectText}>Sum: ${project.sum}</Text>

      <Text style={styles.performersTitle}>Performers</Text>
      {performers.length === 0 && <Text>No performers yet</Text>}
      <FlatList
        data={performers}
        renderItem={renderPerformer}
        keyExtractor={(item) => item.id}
      />

      <Button title="Add Performer" onPress={handleAddPerformer} />
    </View>
  );
};

```

```
app > create-project.tsx > [x] CreateProject > [x] handleCreateProject
2 import React, { useContext, useState } from "react"; 6.9k (gzipped: 2.7k)
3 import { View, Text, TextInput, Button, StyleSheet } from "react-native";
4 import { projectsContext } from "../_layout";
5
6 const CreateProject = () => {
7   const { projects, setProjects } = useContext(projectsContext);
8
9   const [date, setDate] = useState("");
10  const [name, setName] = useState("");
11  const [description, setDescription] = useState("");
12  const [price, setPrice] = useState("");
13  const [hours, setHours] = useState("");
14  const [sum, setSum] = useState("");
15
16  const handleCreateProject = () => {
17    setProjects([
18      ...projects,
19      {
20        id: String(projects.length + 1),
21        date,
22        name,
23        description,
24        price: Number(price),
25        hours: Number(hours),
26        sum: Number(sum),
27        performers: [],
28      },
29    ]);
30    router.back();
31  };
32
33  return (
34    <View style={styles.wrapper}>
35      <Text style={styles.title}>Create New Project</Text>
36      <TextInput
37        placeholder="Date"
38        value={date}
39        onChangeText={setDate}
40        style={styles.textInput}
41      />
42      <TextInput
43        placeholder="Name"
44        value={name}
45        onChangeText={setName}
46        style={styles.textInput}
47      />
48      <TextInput
49        placeholder="Description"
50        value={description}
51        onChangeText={setDescription}
52        style={styles.textInput}
53      />
54      <TextInput
55        placeholder="Price"
```

```

app > add-performer.tsx > ...
2 import React, { useContext, useState } from "react";
3 import { View, Text, TextInput, Button, StyleSheet } from "react-native";
4 import { projectsContext } from "../_layout";
5
6 const AddPerformer = () => {
7   const { projectId } = useLocalSearchParams();
8   const { projects, setProjects } = useContext(projectsContext);
9
10  const [code, setCode] = useState("");
11  const [companyName, setCompanyName] = useState("");
12  const [phone, setPhone] = useState("");
13
14  const handleAddPerformer = () => {
15    const newProjects = projects.map((project) => {
31    });
32
33    setProjects(newProjects);
34    router.back();
35  };
}
}

University/Android/Lab3/README.md
38 <View style={styles.wrapper}>
39   <Text style={styles.title}>Add Performer</Text>
40   <TextInput
41     placeholder="Code"
42     value={code}
43     onChangeText={setCode}
44     style={styles.textInput}
45   />
46   <TextInput
47     placeholder="Company Name"
48     value={companyName}
49     onChangeText={setCompanyName}
50     style={styles.textInput}
51   />
52   <TextInput
53     placeholder="Phone"
54     value={phone}
55     onChangeText={setPhone}
56     keyboardType="phone-pad"
57     style={styles.textInput}
58   />
59   <Button title="Add Performer" onPress={handleAddPerformer} />
60 </View>
61 );
62 };
63
64 const styles = StyleSheet.create({
65   wrapper: {
66     padding: 20,
67     display: "flex",
68     flexDirection: "column",
69     justifyContent: "center",

```


index

List of Projects

NEW PROJECT

Project 1

VIEW PROJECT

Project 2

VIEW PROJECT

← create-project

Create New Project

Date

Name

Description

Price

Hours

Sum

CREATE PROJECT

 project

Project: Project 1

Description: Project details...

Price: \$100

Hours: 10

Sum: \$1000

Performers

Code: P001

Company: Company A

Phone: 123-456-7890

Code: P002

Company: Company B


Phone: 987-654-3210

Code: P221

Company: Test

Phone: 213213213

ADD PERFORMER

 add-performer

Add Performer

ADD PERFORMER