1 Practice 2.1. Create a Business card app

> In this practice you will create a Business card app using basic composables, like Column, Row, Text, Image, etc.

First of all, do the points 3 and 4 of the Build a basic layout codelab.

Recommended: do the point 5, Practice: Compose Basics.

Then, do the practice, point 6.

1.1 Restrictions

We want to do our app modular, so you must divide the business card in several composables that, put together, conform the final card.

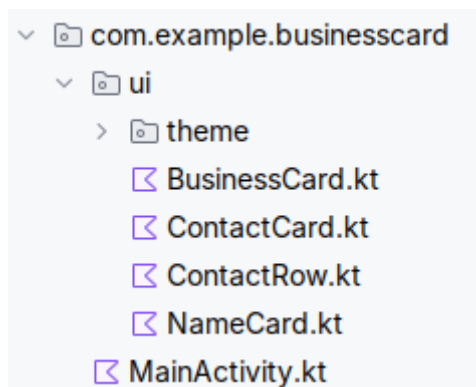For that, make different files in your **ui** folder for each composable:

Figure 1: ui files

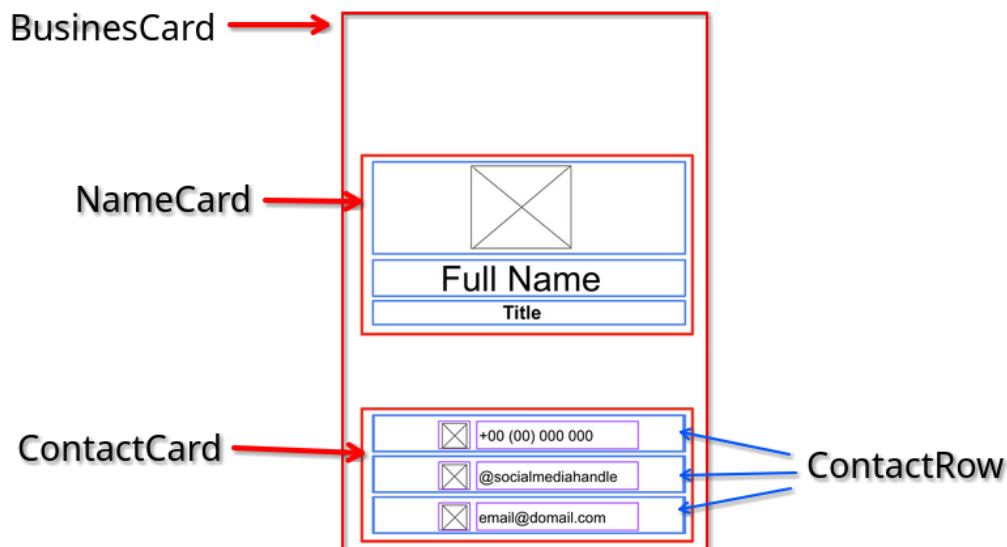Each of these files corresponds to a part of the card:

Figure 2: card

Each composable must accept a parameter for each piece of information. For example, the ContactRow composable could have the next signature:

```
@Composable
fun ContactRow(icon: ImageVector, text: String, modifier: Modifier = Modifier)
```

Finally, call the BusinessCard composable from the MainActivity:

```
class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            BusinessCard(
                name = "Ricardo Sánchez",
                job = "IT Teacher",
                phone = "+34 555 66 77 88",
                socialMedia = "@mymedia",
                email = "myemail@mail.com",
                image = R.drawable.tux_round
            )
        }
    }
```

```
    }
}
```

In addition, you **must** use at least the next composables: Text, Image, Icon, Row, Column, and optionally Card, Surface, etc. And use the appropiate modifiers to padding, center, etc.

1.2 How to submit to GitHub Classroom

1. You need a GitHub account. Create one if you don't have any.
2. Click on the assignment invitation link. Enter your GitHub credentials and authorize GitHub Classroom to access your GiHub account.
3. Select your name from the "Join the Classroom" list.
4. Accept the assignment.
5. Go to your assignment repository. Copy the repository URL (http or ssh), that is in the green "Code" button.
6. In your computer, clone the project on your practices folder:

```
git clone your_repo_url
```

7. Enter into your project folder and create your project inside.

8. Start coding.

9. Once you finish the task, make a commit with the comment **"SUBMISSION COMMIT PRACTICE 2.1"** and push it to GitHub. Before that, you can do the commits and pushes you want. If you change your code after your submission commit, make another commit and push with the same text in the message adding the corrections you've done (for instance: **"SUBMISSION COMMIT PRACTICE 2.1, corrected function mistake"**). Make a pull-request if you want so i could know your code is submitted.

If you have any doubt in your task, you can push your code and ask me by email what's your problem. It will make it easier for both the solutions of code issues.