

# Session 4 Lab Report

## Objective:

To build a console-based Business Card program in Kotlin using reusable functions for Logo, Header, and Contact sections. The goal is to understand composable-like structuring in a text-based environment.

## Kotlin Code:

```
data class BusinessCard(
    val name: String,
    val title: String,
    val phone: String,
    val email: String,
    val github: String
)

fun renderLogo(width: Int): String {
    val text = "MY LOGO"
    val padding = (width - text.length) / 2
    return "|" + " ".repeat(padding) + text +
        " ".repeat(width - text.length - padding) + "|\n"
}

fun renderHeader(name: String, title: String, width: Int): String {
    val nameLine = "|" + name.padEnd(width - 2) + "|\n"
    val titleLine = "|" + title.padEnd(width - 2) + "|\n"
    return nameLine + titleLine
}

fun renderContact(phone: String, email: String, github: String, width: Int): String {
    val phoneLine = "| Phone : $phone".padEnd(width) + "|\n"
    val emailLine = "| Email : $email".padEnd(width) + "|\n"
    val gitLine = "| GitHub: $github".padEnd(width) + "|\n"
    return phoneLine + emailLine + gitLine
}

fun renderBusinessCard(card: BusinessCard, width: Int = 40): String {
    val border = "+" + "-".repeat(width) + "+\n"
    return border +
        renderLogo(width) +
        renderHeader(card.name, card.title, width) +
        renderContact(card.phone, card.email, card.github, width) +
        border
}

fun main() {
    val card = BusinessCard(
        name = "Srajan Kumar",
        title = "Android Developer",
        phone = "+91 9876543210",
        email = "srajan@email.com",
        github = "github.com/srajan"
    )

    println(renderBusinessCard(card))
}
```

## Minimal Explanation:

1. BusinessCard data class stores all card information. 2. renderLogo() creates a centered ASCII logo section. 3. renderHeader() formats the name and title. 4. renderContact() structures phone, email, and GitHub details. 5. renderBusinessCard() combines all sections with borders. 6. main() creates a sample card and prints it.