


```

                                text = "This book is written by
JK Rowling",
                                style =
MaterialTheme.typography.bodyLarge,
                                )
                            }
                        }
                    }
                Row(
                    modifier = Modifier
                        .weight(1f)
                        .padding(top = 12.dp)
                ) {
                    Box(
                        modifier =
                        Modifier
                            .fillMaxHeight()
                            .weight(1f)
                            .padding(end = 12.dp)
                            .background(Color.Black)
                    ) {
                        Column(
                            modifier =
Modifier.align(Alignment.Center),
                        ) {
                            Text(
                                text = "Harry Potter",
                                style =
MaterialTheme.typography.headlineMedium
                                    .copy(color = Color.White),
                            )
                            Text(
                                text = "This book is written by
JK Rowling",
                                style =
MaterialTheme.typography.bodyLarge,
                                color = Color.White
                            )
                        }
                    }
                Box(

```

```

        modifier =
        Modifier
            .fillMaxHeight()
            .weight(1f)
            .padding(start = 12.dp)
            .clip(
                RoundedCornerShape(bottomEnd =
16.dp)
            )
            .background(Color.Magenta)
        )
    }
}
}
}
}

```

```

//@Preview
//@Composable
//fun PreviewForBoxes() {
//    LabExercisesTheme {
//        ForBoxes()
//    }
//}

```

```

@Composable
fun BookAndItsAuthor() {
    Column(
        modifier = Modifier
            .fillMaxWidth()
            .padding(24.dp),
        verticalArrangement = Arrangement.Center
    ) {
        Text(
            text = "The book name is Harry Potter",
            modifier = Modifier
                .padding(bottom = 12.dp)
                .align(Alignment.End),
            style =
MaterialTheme.typography.headlineLarge.copy(
                fontSize = 32.sp,
                lineHeight = 32.sp,

```

```

        textAlign = TextAlign.Center
    )
)
Text(
    modifier = Modifier.align(Alignment.End), text
= "Author JK Rowling",
    style = MaterialTheme.typography.bodyLarge
)
}
}

```

```

//@Preview
//@Composable
//fun PreviewBookAndItsAuthor() {
//    LabExercisesTheme {
//        BookAndItsAuthor()
//    }
//}

```

```

@Composable
fun StudentList(studentsName: List<String>) {
    Column {
        studentsName.forEach { student ->
            StudentListItem(studentName = student)
        }
    }
}

```

```

@Composable
fun StudentLazyList(studentNames: List<String>) {
    Column {
        Text(text = "STUDENTS", style =
MaterialTheme.typography.headlineMedium)
        LazyColumn(modifier = Modifier.weight(1f)) {
            items(studentNames) { student ->
                StudentListItem(studentName = student)
            }
        }
        Row(modifier = Modifier.fillMaxWidth()) {
            Text(

```



```

@Composable
fun StudentListItem(studentName: String) {
    Column {
        Text(
            modifier = Modifier
                .fillMaxWidth()
                .padding(vertical = 16.dp, horizontal =
12.dp),
            text = studentName
        )
        Divider()
    }
}

```

```

@Composable
fun RandomNumberGenerator() {
    var randomNumber by remember {
        mutableStateOf(0)
    }

    val onNextButtonClicked: () -> Unit = {
        randomNumber = (1..100).random()
        Log.d("RANDOM", "RandomNumberGenerator:
$randomNumber")
    }

    Box(
        modifier = Modifier
            .fillMaxSize()
            .background(MaterialTheme.colorScheme.secondary
)
    ) {
        Column(modifier = Modifier.align(Alignment.Center))
    {
        Box(
            modifier = Modifier
                .size(150.dp, 150.dp)
                .background(MaterialTheme.colorScheme.o

```

```

nSecondary)
    ) {
        Text(
            modifier =
Modifier.align(Alignment.Center),
            text = randomNumber.toString(),
            style =
MaterialTheme.typography.headlineLarge.copy(fontSize =
48.sp)
        )
    }
    Button(
        modifier = Modifier
            .padding(top = 64.dp)
            .align(Alignment.CenterHorizontally),
        onClick = onNextButtonClicked,
        colors =
ButtonDefaults.elevatedButtonColors(
            containerColor =
MaterialTheme.colorScheme.onSecondary,
        )
    ) {
        Text(text = "NEXT")
    }
}
}

@Preview
@Composable
fun PreviewRandomNumberGenerator() {
    LabExercisesTheme {
        RandomNumberGenerator()
    }
}

```