|  |
| --- |
| **학번 : 202114226, 이름 : 송재현** |
| **2024년 10주차 실습 체크 리스트** |
| |  |  | | --- | --- | | **실습 1** | **완성** | |
| **실습 1 : 앱 리스트 작성** |
| **소스코드(InstalledAppList.kt)** |
| package com.example.applistexample.lazyui  import android.content.Context import android.content.Intent import android.graphics.drawable.Drawable import android.widget.Toast import androidx.compose.foundation.Image import androidx.compose.foundation.clickable import androidx.compose.foundation.layout.Column import androidx.compose.foundation.layout.Row import androidx.compose.foundation.layout.Spacer import androidx.compose.foundation.layout.size import androidx.compose.foundation.layout.width import androidx.compose.foundation.lazy.LazyColumn import androidx.compose.foundation.lazy.items import androidx.compose.material3.Text import androidx.compose.runtime.Composable import androidx.compose.ui.Alignment import androidx.compose.ui.Modifier import androidx.compose.ui.graphics.Color import androidx.compose.ui.platform.*LocalContext* import androidx.compose.ui.unit.dp import androidx.compose.ui.unit.sp import coil.compose.rememberAsyncImagePainter  data class InstalledAppInfo(  var appLabel: String,  var appClassName: String,  var appPackName: String,  var appIcon: Drawable )  @Composable fun InstalledAppsList(apps: List<InstalledAppInfo>) {  LazyColumn **{** *items*(apps) **{** appInfo **->** AppItem(appInfo = appInfo)  **}  }** }  @Composable fun AppItem(appInfo: InstalledAppInfo) {  val context = *LocalContext*.current  Row(verticalAlignment = Alignment.CenterVertically,  modifier = Modifier.*clickable* **{** *launchApp*(context, appInfo.appPackName)  **}**) **{** Image(  painter = rememberAsyncImagePainter(model = appInfo.appIcon), *//coil-compose 추가* contentDescription = appInfo.appLabel,  modifier = Modifier.*size*(48.*dp*)  )  Spacer(modifier = Modifier.*width*(16.*dp*))  Column **{** Text(text = appInfo.appLabel, fontSize = 20.*sp*)  Text(text = appInfo.appPackName, fontSize = 14.*sp*, color = Color.Gray)  **}  }** }   fun launchApp(context: Context, packageName: String) {  val intent = context.*packageManager*.getLaunchIntentForPackage(packageName)  if (intent != null) {  context.startActivity(intent)  } else {  Toast.makeText(context, "앱을 실행할 수 없습니다.", Toast.*LENGTH\_SHORT*).show()  } }  @Composable fun getInstalledApps(): List<InstalledAppInfo> {  val packageManager = *LocalContext*.current.*packageManager* val intent = Intent(Intent.*ACTION\_MAIN*).*apply* **{** addCategory(Intent.*CATEGORY\_LAUNCHER*)  **}** val resolveInfos = packageManager.queryIntentActivities(intent, 0)  val apps = *mutableListOf*<InstalledAppInfo>()  for (info in resolveInfos) {  val appLabel = info.loadLabel(packageManager).toString()  val appClassName = info.activityInfo.name  val appPackName = info.activityInfo.packageName  val appIcon = info.loadIcon(packageManager)  apps.add(InstalledAppInfo(appLabel, appClassName, appPackName, appIcon))  }  return apps  } |
| **소스코드(MainActivity.kt)** |
| package com.example.a10chomework  import android.os.Bundle import androidx.activity.ComponentActivity import androidx.activity.compose.setContent import androidx.compose.foundation.layout.Arrangement import androidx.compose.foundation.layout.Row import androidx.compose.foundation.layout.fillMaxSize import androidx.compose.foundation.layout.fillMaxWidth import androidx.compose.material3.MaterialTheme import androidx.compose.material3.Surface import androidx.compose.material3.Text import androidx.compose.runtime.Composable import androidx.compose.ui.Modifier import androidx.compose.ui.tooling.preview.Preview import com.example.a10chomework.ui.theme.\_10cHomeWorkTheme import com.example.applistexample.lazyui.InstalledAppsList import com.example.applistexample.lazyui.getInstalledApps  class MainActivity : ComponentActivity() {  override fun onCreate(savedInstanceState: Bundle?) {  super.onCreate(savedInstanceState)  *setContent* **{** \_10cHomeWorkTheme **{** *// A surface container using the 'background' color from the theme* Surface(  modifier = Modifier.*fillMaxSize*(),  color = MaterialTheme.colorScheme.background  ) **{** Row(  modifier = Modifier.*fillMaxWidth*(),  horizontalArrangement = Arrangement.End,  ) **{** Text(text = "202114226 송재현")  **}** val apps = getInstalledApps()  InstalledAppsList(apps)  **}  }  }** } }  @Composable fun Greeting(name: String, modifier: Modifier = Modifier) {  }  @Preview(showBackground = true) @Composable fun GreetingPreview() {  \_10cHomeWorkTheme **{** Greeting("Android")  **}** } |
| **실행결과** |
|  |