|  |
| --- |
| **학번 : 202114226, 이름 : 송재현** |
| **2024년 11주차 실습 체크 리스트** |
| |  |  | | --- | --- | | **실습 1** | **완성** | |
| **실습 1 : html 파싱하기** |
| **소스코드** |
| -MainActivity.kt-  package com.example.week11  import android.os.Bundle import androidx.activity.ComponentActivity import androidx.activity.compose.setContent import androidx.compose.foundation.layout.fillMaxSize import androidx.compose.material3.MaterialTheme import androidx.compose.material3.Surface import androidx.compose.ui.Modifier import com.example.week11.example03.MainScreen import com.example.week11.ui.theme.Week11Theme  class MainActivity : ComponentActivity() {  override fun onCreate(savedInstanceState: Bundle?) {  super.onCreate(savedInstanceState)  *setContent* **{** Week11Theme **{** *// A surface container using the 'background' color from the theme* Surface(  modifier = Modifier.*fillMaxSize*(),  color = MaterialTheme.colorScheme.background  ) **{** MainScreen(url = "https://news.daum.net")  **}  }  }** } }  -HtmlParsing.kt-  package com.example.week11.example03  import androidx.compose.runtime.Composable import androidx.compose.runtime.LaunchedEffect import androidx.lifecycle.viewmodel.compose.viewModel  @Composable fun MainScreen(newsViewModel: NewsViewModel = viewModel(), url: String) {  val newsList = newsViewModel.newsList.value  LaunchedEffect(url) **{** newsViewModel.fetchNews(url)  **}** NewsList(list = newsList) }  -LazyUI.kt-  package com.example.week11.example03  import android.content.Intent import android.net.Uri import androidx.compose.foundation.clickable import androidx.compose.foundation.layout.Row import androidx.compose.foundation.lazy.LazyColumn import androidx.compose.foundation.lazy.items import androidx.compose.material3.Divider 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  data class NewsData(  var title: String,  var newsUrl: String )  @Composable fun NewsList(list: List<NewsData>) {  LazyColumn **{** item **{** Text(text = "202114226 송재현") **}** *items*(list) **{** item **->** NewsItem(item)  Divider(color = Color.Black, thickness = 1.*dp*)  **}  }** }  @Composable fun NewsItem(news: NewsData) {  val context = *LocalContext*.current  Row(  verticalAlignment = Alignment.CenterVertically,  modifier = Modifier.*clickable* **{** val intent = Intent(Intent.*ACTION\_VIEW*, Uri.parse(news.newsUrl))  context.startActivity(intent)  **}** ) **{** Text(news.title, fontSize = 20.*sp*)  **}** }  -NewsViewModel.kt-  package com.example.week11.example03  import android.util.Log import androidx.compose.runtime.MutableState import androidx.compose.runtime.mutableStateOf import androidx.lifecycle.ViewModel import androidx.lifecycle.*viewModelScope* import kotlinx.coroutines.Dispatchers import kotlinx.coroutines.launch import kotlinx.coroutines.withContext import org.jsoup.Jsoup  class NewsViewModel : ViewModel() {   private val \_newsList = *mutableStateOf*<List<NewsData>>(*emptyList*())  val newsList: MutableState<List<NewsData>> = \_newsList   fun fetchNews(url: String) {  *viewModelScope*.*launch* **{** try {  val fetchedNews = getNews(url)  \_newsList.value = fetchedNews  } catch (e: Exception) {  Log.e("error", "fetch 관련 오류 발생")  }  **}** }    private suspend fun getNews(url: String): List<NewsData> = withContext(Dispatchers.IO) **{** val doc = Jsoup.connect(url).get()  val headlines = doc.select("ul.list\_newsissue>li>div.item\_issue>div>strong.tit\_g>a")  headlines.*map* **{** news **->** NewsData(news.text(), news.absUrl("href")) **}  }** } |
| **실행결과** |
|  |