```kotlin
1  package com.bst.assignment3_20bce7458
2
3  import android.annotation.SuppressLint
4  import android.content.Intent
5  import android.net.Uri
6  import android.os.Bundle
7  import androidx.activity.ComponentActivity
8  import androidx.activity.compose.setContent
9  import androidx.compose.foundation.layout.Column
10 import androidx.compose.foundation.layout.fillMaxSize
11 import androidx.compose.foundation.layout.fillMaxWidth
12 import androidx.compose.foundation.layout.padding
13 import androidx.compose.material.icons.Icons
14 import androidx.compose.material.icons.filled.Search
15 import androidx.compose.material3.ExperimentalMaterial3Api
16 import androidx.compose.material3.Icon
17 import androidx.compose.material3.IconButton
18 import androidx.compose.material3.MaterialTheme
19 import androidx.compose.material3.Surface
20 import androidx.compose.material3.Text
21 import androidx.compose.material3.TextField
22 import androidx.compose.runtime.Composable
23 import androidx.compose.runtime.getValue
24 import androidx.compose.runtime.mutableStateOf
25 import androidx.compose.runtime.remember
26 import androidx.compose.runtime.setValue
27 import androidx.compose.ui.Modifier
```

```kotlin
28  import androidx.compose.ui.tooling.preview.Preview
29  import androidx.compose.ui.unit.dp
30  import androidx.lifecycle.viewmodel.compose.viewModel
31  import com.bst.assignment3_20bce7458.ui.theme.Assignment3_20BCE7458Theme
32
33  class MainActivity : ComponentActivity() {
34      override fun onCreate(savedInstanceState: Bundle?) {
35          super.onCreate(savedInstanceState)
36          setContent {
37              Assignment3_20BCE7458Theme {
38                  // A surface container using the 'background' color from the theme
39                  Surface(
40                      modifier = Modifier.fillMaxSize(),
41                      color = MaterialTheme.colorScheme.background
42                  ) {
43                      App()
44                  }
45              }
46          }
47      }
48  }
49  @SuppressLint("RememberReturnType")
50  @OptIn(ExperimentalMaterial3Api::class)
51  @Composable
52  fun App() {
53      val viewModel: YouTubeViewModel = viewModel()
54      val videos = viewModel.videos
```

```kotlin
55
56    var value by remember { mutableStateOf("") }
57
58    Column {
59        TextField(
60            value = value,
61            onValueChange = { value = it },
62            placeholder = { Text("Search videos") },
63            modifier = Modifier
64                .padding(16.dp)
65                .fillMaxWidth(),
66            trailingIcon = {
67                IconButton(onClick = { viewModel.searchVideos(value) }) {
68                    Icon(Icons.Default.Search, contentDescription = "Search")
69                }
70            }
71        )
72        YouTubeVideoList(videos, onClick = {
73            val intent = Intent(Intent.ACTION_VIEW, Uri.parse(it))
74            startActivity(intent)
75        })
76    }
77 }
78
79 @Preview(showBackground = true)
80 @Composable
81 fun GreetingPreview() {
```

```
82        Assignment3_20BCE7458Theme {
83            App()
84        }
85    }
86  }
87
88
```

```
1  package com.bst.assignment3_20bce7458
2
3  data class YouTubeVideo(
4      val title: String,
5      val description: String,
6      val thumbnailUrl: String,
7      val videoUrl: String
8  )
9
```

```kotlin
1  package com.bst.assignment3_20bce7458
2
3  import androidx.compose.foundation.clickable
4  import androidx.compose.foundation.layout.Column
5  import androidx.compose.foundation.layout.Spacer
6  import androidx.compose.foundation.layout.fillMaxWidth
7  import androidx.compose.foundation.layout.height
8  import androidx.compose.foundation.layout.padding
9  import androidx.compose.foundation.lazy.LazyColumn
10 import androidx.compose.foundation.lazy.items
11 import androidx.compose.foundation.shape.RoundedCornerShape
12 import androidx.compose.material3.Text
13 import androidx.compose.material3.TextButton
14 import androidx.compose.runtime.Composable
15 import androidx.compose.ui.Modifier
16 import androidx.compose.ui.draw.clip
17 import androidx.compose.ui.layout.ContentScale
18 import androidx.compose.ui.text.TextStyle
19 import androidx.compose.ui.text.font.FontWeight
20 import androidx.compose.ui.unit.dp
21 import androidx.compose.ui.unit.sp
22 import coil.compose.AsyncImage
23 import com.bst.assignment3_20bce7458.YouTubeVideo
24
25 @Composable
26 fun YouTubeVideoList(
27     videos: List<YouTubeVideo>,
```

```kotlin
28      onClick: (String) -> Unit
29  ) {
30      //update this code according to letest jetpack compose
31      LazyColumn {
32          items(items = videos) { video ->
33              Column(
34                  modifier = Modifier
35                      .padding(16.dp)
36                      .clickable { /* Handle video item click */ }
37              ) {
38                  AsyncImage(
39                      model = video.thumbnailUrl,
40                      contentDescription = "Profile picture",
41                      modifier = Modifier
42                          .height(200.dp)
43                          .fillMaxWidth()
44                          .clip(shape = RoundedCornerShape(8.dp)),
45                      contentScale = ContentScale.Crop
46                  )
47                  Spacer(modifier = Modifier.height(8.dp))
48                  TextButton(onClick = { onClick(video.videoUrl)}) {
49                      Text(
50                          text = video.title,
51                          style = TextStyle(fontSize = 18.sp, fontWeight = FontWeight.
    Bold)
52                      )
53                  }
```

```
54            Spacer(modifier = Modifier.height(4.dp))
55            Text(text = video.description)
56        }
57      }
58    }
59  }
60 }
```

```kotlin
1  package com.bst.assignment3_20bce7458
2
3  import androidx.compose.runtime.mutableStateListOf
4  import androidx.lifecycle.ViewModel
5  import androidx.lifecycle.viewModelScope
6  import kotlinx.coroutines.launch
7  import retrofit2.Retrofit
8  import retrofit2.converter.gson.GsonConverterFactory
9
10 class YouTubeViewModel : ViewModel() {
11     private val apiKey = "AIzaSyDylNfq0p2GeYfa0Uv-ojz8lTATf1to3tQ"  // Replace with
   your API key
12
13     private val retrofit = Retrofit.Builder()
14         .baseUrl("https://www.googleapis.com/youtube/v3/")
15         .addConverterFactory(GsonConverterFactory.create())
16         .build()
17
18     private val apiService = retrofit.create(YouTubeApiService::class.java)
19
20     private val _videos = mutableStateListOf<YouTubeVideo>()
21     val videos: List<YouTubeVideo>
22         get() = _videos
23
24     fun searchVideos(query: String) {
25         viewModelScope.launch {
26             try {
```

```kotlin
27      val response = apiService.searchVideos(apiKey, query =
        query)

28          _videos.clear()
29          _videos.addAll(response.items.map { item ->
30              val videoId = item.id.videoId
31              val snippet = item.snippet
32              YouTubeVideo(
33                  title = snippet.title,
34                  description = snippet.description,
35                  thumbnailUrl = snippet.thumbnails.medium.url,
36                  videoUrl = "https://www.youtube.com/watch?v=$videoId"
37              )
38          })
39      } catch (e: Exception) {
40          // Handle error
41      }
42  }
43  }
44  }
45
```

```kotlin
1  package com.bst.assignment3_20bce7458
2
3  import retrofit2.http.GET
4  import retrofit2.http.Query
5
6  interface YouTubeApiService {
7      @GET("search")
8      suspend fun searchVideos(
9          @Query("key") apiKey: String,
10         @Query("part") part: String = "snippet",
11         @Query("maxResults") maxResults: Int = 10,
12         @Query("q") query: String
13     ): YouTubeSearchResponse
14 }
15
16 data class YouTubeSearchResponse(
17     val items: List<YouTubeSearchResult>
18 )
19
20 data class YouTubeSearchResult(
21     val id: YouTubeVideoId,
22     val snippet: YouTubeVideoSnippet
23 )
24
25 data class YouTubeVideoId(
26     val videoId: String
27 )
```

```kotlin
28
29 data class YouTubeVideoSnippet(
30     val title: String,
31     val description: String,
32     val thumbnails: YouTubeVideoThumbnails
33 )
34
35 data class YouTubeVideoThumbnails(
36     val medium: YouTubeVideoThumbnail
37 )
38
39 data class YouTubeVideoThumbnail(
40     val url: String
41 )
42
43
```

```kotlin
1  package com.bst.assignment3_20bce7458.ui.theme
2
3  import androidx.compose.material3.Typography
4  import androidx.compose.ui.text.TextStyle
5  import androidx.compose.ui.text.font.FontFamily
6  import androidx.compose.ui.text.font.FontWeight
7  import androidx.compose.ui.unit.sp
8
9  // Set of Material typography styles to start with
10 val Typography = Typography(
11     bodyLarge = TextStyle(
12         fontFamily = FontFamily.Default,
13         fontWeight = FontWeight.Normal,
14         fontSize = 16.sp,
15         lineHeight = 24.sp,
16         letterSpacing = 0.5.sp
17     )
18     /* Other default text styles to override
19     titleLarge = TextStyle(
20         fontFamily = FontFamily.Default,
21         fontWeight = FontWeight.Normal,
22         fontSize = 22.sp,
23         lineHeight = 28.sp,
24         letterSpacing = 0.sp
25     ),
26     labelSmall = TextStyle(
27         fontFamily = FontFamily.Default,
```

```
28        fontWeight = FontWeight.Medium,
29        fontSize = 11.sp,
30        lineHeight = 16.sp,
31        letterSpacing = 0.5.sp
32    )
33    */
34 )
```

```
1 package com.bst.assignment3_20bce7458.ui.theme
2
3 import androidx.compose.ui.graphics.Color
4
5 val Purple80 = Color(0xFFD0BCFF)
6 val PurpleGrey80 = Color(0xFFCCC2DC)
7 val Pink80 = Color(0xFFEFB8C8)
8
9 val Purple40 = Color(0xFF6650a4)
10 val PurpleGrey40 = Color(0xFF625b71)
11 val Pink40 = Color(0xFF7D5260)
```

```kotlin
1  package com.bst.assignment3_20bce7458.ui.theme
2
3  import android.app.Activity
4  import android.os.Build
5  import androidx.compose.foundation.isSystemInDarkTheme
6  import androidx.compose.material3.MaterialTheme
7  import androidx.compose.material3.darkColorScheme
8  import androidx.compose.material3.dynamicDarkColorScheme
9  import androidx.compose.material3.dynamicLightColorScheme
10 import androidx.compose.material3.lightColorScheme
11 import androidx.compose.runtime.Composable
12 import androidx.compose.runtime.SideEffect
13 import androidx.compose.ui.graphics.toArgb
14 import androidx.compose.ui.platform.LocalContext
15 import androidx.compose.ui.platform.LocalView
16 import androidx.core.view.WindowCompat
17
18 private val DarkColorScheme = darkColorScheme(
19     primary = Purple80,
20     secondary = PurpleGrey80,
21     tertiary = Pink80
22 )
23
24 private val LightColorScheme = lightColorScheme(
25     primary = Purple40,
26     secondary = PurpleGrey40,
27     tertiary = Pink40
```

```kotlin
28
29      /* Other default colors to override
30      background = Color(0xFFFFFBFE),
31      surface = Color(0xFFFFFBFE),
32      onPrimary = Color.White,
33      onSecondary = Color.White,
34      onTertiary = Color.White,
35      onBackground = Color(0xFF1C1B1F),
36      onSurface = Color(0xFF1C1B1F),
37      */
38  )
39
40  @Composable
41  fun Assignment3_20BCE7458Theme(
42      darkTheme: Boolean = isSystemInDarkTheme(),
43      // Dynamic color is available on Android 12+
44      dynamicColor: Boolean = true,
45      content: @Composable () -> Unit
46  ) {
47      val colorScheme = when {
48          dynamicColor && Build.VERSION.SDK_INT >= Build.VERSION_CODES.S -> {
49              val context = LocalContext.current
50              if (darkTheme) dynamicDarkColorScheme(context) else
    dynamicLightColorScheme(context)
51          }
52
53          darkTheme -> DarkColorScheme
```

```
54          else -> LightColorScheme
55      }
56      val view = LocalView.current
57      if (!view.isInEditMode) {
58          SideEffect {
59              val window = (view.context as Activity).window
60              window.statusBarColor = colorScheme.primary.toArgb()
61              WindowCompat.getInsetsController(window, view).
    isAppearanceLightStatusBars = darkTheme
62          }
63      }
64
65      MaterialTheme(
66          colorScheme = colorScheme,
67          typography = Typography,
68          content = content
69      )
70  }
```