

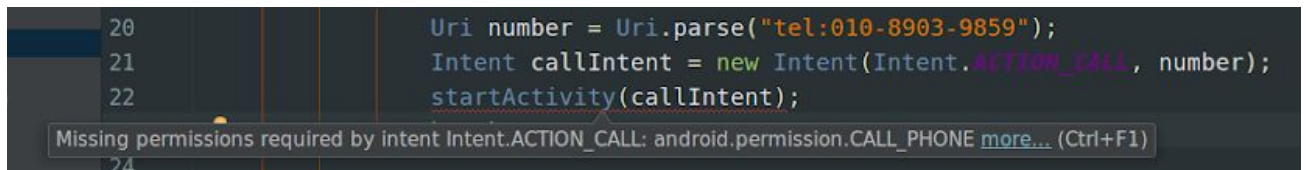
# 모바일 프로그래밍 실습 보고서

학번	201411203	이름	박소영
순서	10	날짜	2018.04.12 목요일
강의자료	06 - 화면전환		
주제	인텐트와 화면 전환 및 권한 체크		

- **화면전환** : 클릭했을 때 새로운 화면으로 넘어간다.
    1. 새로운 액티비티를 추가해야 한다.
    2. 추가한 액티비티를 Manifest에 꼭 등록해줘야 한다.
    3. startActivity(Intent intent)로 액티비티를 호출, 실행할 액티비티의 정보는 인텐트가 가지게 됨.
  - **인텐트** : 각 화면은 독립적이므로 중간에 연결해주는 다리가 필요하다.
    - 컴포넌트끼리 통신하기 위한 메시지 시스템.
    - 어느 뷰에 어느 데이터를 매핑시켜 줄 것인지 정해야 한다.
    - **putExtra(이름, 내용) 함수** : 다음 액티비티로 이름을 지정해 데이터를 보낼 수 있다.
1. **명시적인 인텐트** : 인텐트를 수행할 class를 명시적으로 지정함

```
Intent intent = new Intent(지금 액티비티.this, 다음 액티비티 class);
```

## + 권한에 대한 이야기



- call 같은 경우에는 전화가 바로 가는 액션이며, 비용이 청구되게 됨.
  - 이런 경우 사용자가 허용을 해줘야 전화를 걸 수 있음! 권한을 주는 작업이 필요함.
    1. 권한 정보를 Manifest 파일에 등록한다.
    2. 권한 체크를 프로그램이 시작될 때 해줘야 한다. **checkSelfPermission 함수** (Lec06\_9p)
    3. 권한이 주어지지 않았을 경우 요청한다. **requestPermissions 함수** (Lec06\_10p)
    4. 요청이 끝났는지 아닌지 후처리. **onRequestPermissionsResult 함수** (Lec06\_11p)
- + Can only use lower 8 bits for requestCode : **requestCode는 8비트 이하만 가능!**
2. **묵시적인 인텐트** : 수행할 Action만 명시해주면 됨. 정확한 클래스 이름 모름. (Lec06\_7p)

```
Intent intent = new Intent(Intent.액션이름);
```

- **getIntent() 함수** : 받는 쪽에서 보내온 인텐트의 정보를 확인할 수 있다. 담긴 걸 꺼낼 수 있음.
- 시작 액티비티 지정하기 : 매니페스트에서 해당 액티비티의 자식 노드로 이거 추가.

```
<category android:name="android.intent.category.LAUNCHER" />
```

## AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="moblieprogramming.a20180412">
    <uses-permission android:name="android.permission.CALL_PHONE" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
            </intent-filter>
        </activity>
        <activity android:name=".IntroActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

## java 소스코드 : IntroActivity.java

```
public class IntroActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_intro);
    }

    public void TestButton(View view) {
        Intent intent = new Intent(this, MainActivity.class);
        startActivity(intent);
    }
}
```

## Layout : activity\_intro.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context="moblieprogramming.a20180412.IntroActivity">
    <TextView
        android:layout_width="match_parent" android:layout_height="0dp"
        android:layout_weight="5"          android:gravity="center"
        android:text="201411203 박소영" />
    <Button
        android:layout_width="match_parent" android:layout_height="0dp"
        android:layout_weight="1"          android:text="BUTTON"
        android:onClick="TestButton" />
</LinearLayout>
```

## java 소스코드 : MainActivity.java

```
public class MainActivity extends AppCompatActivity {
    final int CALL_REQUEST = 8; // 승인 요청 시 쓸 요청에 해당하는 상수 값 선언

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    // 1. 퍼미션 체크
    boolean checkAppPermission(String[] requestPermission){
        boolean[] requestResult = new boolean[requestPermission.length];
        for(int i=0; i< requestResult.length; i++){
            requestResult[i] = (ContextCompat.checkSelfPermission (this,
                requestPermission[i]) == PackageManager.PERMISSION_GRANTED);
            if(!requestResult[i]){
                return false;
            }
        }
        return true;
    }
}
```

```

// 2. 승인을 요청
// param : (컨텍스트 정보, 요구하는 퍼미션의 배열, 요청에 해당하는 번호 정보)
void askPermission(String[] requestPermission, int REQ_PERMISSION) {
    ActivityCompat.requestPermissions(this, requestPermission,
        REQ_PERMISSION);
}

// 3. 요청결과 확인
// 여기서 동의했을 때, 거절했을 때의 액션을 처리해주면 된다.
// param : (요청한 퍼미션 정보, 승인, 거절한 배열 정보)
@Override
public void onRequestPermissionsResult (int requestCode,
    @NonNull String[] permissions, @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions,
        grantResults);
    switch (requestCode) {
        case CALL_REQUEST :
            if (checkAppPermission(permissions)) {
                Uri num = Uri.parse("tel:010-8903-9859");
                Intent callIntent = new Intent(Intent.ACTION_CALL, num);
                startActivity(callIntent);
            } else {
                Toast.makeText(getApplicationContext(),
                    "권한이 거절되었습니다.", Toast.LENGTH_SHORT).show();
                finish(); // 앱을 중지시키는 함수
            }
            break;
    }
}

public void btnClick(View view) {
    switch(view.getId()) {
        case R.id.call:
            Uri number = Uri.parse("tel:010-8903-9859");
            Intent callIntent = new Intent(Intent.ACTION_CALL, number);

            // 해당 권한들이 매니페스트에 등록되어 있는지 체크한다.
            if(checkAppPermission(
                new String[]{Manifest.permission.CALL_PHONE})){
                startActivity(callIntent);
            } else {
                // 권한 체크가 되어있지 않을 경우 먼저 권한을 요청한다.
                // 요청할 권한 정보와 코드를 인자로 준다.
            }
        }
    }
}

```

```

        askPermission(new String[]
            {Manifest.permission.CALL_PHONE}, CALL_REQUEST);
    }
    break;

    case R.id.sms:
        Uri msg = Uri.parse("sms:010-8903-9859");
        Intent messageIntent = new Intent(Intent.ACTION_SENDTO, msg);
        messageIntent.putExtra("sms_body", "Hello World");
        startActivity(messageIntent);
        break;

    case R.id.web:
        Uri webpage = Uri.parse("http://www.naver.com");
        Intent webIntent = new Intent(Intent.ACTION_VIEW, webpage);
        startActivity(webIntent);
        break;

    case R.id.map:
        Uri location = Uri.parse("geo:37.543684, 127.077130?z=16");
        Intent mapIntent = new Intent(Intent.ACTION_VIEW, location);
        startActivity(mapIntent);
        break;
    }
}
}
}

```

### Layout : activity\_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context="moblieprogramming.a20180412.MainActivity">

    <Button android:id="@+id/call"
        android:onClick="btnClick" android:text="CALL"/>
    <Button android:id="@+id/web"
        android:onClick="btnClick" android:text="NAVER"/>

```

```

<Button android:id="@+id/sms"
        android:onClick="btnClick" android:text="SMS"/>
<Button android:id="@+id/map"
        android:onClick="btnClick" android:text="MAP"/>
</LinearLayout>

```

## 실행 결과

