

TimePicker 위젯 익히기

배 희호 교수
경북대학교
스마트IT과



시간 및 날짜 관련 Widget

■ DatePicker

- 날짜를 입력 받는 View
- API 21(롤리팝) 이후부터 DatePicker의 모양이 변경되었기 때문에 이전 모양을 유지하려면 datePickerMode 속성에 'spinner'를 입력

■ TimePicker

- 시간을 입력 받는 View
- DatePicker와 마찬가지로 이전 모양을 유지하기 위해 timePickerMode에 'spinner'를 입력



TimePicker 위젯



- 시간을 입력 받으며 특별한 속성 없이 배치하면 자동으로 동작함
- 시간과 분만 입력 받으며 초 단위는 입력 받지 못함
- 오전/오후 선택 가능
- 시간은 24시간제로 표현 가능하며 AM/PM 방식의 12시간제로도 표현 가능 (디폴트는 12시간제)



TimePicker 위젯



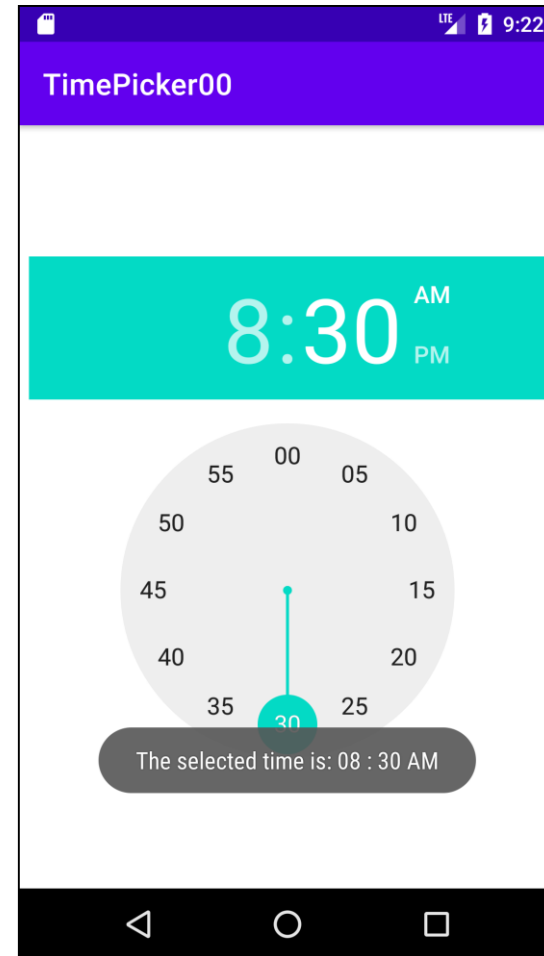
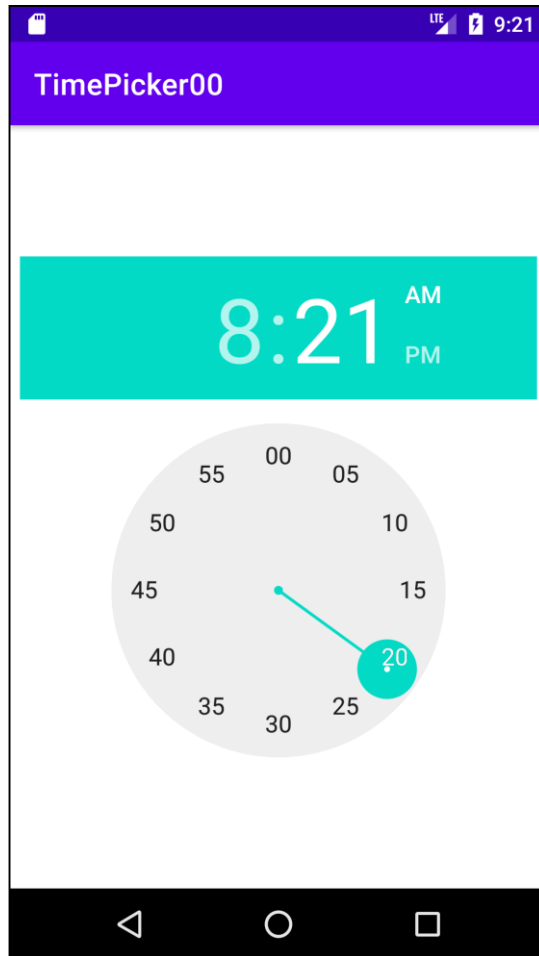
■ 메소드

메소드	설명
<code>void setCurrentHour (Integer currentHour)</code>	현재 시간을 설정
<code>void setCurrentMinute (Integer currentMinute)</code>	현재 분을 설정
<code>void setIs24HourView (Boolean is24HourView)</code>	✓ 24시간 또는 AM/PM 모드로 설정 ✓ 초기값을 설정 할 수 있음 ✓ 시간은 0~23, 분은 0~59 사이의 int 값을 입력
<code>boolean is24HourView()</code>	24시간 보기인 경우 true로 반환
<code>boolean isEnabled()</code>	View에 대한 사용 가능 상태를 반환
<code>void setEnabled (boolean enabled)</code>	View에 대한 enabled state를 세팅 함
<code>void setOnTimeChangeListener (TimePicker.OnTimeChange dListener callback)</code>	abstract <code>onTimeChanged()</code> 메소드가 구현된 Listener 객체를 등록하여 사용자 입력을 저장/처리



TimePicker 예제 0

- Timepicker를 이용해 설정된 시각 출력하는 프로그램을 만들어보자





TimePicker 예제 0



■ activity_main.xml

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

    <TimePicker
        android:id="@+id/timePicker"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

</LinearLayout>
```



TimePicker 예제 0

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);
```

```
        TimePicker timePicker = findViewById(R.id.timePicker);  
        timePicker.setOnTimeChangedListener(  
            new TimePicker.OnTimeChangedListener() {
```

```
            @Override
```

```
            public void onTimeChanged(TimePicker view, int hourOfDay, int minute) {  
                String format = "";  
                if (hourOfDay == 0) {  
                    hourOfDay += 12;  
                    format = "AM";  
                } else if (hourOfDay == 12) {  
                    format = "PM";
```



TimePicker 예제 0

■ MainActivity.JAVA

```
    } else if (hourOfDay > 12) {
        hourOfDay -= 12;
        format = "PM";
    } else {
        format = "AM";
    }
    String hour = String.valueOf(hourOfDay < 10 ? "0" +
                                   hourOfDay : hourOfDay);
    String min = String.valueOf(minute < 10 ? "0" + minute : minute);
    String text = getString(R.string.selected_time) + " " + hour + " : "
                                   + min + " " + format;

    Toast.makeText(getApplicationContext(), text,
                                   Toast.LENGTH_SHORT).show();
    }
});
}
```




TimePicker 예제 0

■ activity_main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/container"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity2">

</LinearLayout>
```



TimePicker 예제 0

■ MainActivity2.JAVA

```
public class MainActivity2 extends AppCompatActivity {
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main2);
```

```
        TimePicker timePicker = new TimePicker(this);
```

```
        LinearLayout.LayoutParams layoutParams = new LinearLayout.LayoutParams(  
            ViewGroup.LayoutParams.WRAP_CONTENT,  
            ViewGroup.LayoutParams.WRAP_CONTENT);
```

```
        layoutParams.setMargins(20, 20, 20, 20);
```

```
        timePicker.setLayoutParams(layoutParams);
```



TimePicker 예제 0

■ MainActivity2.JAVA

```
timePicker.setOnTimeChangeListener(  
    new TimePicker.OnTimeChangeListener() {  
        @Override  
        public void onTimeChanged(TimePicker view, int hourOfDay, int minute) {  
            String format = "";  
            if (hourOfDay == 0) {  
                hourOfDay += 12;  
                format = "AM";  
            } else if (hourOfDay == 12) {  
                format = "PM";  
            } else if (hourOfDay > 12) {  
                hourOfDay -= 12;  
                format = "PM";  
            } else {  
                format = "AM";  
            }  
        }  
    }  
);
```



TimePicker 예제 0

■ MainActivity2.JAVA

```
String hour = String.valueOf(hourOfDay < 10 ? "0" + hourOfDay :  
                                                                    hourOfDay);  
String min = String.valueOf(minute < 10 ? "0" + minute : minute);  
String text = getString(R.string.selected_time) + " " + hour + " : " +  
                                                    min + " " + format;
```

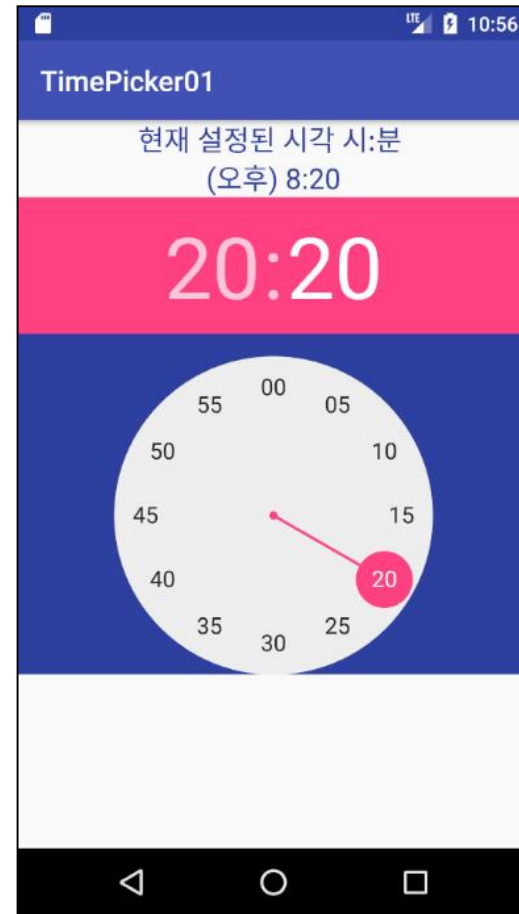
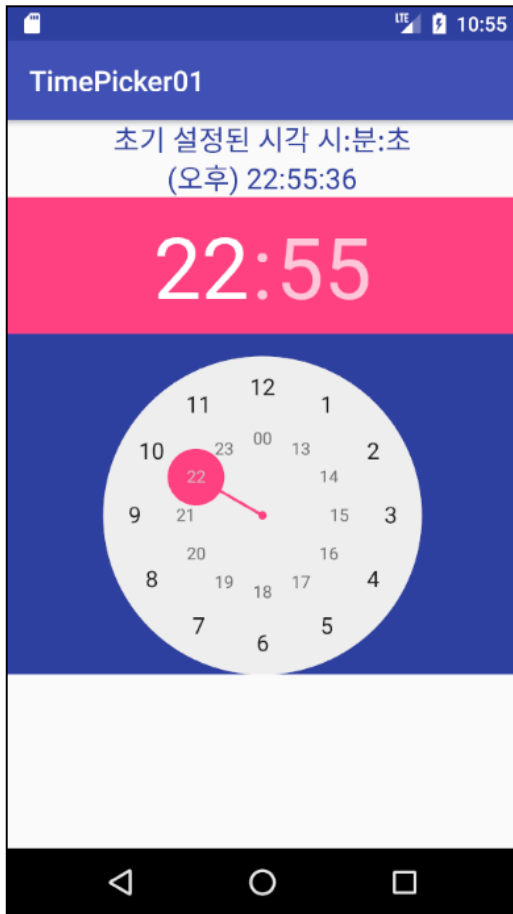
```
Toast.makeText(getBaseContext(), text, Toast.LENGTH_SHORT).show();  
}  
});
```

```
LinearLayout linearLayout = findViewById(R.id.container);  
if (linearLayout != null) {  
    linearLayout.addView(timePicker);  
}  
}  
}
```



TimePicker 예제 1

- Timepicker를 이용해 설정된 시각 출력하는 프로그램을 만들어보자





TimePicker 예제 1

■ 사용자 인터페이스

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textSize="20dp"
        android:text="초기 설정된 시각 시:분:초"
        android:textColor="@color/colorPrimaryDark"/>
```



TimePicker 예제 1



■ 사용자 인터페이스

```
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:textSize="20dp"
    android:textColor="@color/colorPrimaryDark"/>

<TimePicker
    android:id="@+id/timePicker"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:background="@color/colorPrimaryDark"/>
</LinearLayout>
```



TimePicker 예제 1

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity
    implements TimePicker.OnTimeChangedListener{
    TextView textView, textView1;
    TimePicker timePicker;
    Calendar calendar;
    int hour, minute, second;
    int am;

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

        calendar = Calendar.getInstance(); //캘린더 객체 생성
        hour = calendar.get(calendar.HOUR_OF_DAY); //현재 시
        minute = calendar.get(calendar.MINUTE); //현재 분
        second = calendar.get(calendar.SECOND); //현재 초
        am = calendar.get(calendar.AM_PM);
```




TimePicker 예제 1

■ MainActivity.JAVA

```
textView = findViewById(R.id.textView);
textView1 = findViewById(R.id.textView1);
timePicker = findViewById(R.id.timePicker);

if (am == calendar.AM){
    textView1.setText("(오전) " + hour + ":" + minute + ":" + second);
} else{
    textView1.setText("(오후) " + hour + ":" + minute + ":" + second);
}
timePicker.setIs24HourView(true);
timePicker.setOnTimeChangedListener(this);
}
```



TimePicker 예제 1

■ MainActivity.JAVA

@Override

```
public void onTimeChanged(TimePicker timePicker, int i, int i1) {  
    textView.setText("현재 설정된 시각 시:분 ");  
    if (i > 12) {  
        textView1.setText("(오후) " + (i - 12) + ":" + i1 );  
    } else if (i == 12){  
        textView1.setText("(오후) " + i + " : " + i1 );  
    } else {  
        textView1.setText(" (오전) " + i + " : " + i1 );  
    }  
}  
}
```



TimePicker 예제 1



- 시간 선택 Widget
 - TimePicker

- 사용 방법
 - 레이아웃 디자인(layout/picker_test.xml)
 - 시간을 변경한 경우 실행되는 리스너 등록
 - TimePicker 리스너(OnTimeChangeListener) 등록 -
> setOnTimeChangeListener() 메소드로 등록



TimePicker 예제 2

- Timepicker를 이용해 설정된 시각 출력하는 프로그램을 만들어보자





TimePicker 예제 1

- API 21(롤리팝) 이후부터 DatePicker의 모양이 변경되었기 때문에 이전 모양을 유지하려면 datePickerMode 속성에 spinner를 입력해야 함
- TimePicker도 데이트피커와 마찬가지로 이전 모양을 유지하기 위해 timePickerMode에 spinner를 입력해야 함
- 시간 [시간, 분]을 설정
 - 시간(0~23) 및 분(0~59) 오전/오후 토글
- 콜백 개체 제공
 - OnTimeChangeListener 또는 OnTimeSetListener
 - 사용자가 새 시간을 선택한 경우 알림을 받을 수 있음



TimePicker 예제 2

■ 사용자 인터페이스

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textSize="18dp"
        android:textColor="@color/colorPrimaryDark"/>
```



TimePicker 예제 2

■ 사용자 인터페이스

<TimePicker

```
    android:id="@+id/timePicker "  
    android:layout_width="wrap_content "  
    android:layout_height="wrap_content "  
    android:layout_gravity="center "  
    android:background="@color/colorPrimaryDark "  
    android:timePickerMode="spinner " />
```

<TextView

```
    android:id="@+id/textView1 "  
    android:layout_marginTop="20dp "  
    android:layout_width="wrap_content "  
    android:layout_height="wrap_content "  
    android:layout_gravity="center "  
    android:textSize= "18dp "  
    android:textColor="@color/colorAccent " />
```



TimePicker 예제 2

■ 사용자 인터페이스

```
<TimePicker
    android:id="@+id/timePicker1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:background="@color/colorAccent"
    android:timePickerMode="spinner" />
</LinearLayout>
```




TimePicker 예제 2

■ JAVA 프로그램

```
public class MainActivity extends AppCompatActivity
    implements TimePicker.OnTimeChangedListener{
    TextView textView, textView1;
    TimePicker timePicker, timePicker1;
    Calendar calendar;
    int hour, minute, second;
    int am;
```

@Override

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    calendar = Calendar.getInstance(); //캘린더 객체 생성
    hour = calendar.get(calendar.HOUR_OF_DAY); //현재 시
    minute = calendar.get(calendar.MINUTE); //현재 분
    second = calendar.get(calendar.SECOND); //현재 초
    am = calendar.get(calendar.AM_PM);
```



TimePicker 예제 2

■ JAVA 프로그램

```
textView = findViewById(R.id.textView);
timePicker = findViewById(R.id.timePicker);
textView1 = findViewById(R.id.textView1);
timePicker1 = findViewById(R.id.timePicker1);
if (am == calendar.AM){
    textView.setText("초기 설정된 시각 시:분:초 -> (오전) "+
        hour + ":" + minute + ":" + second);
    textView1.setText("초기 설정된 시각 시:분:초 -> (오전) "+
        hour + ":" + minute + ":" + second);
} else{
    textView.setText("초기 설정된 시각 시:분:초 -> (오후) "+
        hour + ":" + minute + ":" + second);
    textView1.setText("초기 설정된 시각 시:분:초 -> (오후) "+
        hour + ":" + minute + ":" + second);
}
timePicker.setIs24HourView(true);
timePicker1.setIs24HourView(false);
timePicker.setOnTimeChangedListener(this);
```



TimePicker 예제 2



```
timePicker1.setOnTimeChangeListener(  
    new TimePicker.OnTimeChangeListener() {  
        @RequiresApi(api = Build.VERSION_CODES.M)  
        @Override  
        public void onTimeChanged(TimePicker timePicker, int i, int i1) {  
            timePicker.is24HourView();  
            int hour = timePicker.getHour();  
            if (hour > 12) {  
                textView1.setText("현재 설정된 시각 시:분 -> (오후) "  
                    + (hour - 12) + ":" + i1 );  
            } else if (hour == 12){  
                textView1.setText("현재 설정된 시각 시:분 -> (오후) "  
                    + hour + ":" + i1);  
            } else {  
                textView1.setText("현재 설정된 시각 시:분 -> (오전) "  
                    + hour + ":" + i1 );  
            }  
        }  
    });  
}
```



TimePicker 예제 2

■ JAVA 프로그램

@Override

```
public void onTimeChanged(TimePicker timePicker, int i, int i1) {  
    if (i > 12) {  
        textView.setText("현재 설정된 시각 시:분 -> (오후) " +  
                           (i - 12) + ":" + i1 );  
    } else if (i == 12){  
        textView.setText("현재 설정된 시각 시:분 -> (오후) " +  
                           i + ":" + i1 );  
    } else {  
        textView.setText("현재 설정된 시각 시:분 -> (오전) " +  
                           i + ":" + i1 );  
    }  
}
```



TimePicker 예제 3

- TimePicker의 글자 크기와 색상, 구분선의 색상, 배경색을 변경하는 프로그램을 만들어보자





TimePicker 예제 3

■ 사용자 인터페이스

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

    <TimePicker
        android:id="@+id/timepicker"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:timePickerMode="spinner"
        android:theme="@style/TimePickerCustomSize"/>
```



TimePicker 예제 3

■ 사용자 인터페이스

```
<TextView
    android:id="@+id/txtdate"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:textSize="20dp"
    android:text="time"/>
```

```
<Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="선택한 시간 조사"/>
```

```
</LinearLayout>
```



TimePicker 예제 3

■ styles.xml

```
<resources>
```

```
    <!-- Base application theme. -->
```

```
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
```

```
        <!-- Customize your theme here. -->
```

```
        <item name="colorPrimary">@color/colorPrimary</item>
```

```
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
```

```
        <item name="colorAccent">@color/colorAccent</item>
```

```
    </style>
```

```
    <style name="TimePickerCustomSize">
```

```
        <item name="android:textSize">25dp</item>
```

```
        <item name="colorControlNormal">#FF0000</item>
```

```
        <item name="android:background">#00FFFF</item>
```

```
        <item name="android:editTextColor">@color/BLUE</item>
```

```
    </style>
```

```
</resources>
```




TimePicker 예제 3

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {
```

```
    @RequiresApi(api = Build.VERSION_CODES.M)
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
        Calendar calendar = Calendar.getInstance(); //캘린더 객체 생성
```

```
        int hour = calendar.get(calendar.HOUR_OF_DAY); //현재 시
```

```
        int minute = calendar.get(calendar.MINUTE); //현재 분
```

```
        int second = calendar.get(calendar.SECOND);
```

```
        int am = calendar.get(calendar.AM_PM);
```



TimePicker 예제 3

■ MainActivity.JAVA

```
final TextView textView = findViewById(R.id.txtdate);
if (am == calendar.AM){
    textView.setText(String.format("(오전) %02d시 %02d분 %02d초",
                                   hour, minute, second));
} else{
    textView.setText(String.format("(오후) %02d시 %02d분 %02d초",
                                   hour, minute, second));
}
final TimePicker picker = findViewById(R.id.timepicker);
// picker.setIs24HourView(true);
picker.setOnTimeChangedListener(new TimePicker.OnTimeChangedListener() {
    @Override
    public void onTimeChanged(TimePicker view, int hourOfDay, int minute) {
        if (hourOfDay > 12) {
            textView.setText(String.format("(오후) %02d시 %02d분",
                                           hourOfDay-12, minute));
        } else if (hourOfDay == 12){
            textView.setText(String.format("(오후) %02d시 %02d분",
                                           hourOfDay, minute));
        }
    }
});
```



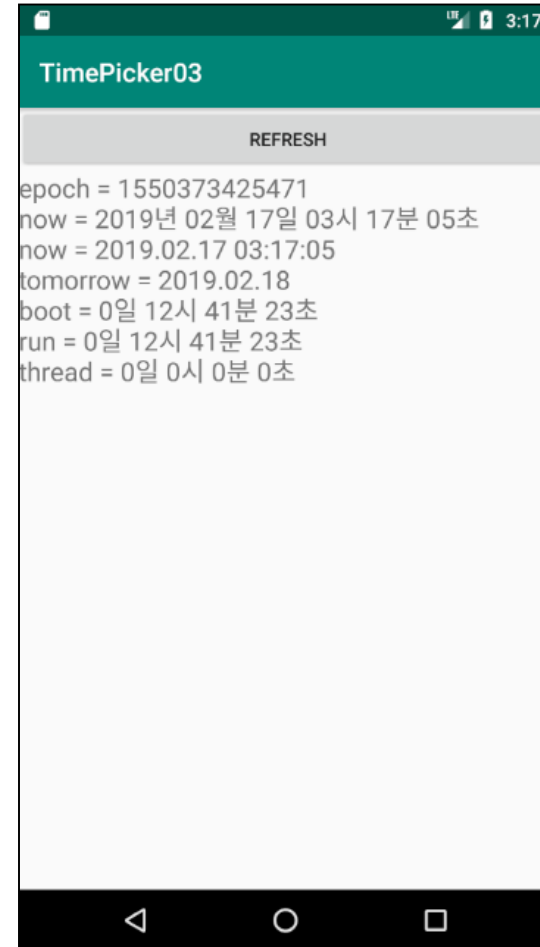
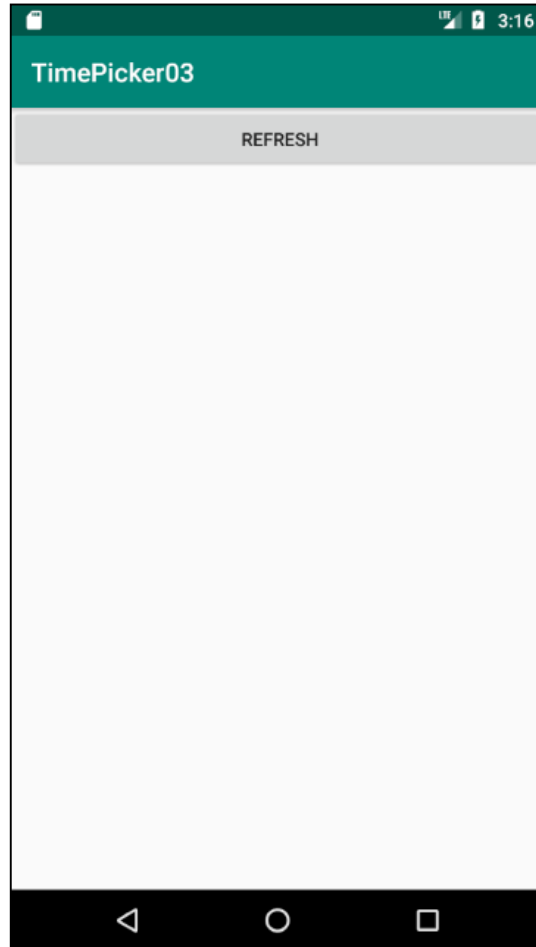
TimePicker 예제 3

■ MainActivity.JAVA

```
        } else {
            textView.setText(String.format("(오전) %02d시 %02d분",
                                            hourOfDay, minute));
        }
    }
});
Button button = findViewById(R.id.button);
button.setOnClickListener(new View.OnClickListener() {
    @RequiresApi(api = Build.VERSION_CODES.M)
    public void onClick(View v) {
        String result = String.format("%02d시 %02d분", picker.getHour(),
                                        picker.getMinute());
        Toast.makeText(getApplicationContext(), result,
                        Toast.LENGTH_SHORT).show();
    }
});
}
```



TimePicker 예제 3





TimePicker 예제 3



■ System의 정적 메소드 이용

```
static long currentTimeMillis ()
```

- 현재 시간 값을 정수 하나로 리턴
- 에폭 타임(Epoch Time) 기준
 - 1970년 1월 1일 자정을 기준으로 한 1/1000초 단위의 경과 시간
 - 년, 월, 일, 시, 분, 초의 요소들을 개별적이 아닌 단 하나의 값으로 일차원화하여 표현하므로 전달, 저장, 비교가 용이함
 - 출력용으로는 적합하지 않음



Epoch 시간



- 유닉스 시간(Unix time)은 시각을 나타내는 방식
- POSIX 시간이나 Epoch 시간이라고 부르기도 함
- 1970년 1월 1일 00:00:00 협정 세계 시(UTC)부터의 경과 시간을 초로 환산하여 정수로 나타낸 것
- 유닉스 시간에서 윤초는 무시됨
- 유닉스 계열 운영체제나 여러 다른 운영체제, 그리고 파일 형식 등에서 사용
- 윤초 처리 방식 때문에 시간을 선형으로 표현하지 못하며, 협정 세계 시의 윤초를 표현할 수 없음
- 유닉스 시간은 대다수의 유닉스 운영 체제에서 `date +%s`를 명령행에 입력하여 확인할 수 있음
- 32비트로 표현된 유닉스 시간은 1970년 1월 1일 00:00 (UTC)에서 2,147,483,647 ($2^{31} - 1$) 지난 후인 2038년 1월 19일 03:14:08 UTC에 2038년 문제를 발생



TimePicker 예제 3



■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <Button
        android:id="@+id/button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text=" REFRESH"/>
    <TextView
        android:id="@+id/result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20dp"/>
</LinearLayout>
```



TimePicker 예제 3

■ JAVA 프로그램

```
public class MainActivity extends AppCompatActivity {  
  
    @RequiresApi(api = Build.VERSION_CODES.M)  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        Button button = findViewById(R.id.button);  
  
        button.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                StringBuilder time = new StringBuilder();  
            }  
        });  
    }  
}
```




TimePicker 예제 3

■ JAVA 프로그램

```
long epoch = System.currentTimeMillis();
time.append("epoch = " + epoch + "\n");
SimpleDateFormat sdf =
    new SimpleDateFormat("yyyy년 MM월 dd일 hh시 mm분 ss초");
time.append("now = " + sdf.format(epoch) + "\n");

Date now = new Date();
SimpleDateFormat sdf1 =
    new SimpleDateFormat("yyyy.MM.dd hh:mm:ss");
time.append("now = " + sdf1.format(now) + "\n");

Calendar tom = Calendar.getInstance();
tom.add(Calendar.DAY_OF_MONTH, 1);
Date tomdate = tom.getTime();
SimpleDateFormat sdf2 = new SimpleDateFormat("yyyy.MM.dd");
time.append("tomorrow = " + sdf2.format(tomdate) + "\n");
```



TimePicker 예제 3

■ JAVA 프로그램

```
time.append("boot = " +  
            UpTime(SystemClock.elapsedRealtime()));  
time.append("run = " + UpTime(SystemClock.uptimeMillis()));  
time.append("thread = " +  
            UpTime(SystemClock.currentThreadTimeMillis()));  
  
TextView result = (TextView)findViewById(R.id.result);  
result.setText(time.toString());  
    }  
});  
}
```



TimePicker 예제 3

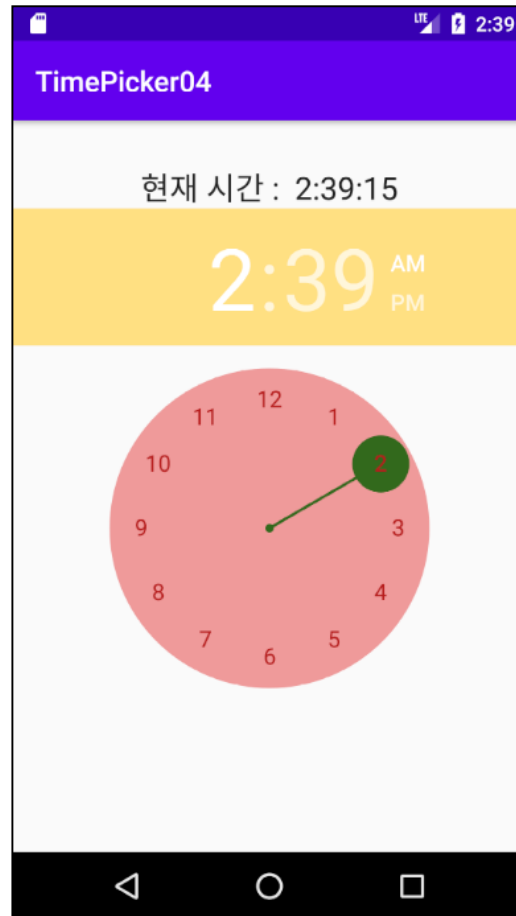
■ JAVA 프로그램

```
private String UpTime(long currentThreadTimeMillis) {  
    long sec = currentThreadTimeMillis / 1000;  
    String result;  
    result = String.format("%d일 %d시 %d분 %d초\n", sec / 86400,  
        sec / 3600 % 24, sec / 60 % 60, sec % 60);  
    return result;  
}  
}
```



TimePicker 예제 4

- TimePicker의 색상을 변경하여보자





TimePicker 예제 4

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center|top"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:text="현재 시간 : "
        android:textAppearance="?android:attr/textAppearanceLarge" />
```



TimePicker 예제 4

■ 사용자 인터페이스

```
<TimePicker  
    android:id="@+id/picker"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:theme="@style/MyDatePickerWidgetStyle"  
    style="@style/MyDatePickerWidgetStyle"/>
```

```
</LinearLayout>
```



TimePicker 예제 4

■ styles.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <style name="MyDatePickerWidgetStyle" >
        <item name="android:headerBackground">#ffe082</item>
        <item name="android:numbersTextColor">#b71c1c</item>
        <item name="android:numbersInnerTextColor">#f44336</item>
        <item name="android:numbersSelectorColor">#33691e</item>
        <item name="android:numbersBackgroundColor">#ef9a9a</item>
        <item name="android:amPmTextColor">#9c27b0</item>
    </style>
</resources>
```



TimePicker 예제 4

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {  
    TextView textview;  
    int hour, minute, second;  
  
    @RequiresApi(api = Build.VERSION_CODES.O)  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        textview = findViewById(R.id.textView1);  
        Calendar calendar = Calendar.getInstance();  
        hour = calendar.get(Calendar.HOUR_OF_DAY);  
        minute = calendar.get(Calendar.MINUTE);  
        second = calendar.get(Calendar.SECOND);  
    }  
}
```




TimePicker 예제 4

■ MainActivity.JAVA

```
textView.setText(String.format("현재 시간 : %2d:%2d:%2d",
                                hour, minute, second));

TimePicker picker = findViewById(R.id.picker);
picker.setOnTimeChangeListener(
    new TimePicker.OnTimeChangeListener() {
        @Override
        public void onTimeChanged(TimePicker view, int hourOfDay, int minute) {
            textView.setText(String.format("현재 시간 : %2d:%2d",
                                            hourOfDay, minute));
        }
    });
}
```



TimePicker 예제 4

■ JAVA 프로그램

```
public class MainActivity extends AppCompatActivity {  
    TextView textView;  
    Button button;  
    int mHour = 0;  
    int mMinute = 0;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        textView = findViewById(R.id.result);  
        button = findViewById(R.id.button);  
    }  
}
```



TimePicker 예제 4

■ JAVA 프로그램

```
button.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        TimePickerDialog timePickerDialog;  
        timePickerDialog = new TimePickerDialog(MainActivity.this,  
            timeSetListener, mHour, mMinute,true);  
        timePickerDialog.show();  
    }  
});
```

```
Calendar calendar = Calendar.getInstance();  
mHour = calendar.get(Calendar.HOUR);  
mMinute = calendar.get(Calendar.MINUTE);  
updateDisplay();  
}
```



TimePicker 예제 4

■ JAVA 프로그램

```
TimePickerDialog.OnTimeSetListener timeSetListener =  
    new TimePickerDialog.OnTimeSetListener() {  
        @Override  
        public void onTimeSet(TimePicker view, int hourOfDay, int minute) {  
            mHour = hourOfDay;  
            mMinute = minute;  
            updateDisplay();  
        }  
    };  
  
private void updateDisplay() {  
    textView.setText(mHour + " 시 " + mMinute + " 분 ");  
}  
}
```