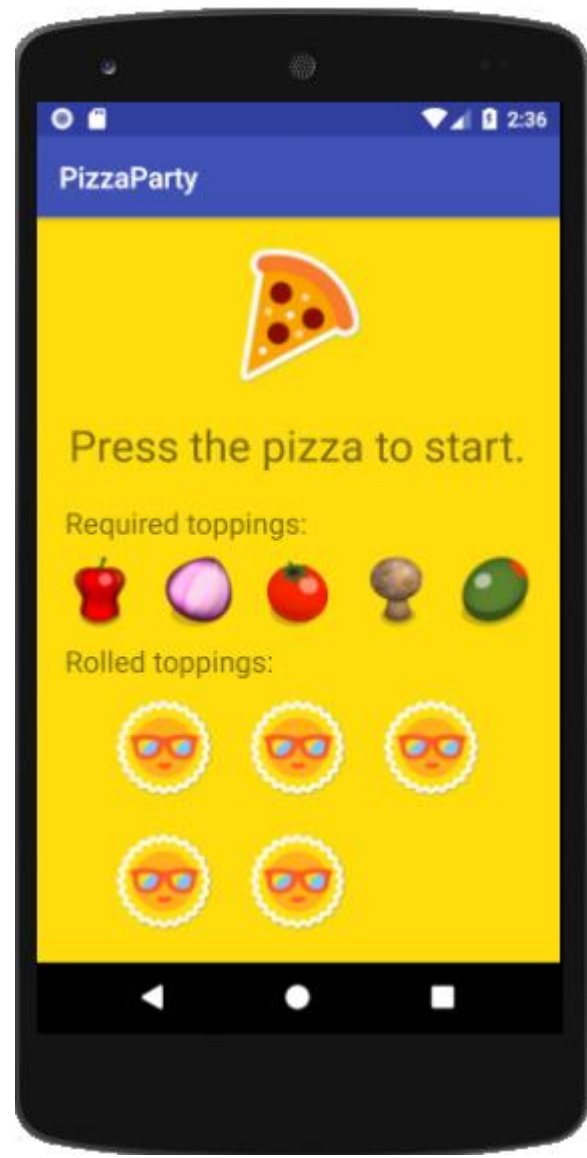
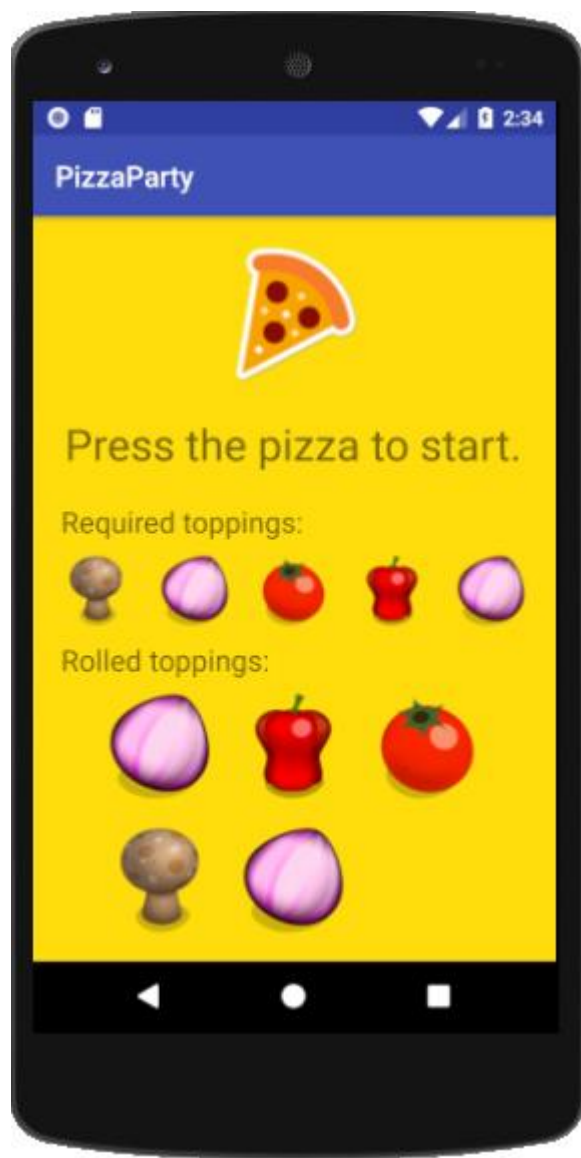
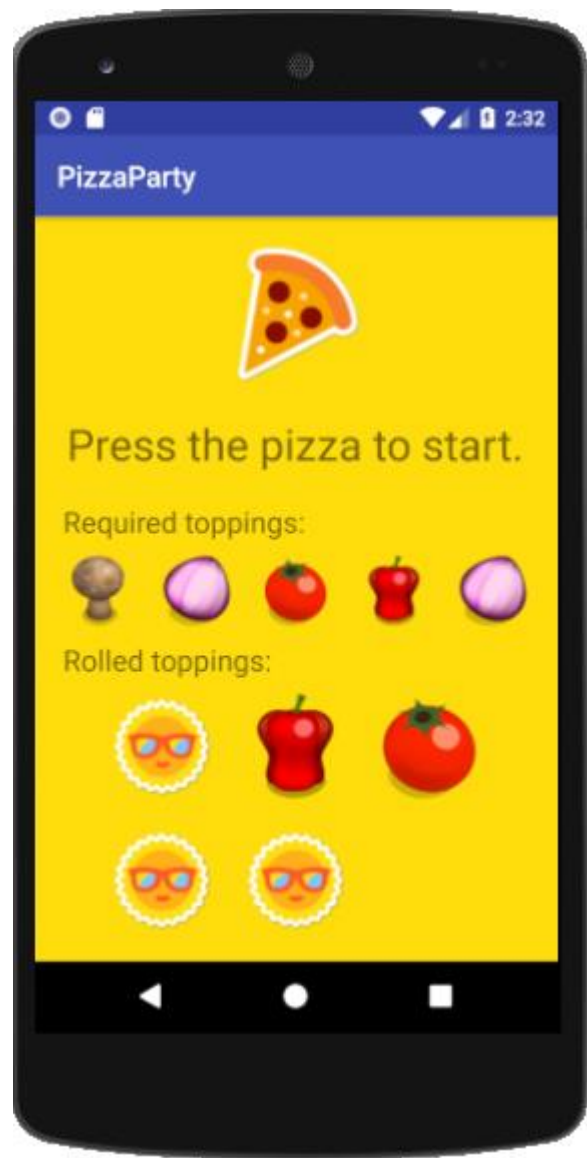
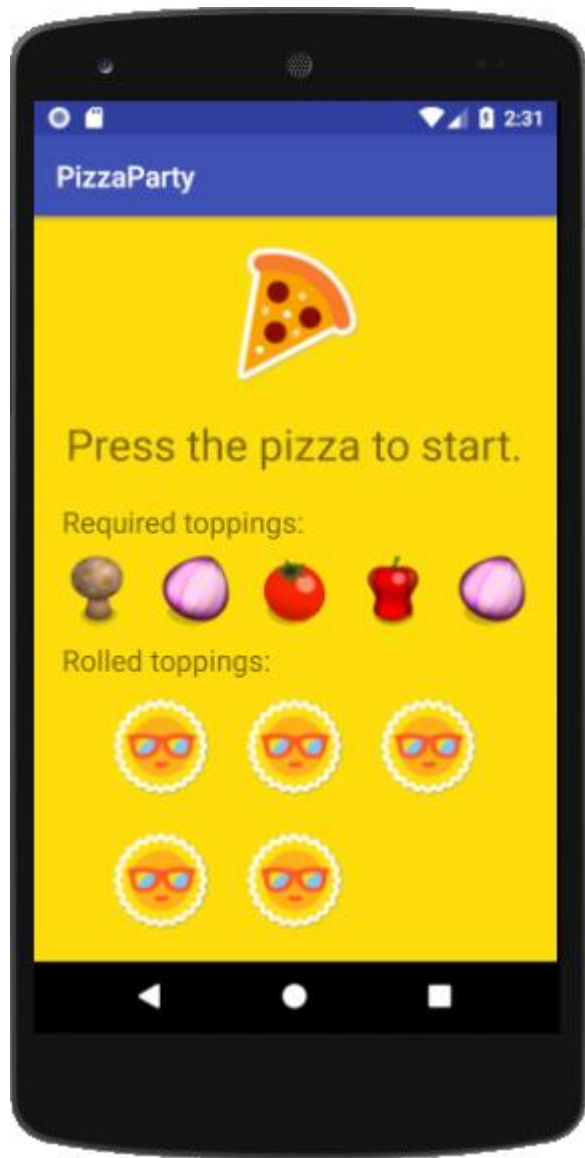


# The Pizza Party





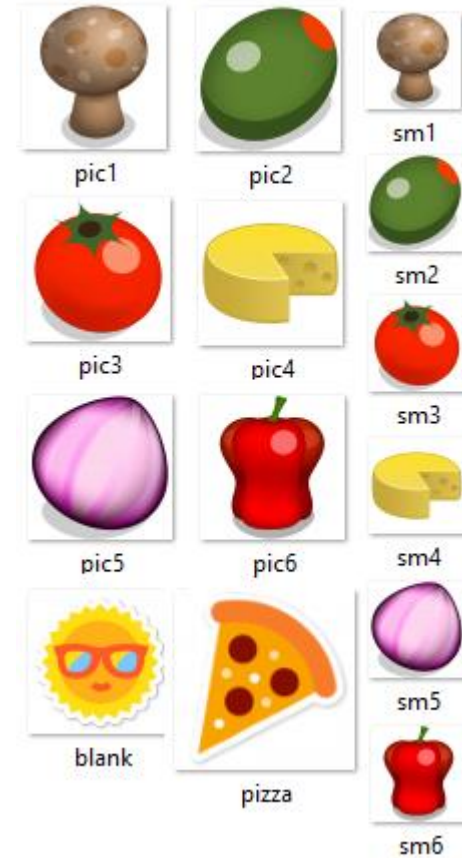
```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="center_horizontal"
    android:orientation="vertical"
    android:background="#FFDD0D">
```

```
<ImageView
    android:id="@+id/go"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:onClick="goClick"
    android:padding="20dp"
    android:src="@drawable/pizza" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingBottom="20dp"
    android:text="Press the pizza to start."
    android:textSize="30dp"
    android:layout_gravity="center_horizontal"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Required toppings:"
    android:paddingLeft="20dp"
    android:textSize="20dp" />
```

@drawable folder





```
<LinearLayout
```

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="center_horizontal"  
    android:orientation="horizontal">
```

```
<ImageView
```

```
    android:id="@+id/r1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:src="@drawable/sm1" />
```

```
<ImageView
```

```
    android:id="@+id/r2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:src="@drawable/sm6" />
```

```
<ImageView
```

```
    android:id="@+id/r3"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:src="@drawable/sm3" />
```

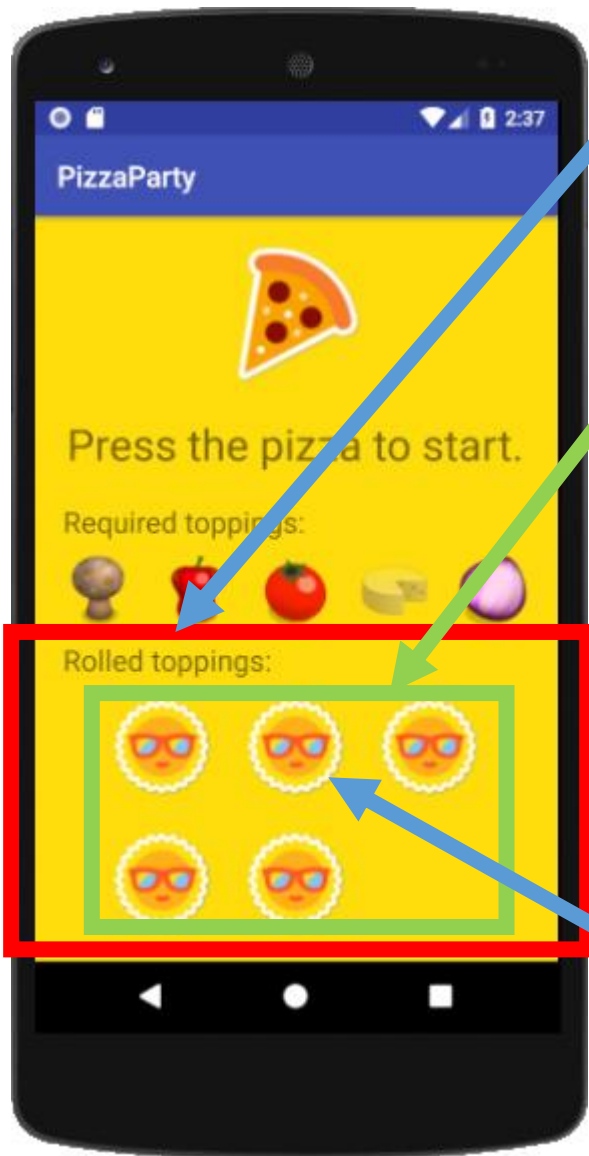
```
<ImageView
```

```
    android:id="@+id/r4"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:src="@drawable/sm4" />
```

```
<ImageView
```

```
    android:id="@+id/r5"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:src="@drawable/sm5" />
```

```
</LinearLayout>
```



```
<TextView
```

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Rolled toppings:"  
    android:paddingLeft="20dp"  
    android:textSize="20dp" />
```

```
<GridLayout
```

```
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:columnCount="3"  
    android:rowCount="2"  
    android:layout_gravity="center_horizontal">
```

```
<ImageView
```

```
    android:id="@+id/dice1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:onClick="dice1Click"  
    android:padding="10dp"  
    android:src="@drawable/blank" />
```

```
<ImageView
```

```
    android:id="@+id/dice2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:onClick="dice2Click"  
    android:padding="10dp"  
    android:src="@drawable/blank" />
```

```
<ImageView
```

```
    android:id="@+id/dice3"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:onClick="dice3Click"  
    android:padding="10dp"  
    android:src="@drawable/blank" />
```

```
<ImageView
```

```
    android:id="@+id/dice4"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:onClick="dice4Click"  
    android:padding="10dp"  
    android:src="@drawable/blank" />
```

```
<ImageView
```

```
    android:id="@+id/dice5"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:onClick="dice5Click"  
    android:padding="10dp"  
    android:src="@drawable/blank" />
```

```
</GridLayout>
```

```
</LinearLayout>
```

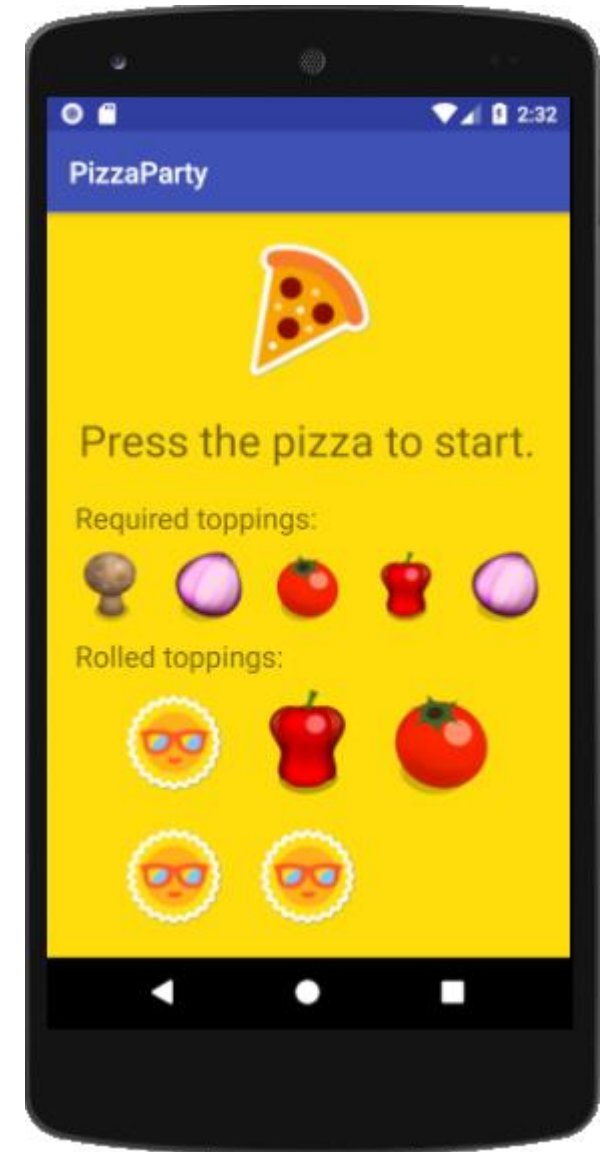
Do NOT cut and  
paste java files

```
package ca.gorskicompsci.www.pizzaparty;
```

```
import android.media.Image;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.ImageView;
```

```
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
}
```

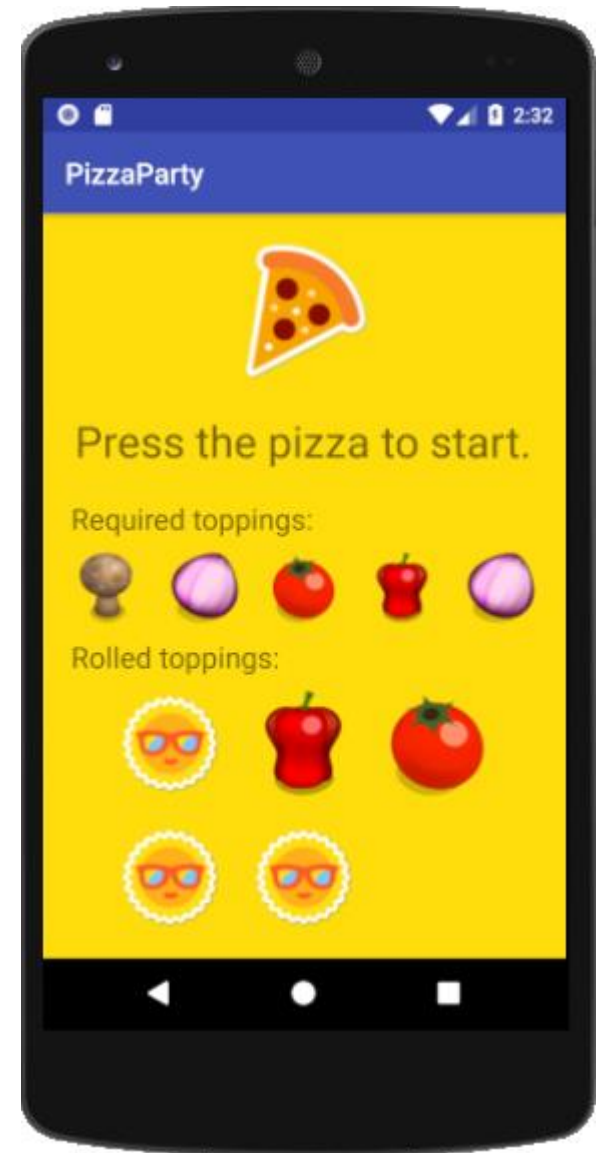
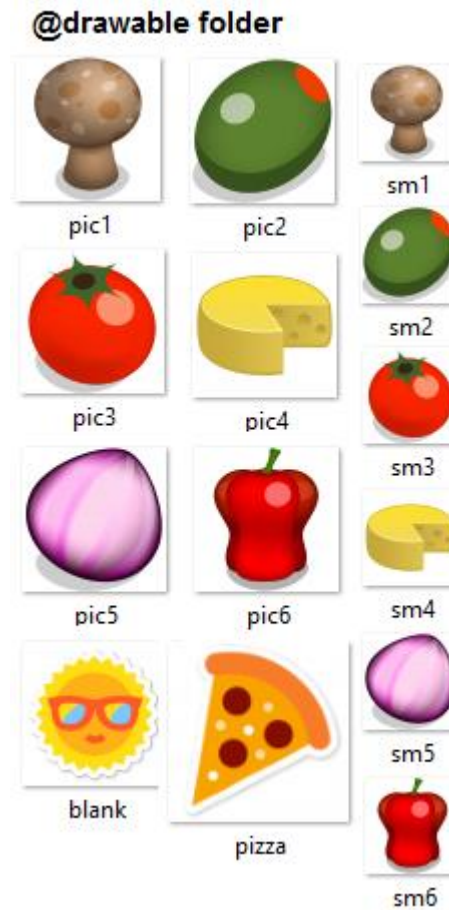
Do NOT  
delete!!!



```

public void dice1Click(View view) {
    ImageView i = (ImageView) findViewById(R.id.dice1);
    int num = (int) (Math.random() * 6);
    if (num == 1)
        i.setImageResource(R.drawable.pic1);
    else if (num == 2)
        i.setImageResource(R.drawable.pic2);
    else if (num == 3)
        i.setImageResource(R.drawable.pic3);
    else if (num == 4)
        i.setImageResource(R.drawable.pic4);
    else if (num == 5)
        i.setImageResource(R.drawable.pic5);
    else
        i.setImageResource(R.drawable.pic6);
}

```



```
public void dice1Click(View view) {
    ImageView i = (ImageView) findViewById(R.id.dice1);
    int num = (int) (Math.random() * 6);
    if (num == 1)
        i.setImageResource(R.drawable.pic1);
    else if (num == 2)
        i.setImageResource(R.drawable.pic2);
    else if (num == 3)
        i.setImageResource(R.drawable.pic3);
    else if (num == 4)
        i.setImageResource(R.drawable.pic4);
    else if (num == 5)
        i.setImageResource(R.drawable.pic5);
    else
        i.setImageResource(R.drawable.pic6);
}
```

```
public void dice2Click(View view) {
    ImageView i = (ImageView) findViewById(R.id.dice2);
    int num = (int) (Math.random() * 6);
    if (num == 1)
        i.setImageResource(R.drawable.pic1);
    else if (num == 2)
        i.setImageResource(R.drawable.pic2);
    else if (num == 3)
        i.setImageResource(R.drawable.pic3);
    else if (num == 4)
        i.setImageResource(R.drawable.pic4);
    else if (num == 5)
        i.setImageResource(R.drawable.pic5);
    else
        i.setImageResource(R.drawable.pic6);
}
```

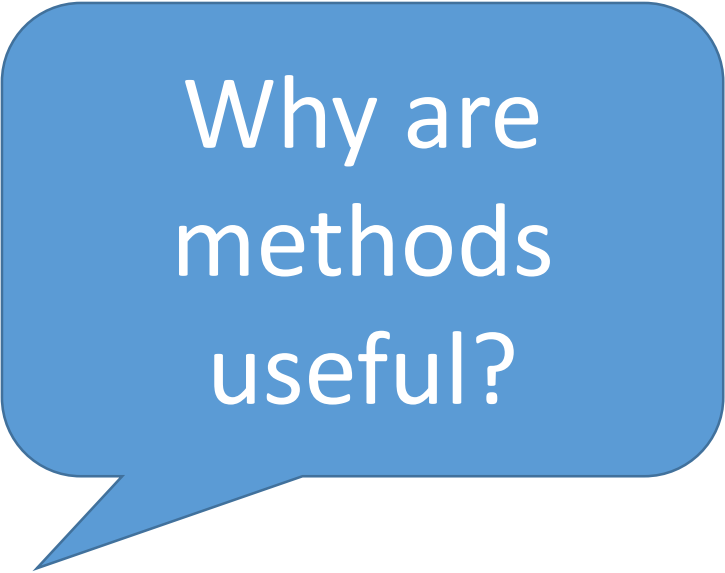
```
public void dice3Click(View view) {
    ImageView i = (ImageView) findViewById(R.id.dice3);
    int num = (int) (Math.random() * 6);
    if (num == 1)
        i.setImageResource(R.drawable.pic1);
    else if (num == 2)
        i.setImageResource(R.drawable.pic2);
    else if (num == 3)
        i.setImageResource(R.drawable.pic3);
    else if (num == 4)
        i.setImageResource(R.drawable.pic4);
    else if (num == 5)
        i.setImageResource(R.drawable.pic5);
    else
        i.setImageResource(R.drawable.pic6);
}
```

```
public void dice4Click(View view) {
    ImageView i = (ImageView) findViewById(R.id.dice4);
    int num = (int) (Math.random() * 6);
    if (num == 1)
        i.setImageResource(R.drawable.pic1);
    else if (num == 2)
        i.setImageResource(R.drawable.pic2);
    else if (num == 3)
        i.setImageResource(R.drawable.pic3);
    else if (num == 4)
        i.setImageResource(R.drawable.pic4);
    else if (num == 5)
        i.setImageResource(R.drawable.pic5);
    else
        i.setImageResource(R.drawable.pic6);
}
```



When you find yourself cutting  
and pasting code, STOP.

You need a method instead.



Why are  
methods  
useful?

Organization

Reusability

Abstraction

Testing

Extensibility

## Organization

Breaks things up into smaller logical units.  
Think: Tasks in your RPG from last year.

## Reusability

Instead of copy/pasting code, you call the method.  
Think: Less chance of copy/paste errors.

## Abstraction

To use someone else's code, you only need the method signature – then you can call it. You don't need to understand the details. Think: IO

## Testing

When we don't repeat code (we call a method instead), it is easier to test. There are fewer lines for white box testing. Think: Testing the if 5 times!

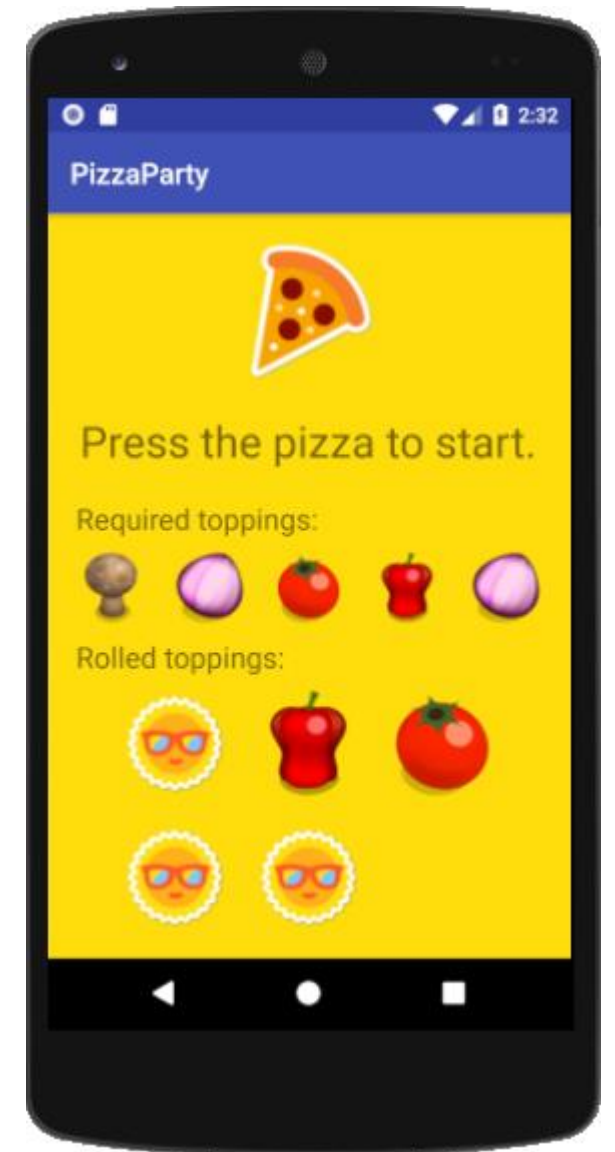
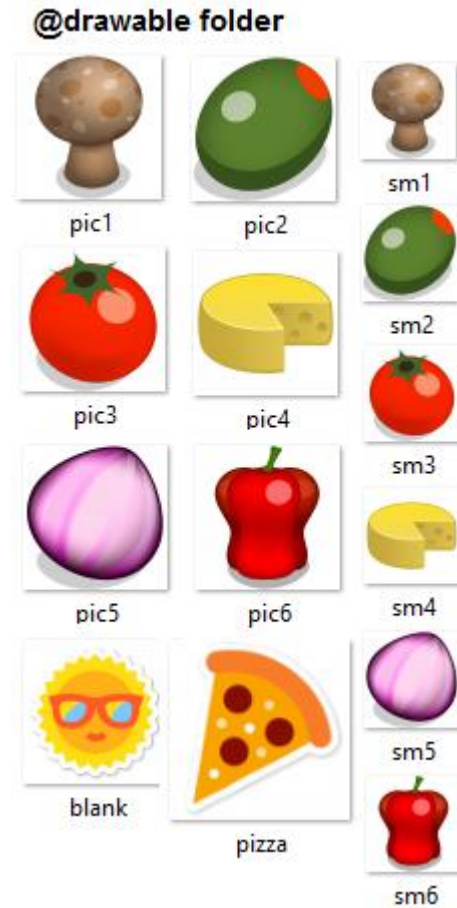
## Extensibility

By putting repeated code in one place, when we want to change it, we only need to change one place – not all of them.

```

public void setPic (ImageView i, int num) {
    if (num == 1 && type == 'b')
        i.setImageResource (R.drawable.pic1);
    else if (num == 2 && type == 'b')
        i.setImageResource (R.drawable.pic2);
    else if (num == 3 && type == 'b')
        i.setImageResource (R.drawable.pic3);
    else if (num == 4 && type == 'b')
        i.setImageResource (R.drawable.pic4);
    else if (num == 5 && type == 'b')
        i.setImageResource (R.drawable.pic5);
    else if (num == 0 && type == 'b')
        i.setImageResource (R.drawable.pic6);
    else
        i.setImageResource (R.drawable.blank);
}

```



```
public void setPic(ImageView i, int num)
```

```
parameterType parameter = value;  
returnType variable = methodName (parameter);
```



```
public void setPic(ImageView i, int num)
```

```
parameterType parameter = value;  
returnType variable = methodName (parameter);
```



```
public void setPic(ImageView i, int num)
```

```
int num = (int) (Math.random() * 6);  
ImageView dice1 = (ImageView) findViewById(R.id.dice1);  
setPic(dice1, num);
```

```
parameterType parameter = value;  
returnType variable = methodName (parameter);
```

```

public void dice1Click(View view) {
    int num = (int) (Math.random() * 6);
    ImageView dice1 = (ImageView) findViewById(R.id.dice1);
    setPic(dice1, num);
}

public void setPic(ImageView i, int num) {
    if (num == 1)
        i.setImageResource(R.drawable.pic1);
    //etc.
}

```

