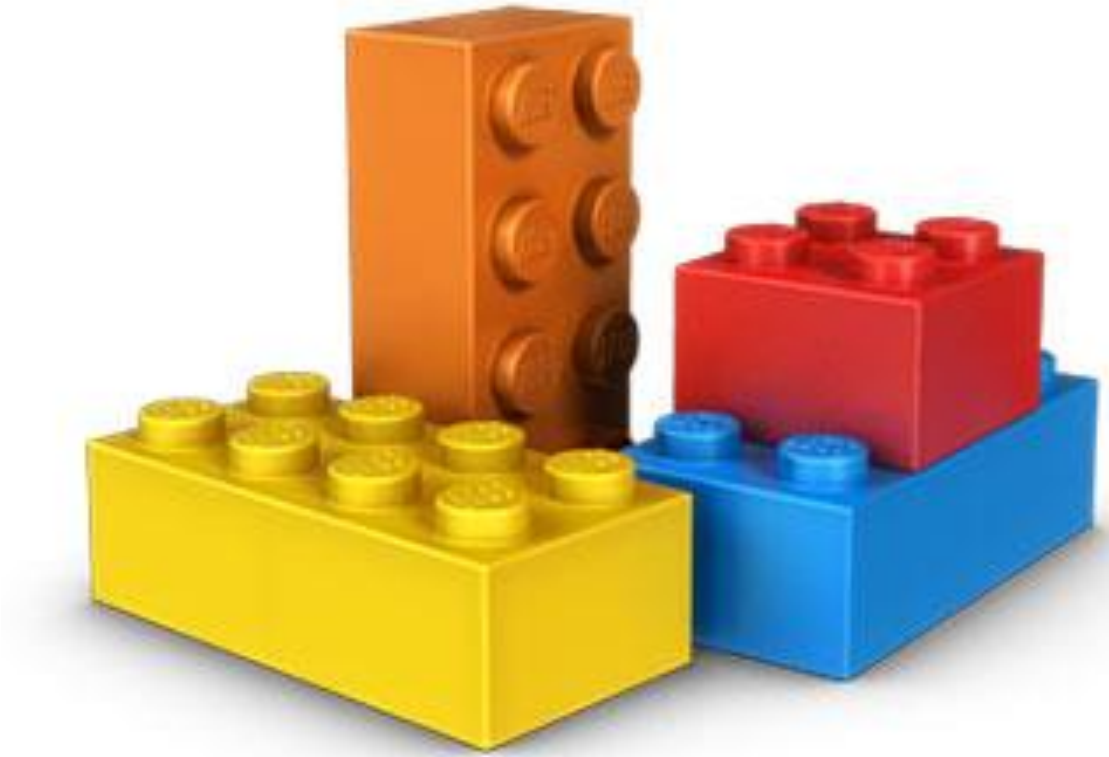
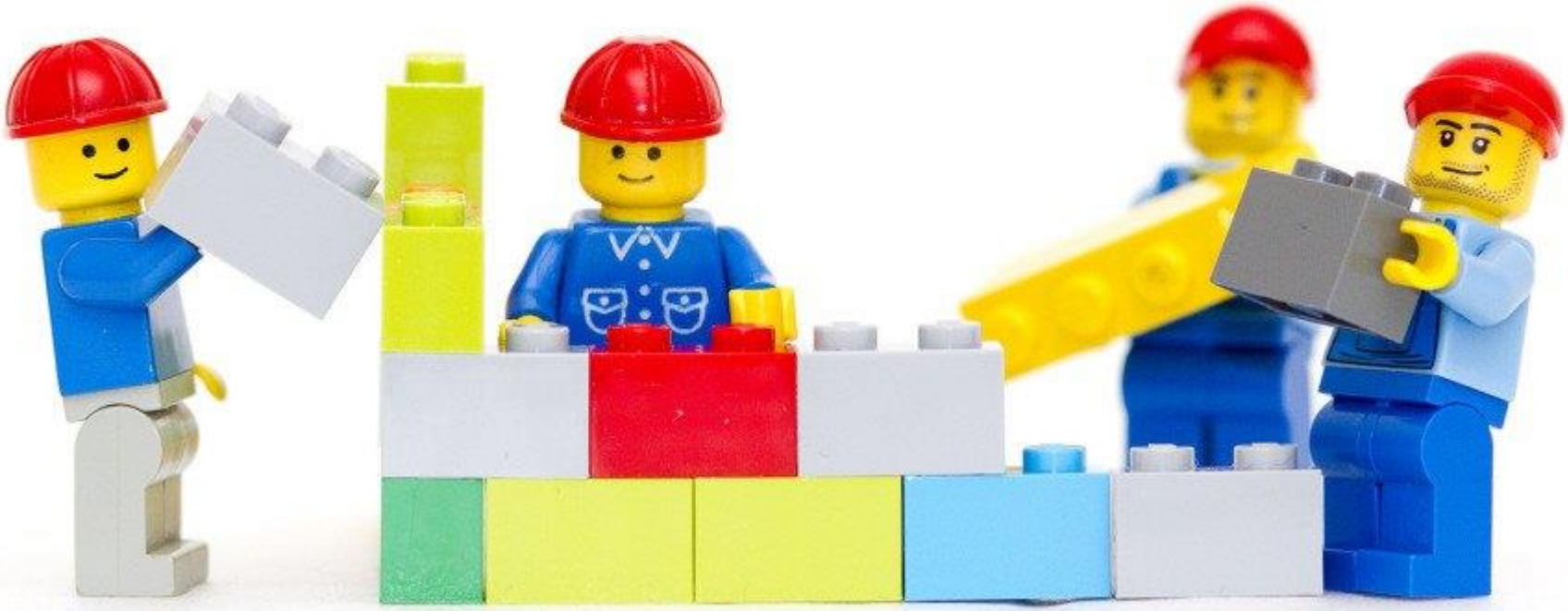
Pieces of a Method

Identification

Method

- A subprogram.
- Used to break a larger program into smaller pieces.





```
public class example
{
    public static void main (String args[])
    {
        new example ();
    }

    public example ()
    {
        circle ();
        int r = IO.inputInt ("What is the radius? ");
        System.out.println ("The area is " + area (r));
        circle ();
    }

    public void circle ()
    {
        System.out.println ("        **");
        System.out.println ("       *  *");
        System.out.println ("        **");
    }

    public double area (int radius)
    {
        double area = Math.PI * radius * radius;
        return area;
    }
}
```

Main method

Constructor

Circle Method

Area Method

```
public class MainActivity extends AppCompatActivity {

    int board[][] = new int[3][3];
    int turn = 1;
```

onCreate

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

```
    public void flip(ImageView i) {
        ImageView turnpic = (ImageView) findViewById(R.id.turn);
        if (turn == 1) {
            i.setImageResource(R.drawable.opic);
            turn = 2;
            turnpic.setImageResource(R.drawable.xturn);
        } else {
            i.setImageResource(R.drawable.xpic);
            turn = 1;
            turnpic.setImageResource(R.drawable.oturn);
        }
    }
```

flip

Method Signature

- First line of the method
- Very important because it specifies all of the input and output of the method AND it’s name

Method Name

```
public void circle ()
{
    System.out.println ("         **");
    System.out.println ("       *   *");
    System.out.println ("         **");
}

public double area (int radius)
{
    double area = Math.PI * radius * radius;
    return area;
}
```

Method Name

```
public void flip(ImageView i) {
    ImageView turnpic = (ImageView) findViewById(R.id.turn);
    if (turn == 1) {
        i.setImageResource(R.drawable.opic);
        turn = 2;
        turnpic.setImageResource(R.drawable.xturn);
    } else {
        i.setImageResource(R.drawable.xpic);
        turn = 1;
        turnpic.setImageResource(R.drawable.oturn);
    }
}

public void aClick(View view) {
    if (board[0][0] == 0) {
        ImageView i = (ImageView) findViewById(R.id.a);
        board[0][0] = turn;
        flip(i);
    } else {
        Toast.makeText(getApplicationContext(), "Place already taken",
           Toast.LENGTH_SHORT).show();
    }
    win();
}
```

Parameter

- A variable sent into a method
- It has a type(possibly a view type) and a name (follows Id's naming rules
- INPUT of the method

Parameter Names

```
public void circle ()
{
    System.out.println ("         **");
    System.out.println ("       *   *");
    System.out.println ("         **");
}

public double area (int radius)
{
    double area = Math.PI * radius * radius;
    return area;
}
```

Parameter Names

```
public void flip(ImageView i) {
    ImageView turnpic = (ImageView) findViewById(R.id.turn);
    if (turn == 1) {
        i.setImageResource(R.drawable.opic);
        turn = 2;
        turnpic.setImageResource(R.drawable.xturn);
    } else {
        i.setImageResource(R.drawable.xpic);
        turn = 1;
        turnpic.setImageResource(R.drawable.oturn);
    }
}

public void aClick(View view) {
    if (board[0][0] == 0) {
        ImageView i = (ImageView) findViewById(R.id.a);
        board[0][0] = turn;
        flip(i);
    } else {
        Toast.makeText(getApplicationContext(), "Place already taken",
           Toast.LENGTH_SHORT).show();
    }
    win();
}
```

Parameter Types

```
public void circle ()
{
    System.out.println ("         **");
    System.out.println ("       *   *");
    System.out.println ("         **");
}

public double area (int radius)
{
    double area = Math.PI * radius * radius;
    return area;
}
```

Parameter Types

```
public void flip(ImageView i) {
    ImageView turnpic = (ImageView) findViewById(R.id.turn);
    if (turn == 1) {
        i.setImageResource(R.drawable.opic);
        turn = 2;
        turnpic.setImageResource(R.drawable.xturn);
    } else {
        i.setImageResource(R.drawable.xpic);
        turn = 1;
        turnpic.setImageResource(R.drawable.oturn);
    }
}

public void aClick(View view) {
    if (board[0][0] == 0) {
        ImageView i = (ImageView) findViewById(R.id.a);
        board[0][0] = turn;
        flip(i);
    } else {
        Toast.makeText(getApplicationContext(), "Place already taken",
           Toast.LENGTH_SHORT).show();
    }
    win();
}
```

Return Type

- The value sent OUT of the method.
- The value is sent back using the “return” line
- It must be the same type specified in the method signature.

Return Type

```
public void circle ()
{
    System.out.println ("         **");
    System.out.println ("       *   *");
    System.out.println ("         **");
}

public double area (int radius)
{
    double area = Math.PI * radius * radius;
    return area;
}
```

Return Type

```
public void flip(ImageView i) {
    ImageView turnpic = (ImageView) findViewById(R.id.turn);
    if (turn == 1) {
        i.setImageResource(R.drawable.opic);
        turn = 2;
        turnpic.setImageResource(R.drawable.xturn);
    } else {
        i.setImageResource(R.drawable.xpic);
        turn = 1;
        turnpic.setImageResource(R.drawable.oturn);
    }
}

public void aClick(View view) {
    if (board[0][0] == 0) {
        ImageView i = (ImageView) findViewById(R.id.a);
        board[0][0] = turn;
        flip(i);
    } else {
        Toast.makeText(getApplicationContext(), "Place already taken",
           Toast.LENGTH_SHORT).show();
    }
    win();
}
```

Return Line

```
public void circle ()
{
    System.out.println ("         **");
    System.out.println ("       *   *");
    System.out.println ("         **");
}

public double area (int radius)
{
    double area = Math.PI * radius * radius;
    return area;
}
```

Return Line

```
public void flip(ImageView i) {
    ImageView turnpic = (ImageView) findViewById(R.id.turn);
    if (turn == 1) {
        i.setImageResource(R.drawable.opic);
        turn = 2;
        turnpic.setImageResource(R.drawable.xturn);
    } else {
        i.setImageResource(R.drawable.xpic);
        turn = 1;
        turnpic.setImageResource(R.drawable.oturn);
    }
}

public void aClick(View view) {
    if (board[0][0] == 0) {
        ImageView i = (ImageView) findViewById(R.id.a);
        board[0][0] = turn;
        flip(i);
    } else {
        Toast.makeText(getApplicationContext(), "Place already taken",
           Toast.LENGTH_SHORT).show();
    }
    win();
}
```

```
public class example
{
    public static void main (String args[])
    {
        new example ();
    }

    public example ()
    {
        circle ();
        int r = IO.inputInt ("What is the radius? ");
        System.out.println ("The area is " + area (r));
        circle ();
    }

    public void circle ()
    {
        System.out.println ("        **");
        System.out.println ("      *  *");
        System.out.println ("        **");
    }

    public double area (int radius)
    {
        double area = Math.PI * radius * radius;
        return area;
    }
}
```

Main method

Constructor

Circle Method

Area Method

```
public example ()
{
    circle ();
    int r = IO.inputInt ("What is the radius? ");
    System.out.println ("The area is " + area (r));
    circle ();
}
```

```
  **
 *  *
  **
What is the radius? 6
The area is 113.09733552923255
  **
 *  *
  **
```

```
public example ()
{
    circle ();
    int r = IO.inputInt ("What is the radius? ");
    System.out.println ("The area is " + area (r));
    circle ();
}
```

Method call of circle.

Method call of circle.

Method call of area.

```
  **
 *  *
  **
What is the radius? 6
The area is 113.09733552923255
  **
 *  *
  **
```

Method Call

```
public void flip(ImageView i) {
    ImageView turnpic = (ImageView) findViewById(R.id.turn);
    if (turn == 1) {
        i.setImageResource(R.drawable.opic);
        turn = 2;
        turnpic.setImageResource(R.drawable.xturn);
    } else {
        i.setImageResource(R.drawable.xpic);
        turn = 1;
        turnpic.setImageResource(R.drawable.oturn);
    }
}

public void aClick(View view) {
    if (board[0][0] == 0) {
        ImageView i = (ImageView) findViewById(R.id.a);
        board[0][0] = turn;
        flip(i);
    } else {
        Toast.makeText(getApplicationContext(), "Place already taken",
           Toast.LENGTH_SHORT).show();
    }
    win();
}
```