

A Game in XForms



Contents

1. Contents
2. Youngest: How do you program a game?
3. Me: What sort of game?
4. Youngest:
5. Take some data
6. Display it
7. Display it
8. Result
9. Add interaction
10. Add an action
11. Add in the action
12. Result
13. Add a condition
14. Condition
15. Result
16. "It would be more fun if it was like a jigsaw"
17. Easy
18. Easy
19. Result

Youngest: How do you program a game?

**Me: What sort of
game?**

Youngest:



Take some data

Four rows of four cells

```
<instance>
  <game xmlns="">
    <row><cell>1</cell><cell>5</cell><cell>8</cell><cell>12</cell></row>
    <row><cell>2</cell><cell>6</cell><cell>9</cell><cell>13</cell></row>
    <row><cell>3</cell><cell>7</cell><cell>10</cell><cell>14</cell></row>
    <row><cell>4</cell><cell>.</cell><cell>11</cell><cell>15</cell></row>
  </game>
</instance>
```

Display it

Four rows of four cells

```
<instance>
  <game xmlns="">
    <row><cell>1</cell><cell>5</cell><cell>8</cell><cell>12</cell></row>
    <row><cell>2</cell><cell>6</cell><cell>9</cell><cell>13</cell></row>
    <row><cell>3</cell><cell>7</cell><cell>10</cell><cell>14</cell></row>
    <row><cell>4</cell><cell>.</cell><cell>11</cell><cell>15</cell></row>
  </game>
</instance>
```

```
<repeat ref="row">
  <repeat ref="cell" >
    <output value="."/>
  </repeat>
</repeat>
```

Display it

Four rows of four cells

```
<instance>
  <game xmlns="">
    <row><cell>1</cell><cell>5</cell><cell>8</cell><cell>12</cell></row>
    <row><cell>2</cell><cell>6</cell><cell>9</cell><cell>13</cell></row>
    <row><cell>3</cell><cell>7</cell><cell>10</cell><cell>14</cell></row>
    <row><cell>4</cell><cell>.</cell><cell>11</cell><cell>15</cell></row>
  </game>
</instance>
```

```
<repeat ref="row">
  <repeat ref="cell" >
    <output value="."/>
  </repeat>
</repeat>
```

Add a dash of CSS.

Result

1	5	8	12
2	6	9	13
3	7	10	14
4	.	11	15

Of course, this doesn't do anything yet.

Add interaction

```
<repeat ref="row">
  <repeat ref="cell" >
    <trigger appearance="minimal">
      <label><output value="."/></label>
    </trigger>
  </repeat>
</repeat>
```

trigger is the XForms version of HTML's button, but XForms tries to be device-neutral.

"appearance="minimal" means it isn't displayed as a button.

It still doesn't do anything.

Add an action

```
<repeat ref="row">
  <repeat ref="cell" >
    <trigger appearance="minimal">
      <label><output value="."/></label>
      <action ev:event="DOMActivate">
        ...
      </action>
    </trigger>
  </repeat>
</repeat>
```

This says to respond to the trigger being activated (eg a click).

Add in the action

```
<repeat ref="row">
  <repeat ref="cell" >
    <trigger appearance="minimal">
      <label><output value="."/></label>
      <action ev:event="DOMActivate">
        <setvalue ref="//cell[.='']" value="context()"/>
        <setvalue ref="." value="'.'"/>
      </action>
    </trigger>
  </repeat>
</repeat>
```

If you click on any of the squares, it will swap the value there with the square holding "."

Result

1	5	8	12
2	6	9	13
3	7	10	14
4	.	11	15

Add a condition

We only want to allow swapping the empty square with an adjacent one.

Add a condition to the action.

```
<repeat ref="row">
  <repeat ref="cell" >
    <trigger appearance="minimal">
      <label><output value="."/></label>
      <action ev:event="DOMActivate" if="...">
        <setvalue ref="//cell[.='.']" value="context()"/>
        <setvalue ref="." value="'.'" />
      </action>
    </trigger>
  </repeat>
</repeat>
```

Condition

Only do the action if the next or previous cell, or the cell at the same position in the previous/next row is ".".

```
following-sibling::cell[1]='.' or
preceding-sibling::cell[1]='.' or
../following-sibling::row[1]/
cell[1+count(context()/preceding-sibling::cell)]= '.' or
../preceding-sibling::row[1]/
cell[1+count(context()/preceding-sibling::cell)]= '.'
```

Result

1	5	8	12
2	6	9	13
3	7	10	14
4	.	11	15

**"It would be more fun
if it was like a jigsaw"**

Easy

```
<trigger appearance="minimal">  
  <label><output mediatype="image/*" value="."/></label>
```

This says: Interpret the value as a URL and display it as an image.

Only problem: you may not have a file called ".".

Easy

```
<trigger appearance="minimal">  
  <label><output mediatype="image/*" value="if(.='.', 'blank', .)" /></lab
```

Result

